A SMART TOY SYSTEM TO EDUCATE CHILDREN WITH AUTISM: A COMPARISON OF TREE-STRUCTURED PROGRAMMING AND MACHINE LEARNING MODELS FOR AN IoT DEVICE

BY

GERRY DERKSEN

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Informatics
in the Graduate College of the
University of Illinois Urbana-Champaign, 2024

Urbana, Illinois

Doctoral Committee:

Professor Stan Ruecker, Chair
Professor Michael Twidale
Associate Professor Robb Lundgren
Assistant Professor Juan Salamanca

Abstract

In this study, we examine the question of individualized versus average learning using a toy that allows children with mild autism to play color-matching games and math games. The toy was first configured with a decision tree that was the same for the first group of 16 participants. The toy was then reconfigured to use a machine learning (ML) system that customized the experience for each member of the second group of 16 participants. The toy was equipped with a number of sensors that were used by both groups to support the decision as to whether the student should get the next question that was easier, the same level, or harder than the previous question. The sensors included force sensors on the buttons, an accelerometer in the circuit board, and an internal clock to track duration. The other features were captured using cloud service for the Internet of Things (IoT) broker MQTT, which captures performance, difficulty level, and "frustration score," which is the basis for one of the software's predictions. The broker also controls the LED's used for the math and color sequence game.

One purpose of the individualized levels of difficulty is to help maintain engagement. It is among the most critical aspects of understanding how people learn. Engaging in a subject is the first step toward learning because it can motivate, stimulate, and instigate a cycle of observing and applying the results of new knowledge and applying it again. It is the precursor to when learning is about to happen or currently taking place. If a student's level of engagement is lost, so is the possibility of learning. This is often due to the targeted level of the information not hitting the audience when they need it. It is hard to struggle through if the information is too dense or complex. Too easy of a task becomes mundane and uninteresting. Learning a subject is often abandoned unless there is something to regain their interest and willingness to return to the subject. These scenarios become even more frustrating if you have autism. Autism spectrum disorder is a mixed array of challenges that are not the same in every individual and to a greater or lesser degree between people. A teaching aid that adjusts to each child's learning level is at the center of this study, and it attempts to capture the information needed to make these adjustments and improve the learning experience for children with autism.

This study aims to determine if an educational toy can predict when levels of engagement are such that learning is probable or if it is necessary to re-engage the learner by making the problem less complex, for example. Understanding engagement from a psychological perspective often takes the form of a mental model to construct what is available to a student through perception stimuli, internal motivation, and the ability to assess the problem and other cognitive triggers. Evaluation of the model's mix of complex cognitive triggers only provides the potential for successful learning but does not measure the outcome to see if learning occurred. In this study, an intelligent toy captures physical and cognitive manifestations and

performance assessments to quantify learning outcomes. This data is collected to map those outcomes in an effort to train the toy on how to predict future outcomes and modify its behavior to accommodate the student's level of engagement by increasing, decreasing, or steadying the level of difficulty. Game design considerations trigger cognitive stimuli seeking to find physical and cognitive data points corresponding to success and failure responses.
As it turns out, engagement was slightly but significantly increased in the ML version of the toy. On average, the ML game group played 29 minutes, while the tree-structured group played 22 minutes. The ML group also played 14 sequences on average, and the TS group played an average of 12.5.

The next questions we asked concerned the students' performance (how many answers they got right) and the toy's performance (did it make useful predictions), which is to say, was the student appropriately challenged. Toy predictions Results were analyzed using both MANOVA and Kruskal-Wallis. MANOVA is a stronger method but requires parametric data, while Kruskal-Wallis is intended for non-parametric data. In the case of this study, the data formed a bell curve, but the curve was cut off on the left. This means that the data was not parametric, but there have been arguments in the literature for the use of MANOVA in similar circumstances since real-world data is often messy.

The results indicate a significant difference between the performance of the toy versions; however, the influence of more accurate predictions on a player's performance is in doubt. The toy's ML prediction rate is a moderate 68%, and although rates over 75% were targeted, the realities of real-world data collection have hindered its performance. Despite this, the MANOVA analysis suggests a significant difference of 2.43 out of 5 questions versus 2.27 out of 5 between the two software's impact on player performance or how often the player answered correctly. However, more study is needed to determine the validity of this finding since Kruskal-Wallis indicates no significant difference.

Our final question related to the possibility that the sensors would form super-clusters in terms of their influence on the prediction. If they did, it would be easier to determine which combination of features is most influential. In the design of the toy, pressure values exerted on buttons and motion data collected from the accelerometer were built into the toy and clustered into distinct groups. Still, connections between pressure and motion and the other data features did not reinforce these super-clusters. For example, ranges of pressure data fell into four distinct touch levels 1200-2100, 2100-2700, 2500-3300, and 2600-3600 using peak range values; within the light-touch group, each player had a unique motion pattern. We would have expected that if a player generally had a light touch on the buttons, they might move the toy less or less often. However, this is not the case: each light touch player has a unique motion pattern. This was true

of all the pressure data groups. Statistically, each feature did have a significant impact on the prediction of the toy. However, without these super-clustered groups influencing the prediction, it is difficult to determine which combination of features is most influential. The mix of results is supported by the literature on educational research and also underlines the complexity of learning further complicated by the variability of autism.

Acknowledgments

I have made academia central to my work, to my life, and my being, always finding the people in my chosen field to be generous with their time, knowledge, and support. I am grateful to my committee, which has continued to provide these valuable assets. I also thank my classmates, professors, and administration at the University of Illinois as well as my home institution for contributing to my success. Without them this would never have been possible.

There is a group of people I consider my inner circle, and to them, I owe you gratitude and all my admiration. To Stan, Grant, Frank, and my loving family, I thank you for your support and guidance, which was more than intellectual or moral support and more than I could have expected.

I know there are many who have struggled in this same pursuit to overcome setbacks and adversity, sacrificing much to gain just a bit more insight into their chosen field. I only wish to have shared it with the person who knew me best. Her support, kindness, and love carried me very far through this process, and I know she would have been so very proud. Thank you for all that you have given me. It is to you Zhabiz, that I dedicate this work, to our shared love of design and for each other.

Table of Contents

Chapter 1 - An Introduction to Specialized Education

As the use of technology in the classroom increases, a more significant schism occurs between neurodiverse and neurotypical populations. In 2018, the Ontario Human Rights Commission wrote about the disparity of access between these two populations in a policy report, “Accessible Education for Students with Disabilities,” which lists insufficient resources as the second most important cause (OHRC, 2018). We often think of these problems in terms of economics. However, the available educational technology for children with disabilities also lacks specialized tools, which are more equitable, according to Martha Smit of the Learning Disabilities Association. The notion of equity, Smit continues, is the distinction that each learner has access to tools they need to learn and thrive (Smit, 2022). The OHRC report on individualized accommodation places greater emphasis on equity over equality: "There is no set formula for accommodation. Each student's needs are unique and must be considered afresh when an accommodation request is made. The emphasis must always be on the individual student’s needs, not the type of disability.” This shift would suggest changes in attitude, but how is this practically accomplished? The answer lies in the notion of flexibility to accommodate change. Systems, data, and the tools used to present, interact, and learn must be designed and built with this in mind.

To address the problem of flexibility within educational devices, this research proposes a toy that is AI enabled. The artificial intelligence is used to predict the level of difficulty for each game, by collecting data input to the toy and designed specifically for children to learn color and math. The details of the design are discussed later in this chapter; however, the size, shape and form of the toy are suited for children to physically engage and stimulate their interest (Figure 1.1). All the component parts are designed and developed specifically for the toy, including the 3D printed body and buttons, the electronics that control pressure sensors, an accelerometer, and LED lights, and the software for the math and color games. The physicality of playing and the cognitive stimulation in a game scenario offers a context that can dynamically respond to any player’s ability. Together these parts form a data collection system that takes advantage of AI technology to help children with autism learn.

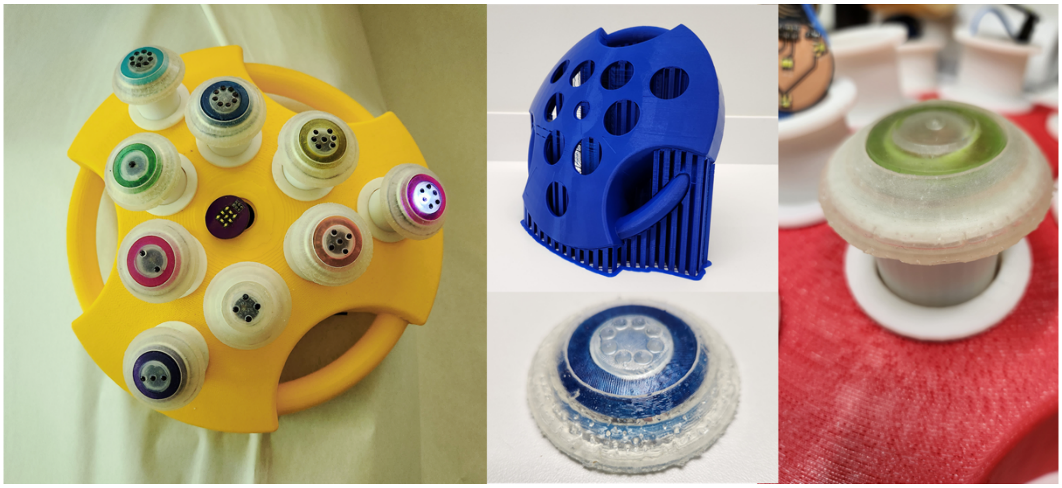


Figure 1.1 Educational Toy Prototype. Machine Learning version (left), 3D printed body and button (center), button electronics fitting in toy body (right).

Educational systems have long followed a one-size-fits-all model, driven by economic priorities rather than a qualitative assessment of student outcomes. The US Department of Education has tried several systemic changes, such as *the Common Core, No Child Left Behind*, and *Individuals with Disabilities Education Act,* to address quality standards and discrimination in schools. Still, these programs lack the specificity to effect change at the individual level. The result is that this issue has been left to special education teachers – depending on the teachers within the system and student access to them. Technology has a similar problem with access. According to a 2019-2020 report on the use of Educational Technology for Instruction in Public Schools (Gray and Lewis, 2021), only 45% of public schools in the United States provide computer use to each child, and only 23% of students were allowed to take home a computer in all or some of the grades between K – 12. For individualized or equitable tools to become more widespread, we need to reconsider a system of availability and what the tools are.

This project was inspired in part by the Reach Higher initiative started by First Lady Michelle Obama, which encouraged students to improve, particularly girls who were average in their education performance. This group of average students tended to get little notice because they were not gifted, nor did they struggle with their education. Similarly, this study focuses on a group in the middle of the autistic spectrum. Although there are participants in this study who are higher achieving students, the target audience centers on students who need sustained support in their education by addressing the specific needs of autistic learners. Characteristic of these needs are deficits in social relationships, communication impairments, repetitive behaviors, and restricted interests (Chamak, 2008 summary from the APA, Diagnostic and Statistical Manual, *DSM-IV* and WHO, International Classification of Diseases, *ICD-10*). This study partially addresses the last two characteristics, suggesting new actions to leverage repetitive behavior to learn new things and engage users by accommodating their engagement levels and customizing the learning experience.

Adopting a customization approach is one way to avoid the one-size-fits-all model, explicitly designing for the neurotypical population and leveraging the functionality of a broad-based platform like computers. With the advent of artificial intelligence, specifically machine learning, customization within specialized contexts can provide the closest solution to one-on-one education without hiring an army of specialty-trained teachers. This research takes the first step toward specialized tools that accommodate Autistic children's unique needs.

Learning with Autism

Institutional definitions (or psychiatric nosography) of Autism Spectrum Disorder (ASD) are provided by both the international and American classification of diseases (ICD-10, DSM-IV). Both sources agree that the significant domains for diagnosis of ASD are deficits in social relationships, communication impairments, repetitive behaviors, and restricted interests (Chamak, 2008). ASD has many symptoms, resulting in equally varied human behaviors. Many researchers describe the disorder as dealing with perceptual, cognitive, and social differences that are not transferred to other scenarios (Bosseler & Massaro, 2003; Cai et al. (2013), American Psychiatric Association, 2018) or knowledge is not generalized (Church et al., 2015). Ingersoll & Schreibman (2006) have suggested that imitating others may be a primary deficit in autism that underlies the abnormal development of social-communicative behaviors. However, deficiencies such as motor skills (Liu and Breslin, 2013) and language skill acquisition are reported to be shared among children with ASD (American Psychiatric Association, 2018). As a result, educating people with the disorder is challenging because of the variety of behaviors and the severity of the symptoms. Harnessing the potential of technology such as artificial intelligence (AI) is promising since it can adapt to human interactions (Cai, 2018).

Technologies have typically been used in education settings to deliver a consistent and tireless method of training that human educators have difficulty matching; however, until recently, computers have failed to adapt to changes in knowledge acquisition and pace of learning, especially in cases where predetermined lesson plans and skill testing are set based on grade level. Where it makes practical sense to build training tools relative to grade level, it rarely meets the needs of children who learn at their own pace. A challenge that also needs to be addressed by technology is the frustration that ASD children have with complex tasks (Anderson, 2016). Multimodal delivery and reframing information are not easy solutions because the students' behavioral responses can be unpredictable, and they need help transitioning from one subject to another. Suppose technologies become more flexible in adapting to behavior or sense a student's unwillingness to change behavior. In that case, there may be the opportunity to adjust the content and change the pace of transitioning from one context to another.

Building a Specialized Tool

A study of the Internet of Things (IoT) that enhances learning has demonstrated the benefits of touch-enabled devices not only in areas of knowledge related to exploring the physical world but also in the development of cognitive skills (Kennedy, 1992). The sense of touch provides a large amount of knowledge, often referred to as tacit knowledge or information known but difficult to name (see Reiner experiment (Riener, 1999). Also, the use of touch to manipulate objects is a kind of inverted manifestation of implicit understanding, transferring experience, curiosity, and emotional responses to the physical world being explored. This toy is embedded with sensors to measure to quantify these interactions particularly their physical manifestations, on learning. The prototype of the toy's electronic circuits allows data to be collected using pressure sensors in order to test the hypothesis that students externalize emotions that are conducive / unfavorable to learning through physical interactions. Although neuroscientists disagree about where tactile information is stored and how it relates to visual information, research shows that both tactile and visual inputs enhance the ability to recall information (Bara, 2004). Adding elements of physical interaction to visual information will make it possible to observe benefits in the memorization process as well as shorten the learning time needed by children with autistic disorders. Once the toy collects data from the user in a context, it is processed to predict the next question. In studies of neurodiversity, data is often collected and analyzed for a specific feature, or data that has been collected is processed or combined with other data to generate new insights. Here, the toy collects and processes data to develop a new hypothesis for learning which combines features not previously combined to see if they offer new insights. The following research questions arise from this hypothesis.

The Research Questions

The following research questions are derived from behavioral issues autistic children have with digital systems. They are specific questions about addressing these issues using the toy and the software that runs it. Moderate to highly functioning autistic children exhibit repetitive behaviors and compartmentalize learning within an activity context. The data collected from the players of the games are used to make predictions and analyzed for player performance and toy performance. The research questions draw the connection between the technology and the user's input.

Research Question 1: Can we capture patterns of learning using machine learning to understand and mitigate the problems surrounding personalized learning environments to provide a customized educational tool?

This question presupposes that there are definite patterns that can be trained and predicted to accommodate individuals. The patterns inform a reasonable prediction as to whether a student is ready for a more difficult, equally difficult, or less difficult question than the previous one provided.

Research Question 2: How does the tree-structure's pre-prediction approach compare with machine learning prediction of question difficulty?

Research Question 3: From the features selected (duration, pressure, motion, difficulty, and performance,) are these useful data points to use?

Research Question: 4 How well can ML predict suitable changes to game level difficulty? (i.e Toy Performance)

Research Question: 5 Can ML improve performance scores over tree structured programming?

To address this set of questions, I have designed and constructed three experimental prototypes: a paper prototype, a predetermined prediction prototype called the tree-structured prototype – based on the structure encoded in the programming – and a smart toy that uses machine learning to make predictions about the difficulty of the next question to ask.

Chapter 2 - Literature Review the Design of Toys

A designer's concerns with interactive objects encompass several fields, such as psychology, behavioral science, and anthropometrics, as well as others. These concerns affect a design's outcome and the user's experience, which is often more than just a ‘good’ or ‘bad’ one. Questions surrounding affordance, engagement, and ability connect the physical with the cognitive within these fields. The physical interaction with devices manifests cognitive reasoning, and conversely, the resulting physical feedback provides more information to be cognitively processed. At every stage of this, back and forth between the physical sensations and visual information provide clues to the ‘how’, leaving the ‘what’ and ‘what next’ to the user's mental capacity. Tying them closely together to recall the actions that produced the desired effect is what learning is about. A closer look at the specific needs and behaviors that favor a positive experience helps reinforce learning. Designing an educational toy for autistic children must address these concerns to ensure a positive experience and as a research tool. The toy should not interfere with data collection with a negative experience.

Perceived Affordance in Context

The concept of affordance comes from J.J Gibson’s work, which considers the perception of an object by the user. Gibson’s concept of affordance was introduced in the field of perceptual psychology (1979), which states that information is stored in the object, awaiting the viewer to perceive what is afforded. In Gibson’s definition (1979), “[t]he affordances of the environment are what it offers the animal, what it provides or furnishes, either for good or ill.” Because the user observes the object, there is a relationship with the object unique to the individual and the context in which the individual finds him or herself. For example, “[the ground] is not sink-into-able like a surface of water or a swamp, that is, not for heavy terrestrial animals. Support for water bugs is different. As an affordance of support for a species of animal, however, they have to be measured relative to the animal” (Gibson, 1979). Gibson’s theory of affordance includes the relationship only between the viewer and the object. Still, if we consider the designed world, we must also include the designer's intention to imbue the object with affordances. Affordance then forms a connected relationship categorized by Maier and Fadel (2008) as a complex adaptive system. Maier and Fadel’s designer-artifact-user (DAU) model (fig 2.1) is complex because of the nature of the human relationships we have with each other and with the designed world. How adaptive the affordances are depending on the designer’s ability and the context (which is constantly changing) (Maier and Fadel, 2008).

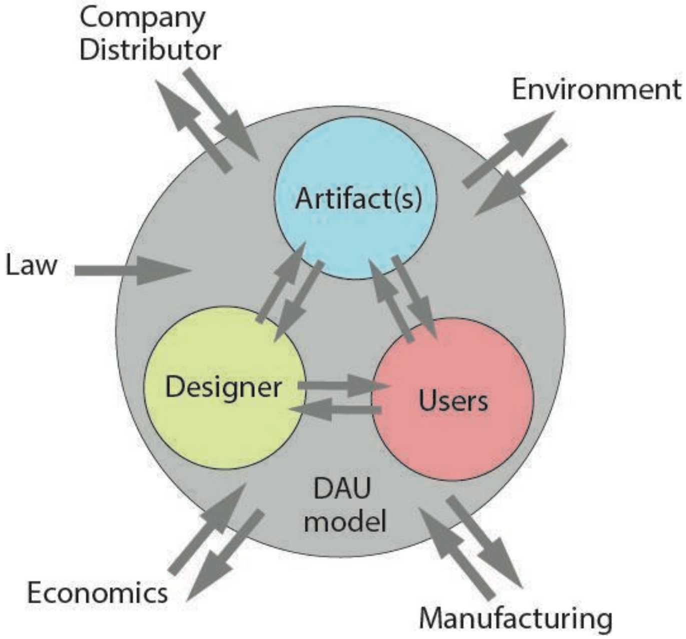


Figure 2.1 Recreated DAU model with permission of the author.

Maier and Fadel's model operate within a production system, but we could also consider the system in which an artifact is perceived and used. A toy, for example, would reside in a daycare or a child's bedroom. Someone could purchase it at a store or play with it after school, after a class the child likes, with a caregiver, or with a friend. In this context, we see not only the toy's functional affordances but also the relational affordances it can provide.

The concept of *access* also includes affordability and proximity, as well as other factors. However, if we assume access is possible, the student must still perceive these tools and the fact that their affordances are useful (Fogg, 2009). In the seminal design text *The Design of Everyday Things*, psychologist and design consultant Don Norman points out that "the real question is about the perceived affordance: Does the user perceive that clicking on that location is a meaningful, useful action to perform?" (Norman, 2015). It is not merely a question of whether an object affords the action. It is also a question of whether the perception of the action available accords with what the user is seeking. Perception of affordance is an ability that must be inherent in the individual or acquired through interaction in a context that allows for exploration (Fogg, 2009). What is possible, what can I try to do, what is the result of an action, and does it map up to the idea prior to the action? These questions frame a type of exploration that indicates engagement in the subject, a kind of curiosity that is the motivation for understanding information associated with the actions.

Learning occurs in the context of our constructed educational systems, helping children succeed. The context is not only the physical spaces and objects students engage with but also the style of teaching, appropriate content for the learner, and the social network that supports learning. Vartiainen et al. state, “the ways in which children’s thinking and sense of agency are related to the affordances and social situations of the moment“ [Schoultz, 2001]. The reason is that the way one understands and conceptualizes young children’s encounters with digital technologies depends more on the tools and sociocultural context” (Vartiainen, Tedre & Valtonen, 2020). Within this context, numerous affordances are presented to children to navigate and choose to participate in. According to Vartiainen et al., “children’s learning and action reside not only in individual abilities; it is also distributed across the tools and affordances that are within reach of children in the ‘‘zone of proximal development’’ (Vartiainen, et al., 2020). The zone of proximal development will be discussed at length in this document’s educational literature review; however, the educational system posits a child who has access to knowledge and is, in essence, provided an opportunity to learn. Engagement, ability, and affordance are linked to a chain of conditions needed to interact with the toy (Borman, 2014).

Engagement

We can formalize the discussion thus far by using the model for behavioral change developed by BJ Fogg (2009). The model identifies three characteristics that need to be present for behavior change to take place: abilities, motivation, and triggers. Although engagement is necessary to assess ability, levels of engagement do not indicate increases in ability. Nonetheless, the two states of mind are closely connected (Rowe et al., 2011). Rowe goes on to say that engagement can interfere with the process of improving skills. He concludes that engagement has a greater influence on the ability than the inverse (Rowe et al., 2011). Still, others, such as Filsecker and Kerres, make the “distinction between motivation on the one hand, which relates to the *building of intentions*, and engagement on the other hand as a volitional process that regulates the *enactment of intentions*” (Ajzen, 1985; Kuhl & Beckmann, 1985 cited by Filsecker and Kerres, 2014). Engagement lies somewhere between abilities and motivation, influencing the latter. Engagement is the precursor to motivation and the justification for measuring it.

Cognitive Engagement

For a user to be engaged, an interactive device must be aesthetically interesting and must stimulate interest through the possibilities it affords. We might consider these two features the toy's ability to engage users, but how do we *keep* players engaged? Filsecker and Kerres suggest that engagement is cyclical, stimulated by game challenges to re-engage players as they accomplish tasks and are rewarded, only to be given another task. It is a cycle reminiscent of ‘flow’ theory. Garris et al. (2002) proposed a model of gaming as “a cycle that entails users’ judgments, behaviors, and the [game] system feedback,” which are critical features of engagement. Filsecker and Kerres’ meta-analysis of engagement reveals multiple levels and conceptualizations, which they summarize as “a deployment of ‘energy in action.’” They identify distinctions between them characterized by three types of engagement: emotional, behavioral, and cognitive. In educational games, emotional and behavioral engagement have a greater impact on cognitive engagement (Filsecker and Kerres, 2014) than, for example, emotions have on behavior.

How does engagement get enacted, captured, and assessed? One dimension suggested by Filsecker and Kerres (2014) is the type of effort, arguing that “effort could be *covert* and represent an ‘effortful processing of information,’ or it can represent a more *overt* activity such as the time an individual is ‘working on a task.’” Covert efforts can be captured through actions and attention given to a game but are more difficult to assess. Engaging a player with the game is possible even though the information provided may not be new or, to the other extreme, poorly understood. In this case, the cognitive engagement of the information is more than that of gameplay or game action. Overt effort is easier to capture using timestamps or time logs, but the level of cognitive engagement will vary between players. Cognitive engagement is, therefore best captured through multiple data outputs to better assess its quality.

Behavioral Engagement

Behavioral engagement is encouraged by designers who follow a process like the ’funnel,’ which broadly entices an audience, provokes desire, and finally converts them to action. From this design perspective, Fikar et al. (2018) describe ‘Zielreaktion’ which is a preference for visually interesting objects of play:

> We define a Zielreaktion as a desirable reaction invoked in a child during a therapeutic session and caused by specific factors. For example, a range of exercises that we observed made children follow a specific object with their eyes. Here, tracking the object was the *Zielreaktion*, and the exercises drew on factors that motivated the child to show this desired behavior. (Fikar et al., 2018)

In their research, they indicate that a child perceives not only toys that offer interesting visual cues but also the affordances that indicate their function: “[s]uch factors included intentional but also unintentional affordances of a toy, for example, sensory cues or signifiers like a visually attractive cartoon provoking fixat[ion] with the eyes of the character” (Fikar et al., 2018). The signals of fixation afford interest in the child even when an inanimate object presents it. It is a principle used in button design for interfaces, often based on the physical appearance of a real button. Button design is limitless on digital platforms because of the ease with which colors can be changed and surface finishes added and modified. Shape, contour, and scale can all be created to encourage the primary function of pushing. Designers appropriate from the physical world because it is familiar with all the attribution that comes with the physical button. How hard we push, the distance needed to cause the signal from the button to perform the action, and the sound of button clicks contribute to the signaling that an illustration of a button holds all the functionality of a physical button. The perception of the function through signals describes the interpretations illustrated by the design. This simulation so closely describes the affordances of the toy that they become synonymous with the toy's capability. Engagement is more than the aesthetic qualities it implies. Early studies indicate that engagement consists of system feedback, user control (Brown & Cairns, 2004), attention, motivation (Chapman, 1999), and the ability of the system to challenge individuals at levels appropriate to their knowledge and skills (Skelly, Fries, Linnett, Nass, & Reeves, 1994).

Emotional Engagement

Filsecker and Kerres, like Fikar et al., suggest that engagement with objects is not just cognitive and behavioral but also emotional. A player's emotional state will significantly influence their level of interest, but this will be highly individual based on their particular experience. Joseph Kaye, describing the relationship between the experience users have and the emotion or aesthetics of an interface, argues that "factors such as fun, enjoyment, pleasure or aesthetics have an influence on the user. That is, the evaluation of the experience is associated with evaluating enjoyment, fun, pleasure, etc." (Kaye 2007). To understand this relationship more fully, we must include game mechanics and its influence on emotion—particularly 'fun.'

Engagement is cyclical in that it should be continually triggered and not a single expression of novelty. Hunicke et al. (2004) developed the Mechanics, Dynamics, and Aesthetics model in the context of games to evaluate this relationship: "[t]he model explains this relationship in which dynamics are the bridge between aesthetics and mechanics through game behavior; between player and designer" (Hunicke et al., 2004). In terms of mechanics, it is the nature of games to include some form of winning or accomplishment (a form of emotional motivation) that restarts the engagement cycle if not immediately achieved. As Hunicke et al. phrase it: "...if the player doesn't see a clear winning condition, or feels like they can't possibly win, the game is suddenly a lot less interesting" (2004).

The Dynamics of gameplay can be described in the following way: "[d]ynamics work to create aesthetic experiences. For example, *challenge* is created by things like time pressure . . . or supplying winning conditions that are more difficult to achieve" (Hunicke et al., 2004). Here, the term "aesthetics" is also used to describe fun or enjoyment from the game's challenges. The challenge derives from the mechanics that influence these emotional responses. The findings of Hunicke et al. (2004) highlight the importance of matching the MDA model to the audience's ability or what is within reason to believe is achievable.

Abilities of the User, and A System to Match

Similar games on the market attempt to adjust gameplay dynamically to a player's skills. At the start of the game "Pure", the player is not presented with the traditional selections of Easy, Medium, and Hard difficulty levels. Rather, the game dynamically adjusts the AI vehicles a player drives to suit the player's performance while they play. The game's developers were able to approximate a learning or performance curve by comparing the player's finishing position after their initial and subsequent play of sections of the game. While adjusting the system's balance, they set the difficulty curve to keep players engaged. Their engagement was evident when users replayed the game, attempting to improve their performance (McAllister, et al. 2015).

It should be noted that the adjustments happen only after the game is played initially and only readjust after subsequent games are played. Although described as 'dynamic' by McAllister et al., it does not continually adjust throughout gameplay. A person's 'ability' can be defined as an overall skill level or a player's ability 'at-the-moment' (ATM). If a game adjusts to a player ATM, it can affect motivation. Critical to our emotional relationship with games is the notion of matching skill with the level of challenge the game offers. This relationship between ability and motivation is described in Fogg's model of behavior. Where motivation may be high, a user's ability may be low, so they cannot perform the task. Conversely, a user's motivation may be low even though they have the ability to perform the task (Fogg, 2009).

Motivation

The feeling of wanting to do something is intrinsic, conjured within ourselves but often prompted by others or by things that pique our interest. As previously discussed, engagement results from motivation, and without its presence, learning becomes difficult. From a design perspective, motivating people to change their behavior often motivates design. Certain games are developed to exploit motivation through challenges, sensations and feelings, competition, and narrative that encourage gamers to play and elicit fun experiences (Hunicke et al. 2004; Lazzaro 2004; Sherry et al. 2006). BJ. Fogg describes motivation in relation to ability, which can fluctuate based on the value a user places on the benefit of continuing the interaction. The motivation may increase if the perceived value is high (Fogg, 2009). For example, reaching a new level in the game is a personal value and may be motivation enough to keep trying. A player's ability may need to be improved, manifesting in attempts to complete a stage or task within a game.

Takatalo et al. (2015) distinguish between what they describe as “presence” and “motivation” in their study on user experience (UX), in-game mechanics, narrative, and game interfaces. They found a clear difference between presence and the motivational concept of involvement. Involvement encompasses the user’s interest in the game or interaction (Takatalo et al., 2015). Often, this interest is triggered by aesthetic evaluation, as well as how relevant the user finds the content: “[i]nvolvement concerns the level of relevance based on inherent needs, values, and interests attached to that situation or an object” (Zaichkowsky, 1985). Thus, involvement determines the meaning and value of the UX. As Takatalo et al. maintain: “[t]he main interest here is not in what motivates gamers to play, but in understanding the meaning and personal relevance of the game” (Takatalo et al., 2015).

It should be said that presence is not the same as engagement but a precursor to it. We may have enough presence of mind to play a game but be easily distracted. It follows that presence – engagement – involvement operates as a continuum increasing in intensity as the perception of value increases (Takatalo et al., 2015). As a result, the motivation for playing increases. Classifying the continuum as *self-motivation* can be done when initiated and regulated by one’s own mind. The perception of games is there to disguise the educational component integrated into gameplay. Self-satisfaction, however, is not the only form of motivation. It may be more sustainable given its tight relationship to engagement, involvement, and interest in pleasing parents, teachers, and others. External challenges, like being part of a group, are also motivating.

A study that examined the impact of external motivations, including competition between peers, debunked the notion that it is harmful or unmotivating. Filsker et al. (2014) found that the negative consequences of competition may be “more indicative of impoverished learning environments and lack of feedback and opportunity to improve than of any fundamental consequence of competition.” In their study, external motivation came from badges earned during the game. The findings of Filsker et al. are also consistent with the notion of the informational value of external rewards (Deci et al., 1999). A study of extrinsic rewards by Birk et al.(2016) expanded these findings by including feedback rewards from the game, not external support or competitive individuals. People who provide rewards are perceived as an obligation, or in a competitive environment, as adversaries who may concede defeat (Birk et al. 2016).

From studies that use Cognitive Evaluation Theory and by way of the “overjustification” hypothesis, we see that external reward systems can diminish intrinsic motivation (Ryan et al. 1983). These theories propose mechanisms to explain how detrimental external rewards can be for motivation (see discussion of theoretical discussion and frameworks in Chapter 3). Simply stated, external reward systems are perceived in two parts: feedback (in the form of communication) and control (in the form of a condition). If performance results are acceptable, then the award is granted. The “overjustification” hypothesis, according to John Bates (1979), “predicts that intrinsic interest of individuals in an activity will be undermined by inducing them to engage in that activity as an explicit means to some extrinsic goal.” The nuance of competition from the game, not human competitors, is perceived as less controlling. This perception is likely because it is an inanimate object that one controls. The communication of feedback, which is meant as a positive reward, is an external motivation. Perceived as an inducement to achieve for someone other than oneself is likely to have an adverse effect. Feedback also supports the notion that it should come from the game, where the communication is objectively delivered.

Motivation is a complex and multi-directional human quality in the context of games and game mechanics. There is the potential to support both internal and external motivation. As Laine and Lindberg (2020) point out: “game mechanics that are meant to support motivators (e.g., points and badges for achievement) can produce different motivational results that depend on the context of use.” The motivators listed by Laine and Lindberg “can support either extrinsic or intrinsic engagement” (2020). The 14 motivations supported by educational games are Challenge, Competence, Control, Competition, Curiosity, Emotions, Fantasy, Feedback, Immersion, Novelty, Rules and Goals, Real World Relations, Social Interaction, and Utility. Although discussed thoroughly in their paper, their important contribution to game design is the connections between the motivations and design principles found in serious games.

Laine and Lindberg (2020) Developed 13 classifications from 54 design principles distributed across different educational games. These classifications align somewhat with the motivations previously listed. It is not their intention to advocate for adding all the design principles to every game; however, based on the context of the game, the following principles are found in many educational games: Challenge, Control, Creativity, Exploration, Fairness, Feedback, Goals, Learning, Profile and Ownership, Relevance and Relatedness, Resources and Economy, Social Play, Storytelling, and Fantasy. The toy design includes many of these principles, such as Challenge, Control, Fairness, Feedback, Goals, and Learning. The game's mechanics are direct, leaving little room for exploration or creativity because the context is for younger children learning math. Unlike more creative subjects such as writing stories, the emotional relationship to the toy is nuanced. Any feelings of joy or satisfaction are difficult to parse from the child's internal motivations and the toy's accommodation strategy.

Game Design and Emotions

It is difficult to discuss motivation without including emotional responses. Based on Ekman's analyses of facial expressions, a subset of emotions includes joy, anger, fear, surprise, disgust/contempt, and interest. The three former expressions are more commonly studied in their full representation of emotion, whereas the latter three are not given the same investigative attention. This is also the case in game design: "...some emotional experiences are well represented in the video game literature (e.g., anger and pleasure, fear and guilt)… there is very little extant work examining how video games impact basic emotions such as pride, interest, surprise, disgust, and sadness or complex, higher-order emotional experiences such as jealousy, embarrassment, scorn, or love" (Hemenover, 2018). However, there have been numerous studies on the impact of video games on emotion (Granic, 2014), One study suggests that "some of the most intense positive emotional experiences are triggered in the context of playing video games" (McGonigal, 2011). Still, many studies have considered games from a negative perspective and the harm they impose through emotions. This research focuses mainly on positive emotions, particularly their ability to bolster learning.

Observational studies have become common for evaluating the usability of designers' products and services. We have established the relationship between the designer and user as one that captures the user's desires, wants, and needs through their behavior as they interact with the product (Neilson, 2002). However, previous studies on emotion with respect to games are viewed through the theoretical lens of immersion (McMahan, 2003). Interest, joy, or immersion in the game is the objective of game developers and, therefore a primary focus (Takatalo, 2015). Observing the behaviors and recognizing a resulting emotion, such as joy or frustration from an interaction, is not always immediate, overt, or easy to identify (McAllister and White, 2015). Few studies quantify human behavior in space through expressions of the user's emotion (Chamak, 2008). In part because of the complexity of capturing these behaviors as a collection of data points and the volume of data needed to capture the behavior, has until now, been difficult.

Additionally, the more obvious expressions captured through resulting behavior are fraught with complexity and the potential for inaccurate assessment of a player's emotions (Chamak, 2008). Design has had to consider the Internet of Things equipped with sensors and micro-controllers that capture this type of data in real-time. Observational studies provide a snapshot of behavior that can corroborate an emotional response (Chamak, 2008). Comparing sensor data with the response can provide unique markers for the various emotional states of a player, as well as the data leading up to an emotional response being observed. Suppose observations identify an emotional event, and the corresponding data collected provide a unique marker for the feeling of joy. In this example, it may only be unique to that player or that context. Even if user experience researchers considered this emotional trajectory, it does not always follow in its continued direction. It is a relative theory of design that looks at patterns of behavior within a particular context, offering an opportunity to study user interaction based on what they perceive is afforded to them. User testing is a form of observational study that identifies issues in their context.

Triggers

Fogg (2009) defines 'triggers' as any precursor to a behavior, such as an alarm, a notification, an invitation, or a thought from within our own minds. Within visual communication, the trigger is the design, central to concepts enticing the viewer to consider what is being said. Fogg and Nass (1997) further describe triggers in the context of computer interaction using the rule of reciprocity. The computer interface encourages and affords the option to select, click, type, etc., to trigger a behavior. Their study examined whether the power of reciprocity, or 'do-unto-others,' is transferable from human/human interactions to human/computer interactions. (Fogg & Nass, 1997). More specifically, Gernsbacher (2006) reminds us of the definition "that reciprocity needs to be mutual and symmetrical—that reciprocity is a two-way street." He goes on to say that the mutual benefit may be missing from simple forms of imitation. In a reciprocating play study by Ingersoll and Schreibman (2006), children with autism did demonstrate an improvement in their ability to reciprocate: "...all children exhibited higher than baseline rates of total object imitation, suggesting the behavior was durable" (Ingersoll and Schreibman, 2006). This was true not just for identical or similar object interactions. The reciprocation was transferable to other scenarios or objects: "The children also continued to exhibit generalized responding during the probes sessions" (Ingersoll and Schreibman, 2006). The result indicated that reciprocal behavior was learned and, more surprisingly, transferred to a new context with a similar or the same object - two characteristic behaviors for many autistic children. Not all reciprocal behaviors had the same strong result. In the same study, Ingersoll and Schreibman (2006) also tested a form of reciprocity they labeled 'pretend play' where a child takes an action from another context (within 30 seconds) and applies it to an inanimate object, themselves, or an adult. The action is triggered somehow but then imitated toward a different target than the trigger originator (Ingersoll & Schreibman, 2006). Here, the context of the action changes, but the action is repeated. Probes for generalization proved less sticky and only showed slight improvement from the baseline to the follow-up stage. This stickiness could be due to the mutual benefit described earlier by Gernsbacher.

Gernsbacher also suggests that when an autistic child initiates imitation and the adult repeats the behavior, the child is more attentive. Although attention or engagement is important, the educational value of an autistic child's reciprocating behavior may be triggered if they are allowed to start the interaction (Gernsbacher, 2006). Still, other possible correlations Ingersoll & Schreibman did not examine were imitation of actions in different contexts using only the same object because they were not using objects that could change or transform in some way. They concede the study only used triggers initiated by therapists or parents rather than computers or even other children. This approach lacks a naturalized imitation behavior, particularly if a child is imitating their peers rather than adults. The study offered no speculation on the use of imitating a digital object.

Haptics and Other Design Considerations

The feeling of five-ness – social and cultural organization of pips on dice or domino tiles.

A design's visual and tactile qualities are central considerations for communication between the interface and the user (Seminara, 2019). Most importantly, developing interactive tools for education continues to move online or to devices connected to the internet (IoT). Children with autism may benefit from a computer's persistence to continually explain lessons. However, it may also further feelings of frustration if communication targets the student's comprehension level. Haptic interaction that provides physical relationships to the information and visual cues supports learning more fully (Mongue & James, 2006). Relating the visual and physical representations to cultural similes like dice or domino tiles also triggers memory of these game devices (Seminara, 2019) that carry over onto the toy. Studies suggest it is not limited to a cognitive process alone, embodied in the interaction with the physical environment, aiding in memorization. "Active touch involves the concomitant excitation of receptors in the joints and tendons along with new and changing patterns in the skin. Moreover, when the hand is feeling an object, the movement or angle of each joint from the first phalanx of each finger up to the shoulder and the backbone makes its contribution" (Gibson, 1962). The physical positioning that leads to the success of a task is stored in goal memory. Our motivation to achieve goals prefers these memories over failures. Gibson continues, "These inputs occur relative to a continuous input from the vestibular organs, along with the cutaneous input from contact with the ground. Presumably, the feeling of an object by the hand involves the feeling of the position of the fingers, hand, arm, body, and even the head relative to gravity, all being integrated in some hierarchy of positional information." (Gibson, 1962). This framework associates the body's physicality in space with memory reinforced by positive feedback, such as achieving the goal. It is a framework supported by other researchers in constructivist learning and embodiment. (Papert, 1980; Lundgren et al., 2016)

As neurotypical children learn math, they often favor one of three strategies for solving addition problems (Calik and Kargin, 2010), depending on their numeracy skills. The first of these strategies is 'count-all', which uses fingers or other objects, each starting at 1 until all the numbers of a math problem have been counted. For example, 4 + 5 will initiate counting on one hand, four fingers, followed by the second hand of five. The second strategy is the count-on strategy, which involves saying the first addend of the addition problem and then counting from that number (Carpenter & Moser, 1984; Secada, Fuson, & Hall, 1983). The final strategy identified by Carpenter and Moser (1984) involves store-and-retrieve once the child has memorized addition facts from which to draw numbers and totals. One way to avoid any sigma of counting using your fingers is using a dot notation method, whereby dots are associated with each number from 1 to 9 according to a specified pattern. Referred to as "touch math," this method involves visual, auditory, and tactile learning. The students mark the pips in their own configurations while looking at a reference number (the symbol or verbal description) and counting the number (auditory) with their pencils or moving pips with their finger (tactile) (Calik and Kargin, 2010). The pips are then assigned to the symbol of the number for the children to learn each representation (Figure 2.2).

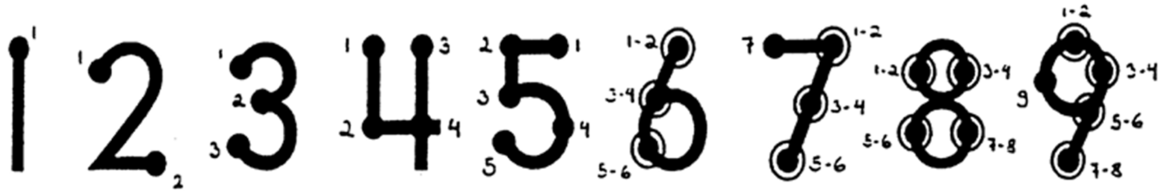


Figure 2.2 Touch math method for simple math equations. Numbers 1 – 5 single touch points and 6 – 9 single and double touch points.

Children are encouraged to vocalize the counting to help them memorize each number using both the count-all and count-on technique (Calik and Kargin, 2010). The cognitive process that reinforces memorizing numbers and equations using touch math method successfully teaches children with disabilities. (Pupo, 1994; Newman, 1994; and Calik and Kargin, 2010). The drawback is memorizing when to use single and double touch points on the numeric symbols. Learning these patterns took 10 to 20 sessions in the Calik and Kargin study. However, how long these sessions lasted and what material was covered is unclear. In any case, a combination of visual, auditory, and tactile learning has some effect on teaching children with cognitive disabilities.

Conclusion

There is a great deal of literature on education and there have been a number of education initiatives that propose a new way of teaching, but from this literature there are some common areas of interest; engagement, triggers or novelty, motivation, and presence of mind. All of these areas are contributing factors in educating children however, these studies often deal with neuro-typical children that assess learning centered around performance. It is not that performance is not important but when it is central to education, measuring it becomes narrow. Teaching autistic children adds to the complexity but also is an opportunity to think broadly about how to measure it when learning occurs.

A Review of Autistic Characteristics

Introduction

One of the most challenging problems with autism is the number and diversity of impairments expressed in children with the disorder. They range from physical difficulty to cognitive impairment to missing social and emotional cues. According to Yirmiya and Sigman (1991), "[r]egardless of intellectual level, autistic individuals reveal deficits in conceptual problem solving, meta representational ability, pragmatic aspects of communication, joint attention, symbolic play, and recognition of emotions." A meta-analysis by Greene et al. (1995) found individual characteristics of autism in which "somewhere between one-half to four-fifths of samples of autistic children fit the 'typical' profile or deficit." However, no one is suggesting that all autistic children possess every trait nor to the same degree. What is clear is that children with autism have a variety of these particular challenges. An adaptive educational device can best address each student's manifestation of learning difficulty when it comes to learning. In this study, a device is designed to assuage the manifestation by supporting cognitive activities, communicating clearly, and supporting social behaviors in an object consistent in its response to the child's interaction. Previous studies that applied machine learning to address some of these difficulties include the following approaches and results. (Schopler & Mesibov, 1995)

Difficulty with language and communication: Many children with autism have difficulty with language and communication, which impacts their ability to understand and express themselves. This problem may include difficulty with verbal communication, nonverbal communication, and social language skills such as taking turns in conversation.

Difficulty with social skills: Children with autism may struggle with social skills, such as initiating and maintaining social interactions, recognizing and responding to social cues, and engaging in cooperative play with peers. These difficulties can make it challenging for them to learn in social settings and to develop social relationships.

Repetitive behaviors: Children with autism may engage in repetitive behaviors, such as hand-flapping or pacing, which can disrupt learning.

Difficulty with attention and executive function: Children with autism may have trouble with attention and executive function, impacting their ability to focus, plan, and organize their learning.

Difficulty with sensory processing: Children with autism may have trouble processing sensory information, such as sights, sounds, and textures, impacting their ability to learn and focus.

From this list, few studies attempt to consider the complex issue of coalesces of problems that impact learning. Although isolating the response to an impairment may indicate its importance and influence on the learner, it does not always address the real-world learning experience. This study also does not address all the impairments or positions on the spectrum they occupy. Instead, the broader categorical definitions described by Chamak (2008) offer insight into the characteristics of autism (social, behavioral, communication, and motivational) and how to address the interrelated nature of these issues. To begin, a brief look at some common characteristics impeding learning and therapies that address the problem provides us with a framework.

Learning-Related Characteristics of Autism

Mimicry

Current therapies to help children with autism focus on social behaviors that are tied to cognitive processes and influence learning strategies. One of these learning strategies is mimicry or social learning theory, popularized by Albert Bandura (1971). It is the skill of observing others in social interactions and following their lead to learn social norms, relationship interactions, and learning skills and information common to the group. A characteristic of autism is, for many children, recognizing social interaction as a source of learning and using their observations to adapt to similar experiences (Forbes et al., 2016). The four stages of the theory are attention, retention, initiation, and motivation, each centering on observation of not only the activity of others but also the reaction and outcome of the action to evaluate its effectiveness for our use. Social learning comes naturally to most of us; however, a concerted effort from autistic children is needed to train their minds to practice social skills, accompanied by further explanation of future outcomes to identify negative repercussions (Forbes et al., 2016). Using these social observations in future contexts is also difficult and requires repetitive demonstrations along with applications to new contexts to explain their versatility. An inability to implement this strategy places autistic students at a significant disadvantage (Keay-Bright & Howarth, 2012), removing a large set of references for learning.

Cognitive Flexibility

Cognitive flexibility, categorized within executive function, is discussed further in the theory section (Chapter 3); however, it is worth noting here because of the toy's implications and function in this study. The ability to shape our thoughts later used in different contexts is a deficit of many autistic children and impedes learning (Goldman-Rakic, 1987). Rather than use the mental models to "access and hold mental representations" to shape the new contexts, "autistic children might instead rely on cues in the external environment to guide their behavior" (Goldman-Rakic, 1987). It is not for lack of attention or retention of information; it is a chosen strategy (Klin et al. 2007). ASD individuals often study the details in situations, which is an advantage of knowledge that requires featural, fragmented, or rote learning called Circumscribed Learning (Klin et al. 2007), discussed in the education lit review. It is essential to note the subtlety of knowing and applying knowledge appropriately.

Behavioral Control

Both mimicry and cognitive flexibility assume a certain amount of behavioral control. Children learn the concept of controlling behavior at an early age by experiencing the benefits and consequences of the associated behavior. Butera and Haywood (1995) suggest behavioral control is a necessity for learning to take place: "[m]any of these cognitive functions are in a sense precognitive and prelinguistic in that they are necessary prerequisites to language learning and conscious planned thinking. It is at precisely this level of functioning that children with autism commonly begin to experience cognitive processing deficits." Behaviors stem from the internalization of one's behavior and an external view of social behaviors that govern the selection of our own behavior, "thinking about one's own behavior, listening or looking carefully and gathering clear and complete information, accommodating to changes, learning and using rules, role taking, and other basic cognitive processes that allow children to function effectively in social settings." (Butera and Haywood, 1995) It is essential to note the central role behavior plays in mimicry (a social behavior) and cognitive flexibility (a cognitive behavior). To that end, it deserves a closer look and an overview of common therapies for autistic behaviors.

The Role of Social Behavior

Applied Behavioral Analysis

Educating children with autism typically involves a range of interventions designed to support their social, communication, and behavioral development. One typical therapy is applied behavior analysis (ABA). ABA is a type of therapy that uses principles of learning and behavior to help children with autism develop new skills and reduce challenging behaviors. ABA can involve one-on-one therapy sessions and structured activities incorporating reinforcement and other behavioral principles. According to Autism Speaks an advocacy group for autistic children, applied behavior analysis is interpersonal therapy where a child works with a practitioner one-on-one. The goal of applied behavior analysis is to improve social skills by using interventions based on learning theory principles. More empirical studies show that 47 percent of children in the high-intensity treatment achieved an average-level IQ score and succeeded in general education classrooms without additional support compared to only 2 percent of those receiving low-intensity treatment (Linstead et al., 2015). Subsequent studies evaluating the effectiveness of ABA treatment for ASD have shown similar results (Alberto et al., 2006; Cooper et al., 2007; Jones et al., 2014).

ABA therapy helps children on the autism spectrum by:

Increasing their social abilities like completing tasks, communicating, and learning new skills

Implementing maintenance behaviors like self-control and self-regulation

Teaching them to transfer learned behaviors to new environments

Modifying the learning environment to challenge them in specific scenarios

Reducing negative behaviors like self-harm

Applied behavior analysis (ABA) is a type of therapy that uses principles of learning and behavior to help children with autism develop new skills and reduce challenging behaviors. ABA can involve one-on-one therapy sessions and structured activities incorporating reinforcement and other behavioral principles.

ABA therapists use a variety of techniques to teach new skills to children with autism, including:

1. Positive reinforcement: Positive reinforcement involves reinforcing desired behaviors with rewards or other positive consequences. This can include praising a child for completing a task, giving them a sticker, or providing them with a preferred activity.
2. Shaping: Shaping involves reinforcing successive approximations of a desired behavior. For example, a therapist might reinforce a child for making eye contact, then gradually reinforce the child for making longer periods of eye contact until the child consistently makes eye contact.

To shape behavior, a therapist might start by reinforcing a child for any behavior similar to the desired behavior and gradually reinforce more specific approximations of the behavior. For example, a therapist might reinforce a child for making eye contact, then gradually reinforce the child for making longer periods of eye contact until the child consistently makes eye contact.

Shaping can be a beneficial technique for children with autism, as it allows them to learn new skills in small, manageable steps. It can also be an effective way to teach skills that may be challenging for children with autism, such as social or communication skills, as it allows them to build on their existing abilities and gradually develop more complex skills.

3. Modeling: Modeling involves demonstrating a desired behavior and then reinforcing the child's imitability of that behavior. Video modeling can involve the child watching a video of someone demonstrating the desired behavior or live modeling, in which the therapist demonstrates the behavior in person.

Video modeling can be especially useful for children with autism. It allows them to watch the behavior multiple times and at their own pace, making it easier for them to learn.
In addition to reinforcing the child to imitate the modeled behavior, it is also important to provide feedback and reinforcement for any progress or improvement the child makes. Modeling behavior can build the child's confidence and encourage them to continue practicing the new skill.

ABA therapists may also use techniques to reduce challenging behaviors, such as:

1. Positive punishment involves reinforcing behavior by adding an unpleasant consequence. For example, a therapist might remove a preferred activity from a child who exhibits challenging behavior.
2. Differential reinforcement involves reinforcing an alternative behavior incompatible with the challenging behavior. For example, a therapist might reinforce a child for sitting quietly instead of exhibiting challenging behavior such as screaming.

Overall, ABA can be a highly effective intervention for helping autistic children develop new skills and reduce challenging behaviors. To ensure the most effective approach, it is important to carefully tailor the interventions to each child's specific needs and abilities and work closely with a trained ABA therapist.

Cognitive Behavior Therapy

Other behavioral therapies include cognitive behavior therapy (CBT), which focuses on helping autistic children develop strategies for managing their emotions and behaviors. This therapy can involve helping children identify and change negative thought patterns, as well as teaching them skills such as relaxation techniques and problem-solving (Hofmann, 2010; Reaven, 2012). CBT therapists typically work with children to identify negative thought patterns, such as negative self-talk or automatic thoughts, and help them to challenge and replace these thoughts with more realistic or positive ones. For example, a child might be taught to replace the negative emotions they learn to recognize with positive strategies for overcoming the problem and reinforcement statements (Reaven, 2012).

CBT therapists may also teach autistic children relaxation techniques, such as deep breathing or progressive muscle relaxation, to help them manage their emotions and reduce stress. These techniques can be particularly useful for children with autism who may struggle with regulating their emotions or managing anxiety. In addition, CBT therapists may teach children problem-solving skills, such as identifying problems, generating potential solutions, and evaluating each solution's pros and cons. These skills can help autistic children more effectively manage challenging situations and behaviors. One of the more important roles of a CBT therapist is to ensure the most effective approach and tailor the interventions to each child's specific needs and abilities.

Neuroscientists have mapped brain activity to identify generalizable patterns of thought during different activities and non-active states such as rest. Wang et al. (2018) have mapped these patterns verifying the brain activity using self-reporting techniques, and coincidentally used machine learning to categorize the thought processes into four human experiences:

> We used machine learning to determine patterns of association between the neural and self-reported data, finding variation along four dimensions. ‘Purposeful’ experiences were associated with lower connectivity - in particular, default mode and limbic networks were less correlated with attention and sensorimotor networks. ‘Emotional’ experiences were associated with higher connectivity, especially between limbic and ventral attention networks. Experiences focused on themes of ‘personal importance’ were associated with reduced functional connectivity within attention and control systems. Finally, visual experiences were associated with stronger connectivity between visual and other networks, particularly the limbic system. Some of these patterns had contrasting links with cognitive function as assessed in a separate laboratory session.

Although brain mapping is outside the scope of this study, it does indicate that there are broad patterns of thinking that are more or less associative to sensorimotor interactions, attention, and the limbic system, which controls behavior and emotions.

This study's intended audience is children who are mid- to high-functioning autistic and have the capacity to learn if they are provided the support they need (Keay-Bright & Howarth, 2010). Studies that use other technologies for children with autism aid learning in these specific ways.

Repetitive Behavior

Education and treatment for children with autism often involve interventions to address repetitive behaviors, which can be a common characteristic of autism. These behaviors include repetitive body movements (e.g., hand-flapping, rocking), repetitive vocalizations, and repetitive interests or routines. Repetitive behaviors can interfere with a child's ability to learn and participate in daily activities. They can be a source of frustration and stress for the child and their caregivers. A more recent study used machine learning to predict the likelihood of repetitive behaviors in children with autism and found that the model could accurately predict these behaviors (Muller et al., 2021).

One of the most challenging aspects of learning, and more fundamentally, is engagement in the subject. Communications research has often struggled with measuring engagement typically because a viewer could be reading text but thinking of something else. Or they could interact with an interface without considering the content while the interactions occur. With skills of affect perception, a computer that detects the learner making a mistake while appearing curious and engaged could leave the learner alone since mistakes can be important for facilitating learning and exploration; however, if the learner is frowning, fidgeting, and looking around while making the same mistake, then the computer might use this affective feedback to encourage a different strategy.

Interventions for repetitive behaviors in children with autism may include:

1. Applied behavior analysis (ABA): ABA is a type of therapy that uses principles of learning and behavior to help children with autism develop new skills and reduce challenging behaviors. ABA can involve one-on-one therapy sessions and structured activities incorporating reinforcement and other behavioral principles.
2. Cognitive behavior therapy (CBT): CBT is a type of therapy that focuses on helping children with autism develop strategies for managing their emotions and behaviors. This can involve helping children identify and change negative thought patterns, as well as teaching them skills such as relaxation techniques and problem-solving.
3. Medication: In some cases, medication may be used to reduce repetitive behaviors in children with autism. Antidepressants and antipsychotics are the most commonly used medications for this purpose, although their effectiveness varies, and they can have side effects.

In terms of the analysis of repetitive behaviors, machine learning techniques have been used to identify patterns in the repetitive behaviors of children with autism. For example, one study used machine learning to predict the likelihood of repetitive behaviors in children with autism and found that the model was able to accurately predict these behaviors with high accuracy

(Muller et al., 2021). Another study used machine learning to identify patterns in the language use of children with autism, which could be used to predict language outcomes and inform language interventions (Bhat et al., 2016). These types of analyses can provide valuable insights into the repetitive behaviors of children with autism and can inform the design of interventions to support their development.

In contrast to these therapist-led approaches, Pivotal Response Treatment (PRT) is a child-led therapy that allows children to explore social and cognitive activities through the use of games: "[t]he child-led gameplay and toy used in PRT have been widely recognized as being beneficial for children's cognitive and social growth (Jarvis et al., 2014; Piaget & Inhelder, 2008). The traditional Montessori physical, analog toy is an example of such devices used in PRT's child-led gameplay" (Jahakbar et al. 2023). Often, the games are not specifically designed for autistic children and typically trigger other unwanted behaviors or limit progress in some cognitive skills. Still, these games demonstrate the potential for games used in therapy as a way to stimulate a child's intellectual growth.

Conclusion

As discussed in the literature review of design, behavior plays a role in how people interact with objects and what they perceive the objects afford. Much of autistic literature deals with behavioral therapies and how to mitigate and provide tools for the child to manage behavior. Although some games attempt to accommodate for the user it is pre-determined and made broad enough to appeal to all players. These behaviors are clearly triggered by cognitive action, triggered by motivations. Autism throttles these behavioral and cognitive interactions to a greater or lesser degree. It seems flipping the interaction so that the game adjusts to the player in real time is an approach now possible with machines that predicts the next interaction.

The Process of Learning, Beginning with Sensing

There is a wealth of information supporting approaches to how we learn as human beings (McGeoch, 1942; Jarvis, 2012; Tall, 2013; Illeris, 2018). As early as infancy, learning is often described in terms of input senses of the human body, most prominently our eyes, to observe the information but also through the senses of hearing, touch, as well as taste and smell. The information acquired by the senses is then transposed into our brains in a mental model of categorization and processes for retrieval, what Piaget coined as organizational action schemes (Piaget, 1954). This develops slowly in children, indicated by Berger and Hatwell's (1995) study on haptic education, which asked children between the ages of 5 and 9 to classify 16 cubes that varied in hardness and texture density: "[u]sing only touch, [children] each explored a cube and were then asked to choose which of three comparison objects ‘goes better with’ the original cube” (Berger and Hatwell, 1995). Their findings suggest that children use hardness to describe each individual cube, indicating their preference for dimensional qualities rather than the more subtle differences that texture provides. In adults, on the other hand, texture was used most often to categorize the cubes, which is a nuanced tactile quality. Berger and Hatwell also found that children localized their focus on individual cubes as opposed to a global approach taken by adults, who considered the categorization of all the cubes (Berger and Hatwell, 1995). Additionally, they observed, “differences may be due to the sequential nature of haptic processing. That is to say, in young children haptic exploration is not yet systematically organized” (Berger and Hatwell, 1995). The physical nature of learning in younger children is more important to establish a fundamental knowledge to navigate the world at this stage more central to their understanding.

Knowledge acquisition is a critical aspect of learning, but other processes are being developed even at early stages of a child’s development. As we learn language, for example, cultural meaning is applied to entities in the categories, what Illeris describes as “mediated experiences” (Illeris, 2018). The mediated experience relegates the senses in favor of cognitive processes to be central to learning (Illeris, 2018). The most widely known framework to organize the mediated experiences in our brains is Bloom’s taxonomy and what has later been referred to as the “revised taxonomy,” after changes made by Anderson et al. in 2001 (Anderson et al, 2001). The original taxonomy is a progression of five levels of understanding from knowledge, comprehension, application, analysis, synthesis, and evaluation (Krathwohl, 2010), with the latter four characterizing mediated experiences. This is to say that both knowledge acquisition and comprehension require cognitive energy but are triggered primarily by sense stimulation.

Cognitive Process of Revised Taxonomy

1. Remember (formerly part of Knowledge) – Retrieving relevant knowledge from long-term memory.
2. Understand (formerly Comprehend) – Determining the meaning of instructional messages, including oral, written, and graphic communication.
3. Apply – Carrying out or using a procedure in a given situation.
4. Analyze – Breaking material into its constituent parts and detecting how the parts relate to one another and an overall structure or purpose.
5. Evaluate – Making judgments based on criteria and standards.
6. Create (formerly synthesis and moved to the last process stage) – Putting elements together to form a novel, coherent whole or make an original product.

(Krathwohl, 2010)

Fowler and Mayes (2004) argue that the requirements for learning to take place are best described through generic learning activities outlined by Bloom's (1956) taxonomy. This approach is not dissimilar to using the revised and updated version of Bloom's (Anderson et al, 2001). Anderson et al clarify the taxonomy table by mapping the cognitive process dimension of creating, understanding, analyzing, applying, and remembering onto a knowledge dimension (see table X: metacognitive, procedural, conceptual, and factual). This is done with the understanding that sense acquisition or fundamental knowledge is a different process than the procedural forms of learning that distinguish the two taxonomies: "[t]he major differences lie in the more useful and comprehensive additions of how the taxonomy intersects and acts upon different types and levels of knowledge" (Anderson et al, 2001). The intersection of the cognitive process with the knowledge process is a point where learning objectives can be identified (Table X). Most importantly, Anderson et. al. concede the knowledge dimension of metacognitive was included to cover cognitive knowledge expanding the category previously dominated by the sensor receptor-focused knowledge. In Bloom's original taxonomy, knowledge of universals and abstractions (conceptual in the revised taxonomy) deals with the transformation of sensory knowledge to cognitive understanding such as principles and theories (Krathwohl, 2010). The metacognitive category covers strategic knowledge, knowledge of cognitive processes, and self-knowledge. (Table 2.1)

Table 2.1: Revised Bloom's Taxonomy of Learning, 2001

		The Knowledge Dimension			
		Factual	Conceptual	Procedural	Metacognitive
Cognitive Dimension	Remember	List	Recognize	Recall	Identify
	Understand	Summarize	Classify	Clarify	Predict
	Apply	Respond	Provide	Carry out	Use
	Analyze	Select	Differentiate	Integrate	Deconstruct
	Evaluate	Check	Determine	Judge	Reflect
	Create	Generate	Assemble	Design	Create

The revised taxonomy groups the cognitive processes, which changes how we think of the learning process and emphasizes cognitive processes specifically. However, more sophisticated uses of combined sensing occur while learning through what Vartiainen et. al. (2020) describe as apprentice-style relationships when participating in everyday activities with our families and communities. Rogoff's (1990) notion of guided participation stresses the active role of children in observing and participating in organized societal activity. Such interaction between children and their caregivers involves the use of various kinds of cultural tools and semiotic signs adapted to the specific activity at hand, including tacit forms of communication (Rogoff, 1990 and 1993). In guided participation, the parents informally teach their children, who gradually gain an understanding of various tools, artifacts, and discourses that are integral to their everyday family and by extension community life (Rogoff, 1990). As children's understanding of and skills in using cultural tools grow, they can take more and more responsibility from the hands of their caregivers and do things independently (Rogoff, 1990, Wertsch, 2007). What Vartiainen and Rogoff suggest is that our senses, culture, and cognitive activities are intertwined in the learning process, and one or more are favored depending on the abilities of the learner. With a greater distinction between cognitive process and content, Gentry et al. indicate that process addresses how one solves problems, while content consists of the knowledge required to make these processes meaningful. (Gentry, Stoltman, and Curtis 1992). This would suggest that individuals who favor physical interactions learn from objects, tools, and other tangible things.

Measuring the physicality of an interaction could hint at the stored knowledge in the object. For example, Picard et. al. (2004) focused their study on "a child's interest level in natural learning situations, using a combination of information from chair pressure patterns sensed using Tekscan pressure arrays (recording how human postures change during learning)." Here the intention is to measure 'subject interest' based on movement and posture during study times. Haptic information from the Tekscan is closely related to knowledge acquisition because it informs how much or little attention is being paid to the task at hand if a person is moving on a chair or not, and is therefore potentially an important data point to be measured. Picard et al. report their system identified what they term affective behaviours, achieved an accuracy of 76 percent on affect category recognition from chair pressure patterns, and 88 percent on nine 'basic' postures. This increased to 82 percent and 98 percent, respectively, on children who have had portions of their data included as part of the training process (Picard et. al., 2004). In addition to the Tekscan, MIT Media Lab has worked on a computer mouse that also measures pressure but is tied more specifically to the emotion of frustration:

[W]e have been exploring uses of a newly developed 'pressure mouse' device, a mouse augmented with eight pressure pads that indicate 'how' the mouse is being handled. An increase in physical pressure applied to a pressure-sensitive mouse has recently been shown to be associated with the frustration caused by poor usability in a computer interface. (Hernandez et. al. 2014).

Although the study measures the level of stress, Hernandez et al. look at the affective behaviors as they relate to pressure on a mouse. Ass stress is increased, greater pressure is applied to the mouse. This is only one emotional condition being controlled and where other emotions may produce similar results, the important outcome is the effect on performance. Neither Hernandez et al nor Picard et al measure for this important condition.

The Physical is Experiential, which is Socially Constructed

In Raskin's psychology paper, Psychology: Personal Construct Psychology, Radical Constructivism, and Social Constructionism (2002), he approaches constructivism as a social interaction. Raskin takes the epistemological view of knowledge transitioning from 'what' we learn to 'how' we learn, a characteristic of the "postmodern/constructivist era [which] stresses the viability, as opposed to the validity of knowledge" (Raskin, 2002). The constructivist educational theory posits; that student's "discover and transform information, check new information against old, and revise rules when they no longer apply. This constructivist view of learning considers the learner as an active agent in the process of knowledge acquisition" (Dewey (1929), Bruner (1961), Vygotsky (1962), and Piaget (1980)" (cited from Bada, 2015). Discussed in more detail in the theory section, however, Raskin's quotation from Sexton (1997) provides an operational summary: "[t]he perspective of the observer and the object of observation are inseparable; the nature of meaning is relative; phenomena are context-based; and the process of knowledge and understanding is social, inductive, hermeneutical, and qualitative." (Sexton, 1997) Sexton is describing a more fluid conception of learning which is dynamic and tied to the experience of knowledge acquisition rather than the accumulation of factoids. How then do we measure experience to assess an individual's knowledge experience more fully?

Macintosh, Gentry, and Stoltman (1993) draw on the work of Wagner and Sternberg (1985 & 1987) to speculate that the effectiveness of experiential exercises can only be measured accurately when the measures include tacit learning. Wagner and Sternberg define tacit knowledge as that which is not typically expressed or stated openly, not directly taught, or spoken about. It can be taught, but the "stuff is typically disorganized, informal, and relatively inaccessible, making it ill-suited for conventional methods of formal instruction" (Wagner and Sternberg 1985). Although this study does not attempt to measure tacit knowledge, it is worth noting the shape of experiential knowledge and how it is distinguished from other processes such as Bloom or the revision. It also provides a comparison and evaluation relative to constructivist models previously discussed (Delay, 1996). There are subtle distinctions between constructivism and experiential learning theories, but mainly the focus on the objectified versus embodied (Delay, 1996). Delay further compares constructivism to rationalism where experiential educators rely on feelings and sensations (Delay, 1996). In an effort to formalize experiential learning, Kirkpatrick and Kirkpatrick (2015) updated their model developed by Kirkpatrick in 1998, which evaluates the effectiveness of learning according to levels of experience: level (1) was the material relevant and engaging; (2) did the student acquire knowledge, skills, attitudes, confidence, and commitment; (3) can the student apply what was learned; and (4) the extent to which results occur and are improved. Throughout our pedagogical history framing and assessing the structures that underpin learning, an emphasis is placed on three areas of educational research: acquisition, experience, and application.

ASD students have limits to their participation in this social aspect of learning simply because of their neurological makeup, as well as being ostracized by their peers in social settings. Inevitably this shapes how they learn and limits the process. The value of a social equivalent teaching aid cannot be understated, as technology has a larger capacity than humans to provide knowledge. One example is the inclusive nature of the toy, developed using a data set collected from other autistic children to reflect a shared experience of learning. One could argue that it is the collective human contribution that educates ASD students, which is an ethical, moral, or philosophical position. The position posited here is the combination of the rational constructivist approach and the experiential learning model that focuses on the sensational aspect of learning. Couched in the experience of a fun game, the structure of learning inherent in an educational activity supported by human experiences in the form of data distinguishes this research from others. Without technology processing the human data the experience would be uniform in shape and usable to ASD students who experience it. Conversely, even though the technology was developed with ASD students in mind the technology delivers human experiences, albeit a computational aggregate experience.

Active Learning

Instead of strict adult control or acquisition-oriented instructions, participatory learning also emphasizes children's active contribution in shared meaning-making and endeavors (Hedges and Cullen, 2003) This highlights the creation of environments that provide children with opportunities to explore real-world phenomena in an interest-driven and inquiry-oriented manner (Bulunuz, 2013, Vartiainen et. al, 2018). The children should be able to connect their interests, previous knowledge, and experiences to the learning situation, and have the opportunity to increasingly explain, interpret, and share their observations with their peers and teachers (Vartiainen et. al, 2018, Johnston, 2009). The more skilled partner may provide encouragement, means, and metacognitive support adjusted to children's interests and skills, within their dynamic zone of proximal development (Rogoff, 1990). In essence, the key issue in participatory learning is to create positive experiences that promote children's sense of agency or the feeling of being the author of one's actions in the world (Hilpp, 2016). (Vartiainen, Tedre & Valtonen, 2020)

Input Information to Output Knowledge and the Role of Intuitive Processes

The use of computer inputs has enabled us to quantify learning through measures of sequenced interactions, binary or multiple choices made by the user, or fill-in-the-blank among other similar inputs. The type of input that is possible on many electronic devices is limited, however, to the platform and capabilities of standard keystrokes, mouse movements, and verbal commands. Text inputs are limited because the qualitative information captured is removed from sentiment and nuanced meaning (Picard et. al., 2004). At best, the qualitative inputs are quantified using indexing techniques, term frequency, or other natural language processes. Numeric input is more easily captured and assessed, to provide users with feedback. Although this is a valuable estimation of the user's language, numeracy, and navigating skill, a more telling set of data can be captured from the output of knowledge after it has been internalized, and from what many scholars describe as reflection and metacognition (Picard et al, 2004; Bada, 2015; Pande and Bharathi, 2020). Connections have been made between pedagogy embedded with formative assessment techniques and deep learning, because this form of pedagogy can achieve engagement, student-autonomous learning, and self-regulated learning that enables the development of understanding (Shepard 2019). Shepard is referring to deep learning in the human sense of fully understanding. Other forms of deep learning are described in Through Teaching Learning, studies by Emanuel Cortese (2005). He observed students who demonstrate or teach a topic to others, once they have mastered the information well enough to explain the nuance and details beyond the principle concept. Described as intuitive knowledge by Blackler when knowledge is internalized, "[t]he intuitive process integrates the information that one already has with what is perceived by the senses, and new associations between these two information sources produce insights, answers, recognition or judgements." (Blackler, 2008).

Although not explicitly part of Bloom's taxonomy or its revision, reflection is a form of knowledge that requires the process of applying, evaluating, and creating as well as others. Captured in a comparison between Ackermann and Papert's work described by Vartianen et al, "To Papert, projecting out – or externalizing – our inner thoughts was as important as internalization of our actions" (Ackermann, 2004). Vartianen et al. argued that, when the children themselves are responsible for teaching and giving directions to a computer, they need to align their internal mental model with the external representation". (Vartiainen, Tedre & Valtonen, 2020). The games developed for this study encourage a process of internalizing information by observing the sequences played and replaying them as a demonstration of their understanding. This is an elemental description of the game; however, it does characterize how information is transmitted from the toy to the player and back to the toy. It also emphasizes the importance of the child's contribution to the toy's understanding of them and their participation input on the system.

How Virtual Toys/Games Educate

Studies show that gamification of content has some value in providing novel effects to children who play along with the game. (Prensky, 2003; Charlton et. al. 2005; Freitas, 2018) This has motivational implications and will often attract children to play; however, long-term interest and engagement require a greater degree of integration of content and gameplay to truly educate players (Hodent, 2014). A now well-known example of an unsuccessful game is Math Blaster. Math Blaster has been reviewed by several game designers and educators (Scott Osterweil & Eric Klopfer, 2015; Katrin Becker, 2006) and found to have little value in helping kids learn. The object of the game is to shoot at the correct answer from a series of numbers that appear in front of the player as they fly through space. The game is a drill-and-kill type format that assumes that the learner knows the answer and this game just reinforces their understanding through a series of drills. The game is essentially a multiple-choice game; however, the developers decided to highlight the answer, so that the correct answer is obvious. There is so much more attention paid to the game aspect that it overshadows or at least diminishes the math portion. Much of a player's time is spent just catching tokens and avoiding obstacles that have nothing to do with the learning objectives. Once the game begins, it is unclear what should happen next because a math problem appears, focusing attention on the center of the screen. This shifts quickly as you are engulfed in flying the ship to shoot at or avoid objects ahead. Possible answers to the problem stream out along with other targets, and when hit, score points for the correct answer. One of the main criticisms of the game is that players are not given feedback on incorrect answers; they just move on to the next problem. Serious game developers today understand the importance of relating game interaction (gameplay) with learning tasks. This is a direct influence of constructivist theory and the importance of active learning.

Computer Community Learning from Imitation to Transference

Observations of improved student outcomes have been identified in prototype tests, which measure the performance of networked tools as well as connected digital teaching aids. These content-driven observation studies were designed to measure supplemental training outside the class, but concluded that significant gains are made in knowledge acquisition and retention of information to memory (Bosseler & Massaro, 2003). Bosseler and Massaro state that, "children also attended appropriately more often in the computer than in the teacher condition (75% vs. 65%), with a significant correlation between attention during training and accurate memory for the vocabulary tests. . . . Furthermore, the students were able to recall 85% of the newly learned items at least 30 days after training was completed" (Bosseler & Massaro, 2003). This was also true of the social-behavioral knowledge that addresses more specific interactions. The implications of these results suggest an underlying disability that manifests through physical interactions. For example, "four of the five children also exhibited an increase in their performance on the body imitation portion of the Motor Imitation Scale; all children exhibited increases in their ability to respond to joint attention" (Ingersoll & Schreibman, 2006).

The correlation between cognitive improvement and the use of physical tools was also notable in many studies (Gibson, 1962; Minogue, 2006; Blackler, 2008). For example, Ingersoll and Schreibman state "all children exhibited an increase in their use of total object imitation, appropriate language, and appropriate play from pre- to post-treatment" (Ingersoll & Schreibman, 2006). Similarly, Cai, Wu, Liu, and Hu (2018) also showed improved generalization results: "participants were able to transfer the rules and skills, suggesting the positive effects of using gesture-based games" (Cai et. al., 2018). It is unclear if these two studies used a control group to determine the difference between tool use and more traditional teaching methods. Also, the question of the long-term effect on memory and true knowledge, attained for longer periods, goes unanswered.

Brain studies provide the most sanguine results, particularly for autistic students, specifically outlining methods for training students to learn and transfer what they learned. Strategies for teaching use a "novel and counterintuitive prediction that the subgroup of individuals with [high functioning] ASD who perform better when trained with highly simplified input sets consisting of a single prototypical exemplar rather than multiple variable examples (Church et. al., 2015). These findings support the findings of Cai et. al.: "Children with ASD prefer to interact with simplistic devices, as they can easily get overwhelmed by sensory stimuli." Building on simple concepts to develop more complex skills also maps to most AI training sets. The student provides a wider range of response data to predict user behavior where technologies outperform humans in storing and calculating response probability.

AI Performance in Education

In a study that measures the potential of AI in predicting learning strategy based on personal preference for content and difficulty level, Wei et al. (2021) discuss their approach to recommending new video resources. The model being tested “use[s] the fraction of students participating in learning educational videos, the degree of completion, and the correct ratio of answering quizzes to estimate their ability.” The three criteria for inclusion here only consider their use of video resources, completion rates, and question correctness drawn from the user quizzes to describe ‘ability.’ Use of the resource is required just to be part of the data set. The categorization of students is “[b]ased on their ability, students can be divided into three main types” (Wei et. al., 2021). Although it is unclear what the three groups are, the category of correctness is heavily weighted in this scenario. As we have seen in many studies on student evaluation, completion and correct responses are too narrow a definition of ability (Bloom, 1957; Anderson, 2001; and Krathwohl, 2002). Although we have seen from traditional evaluations of student performance measures, many more factors go into assessing how much a student understands.

Student performance is central to many studies of machine learning predicting the success or failure rates of exams or courses (Sekeroglu et. al, 2019; Mansur, 2019; Rivas et, al., 2019) which informs teaching strategies for students (Zoana et. al., 2022), as well as pre-performance strategies students use, described by Wei et. al. (2021). These studies are useful in their intended understanding of the current state of student approaches and performance which follow up with decision support mitigation, teaching strategies to follow, or provide new or previously viewed information to correct performance outcomes. For example, Zoana et al. (2022) use an AI model that follows a history of performance and offers a teaching strategy that best suits the student. These approaches assume the follow-up information or strategy will be adopted and implemented in the future. However, we have seen that students are most likely to change thought behavior if corrections are made at the time of the error (Lhyle and Kulhavy, 1987). Education, particularly with young children, requires a just-in-time approach to error correction having a greater impact on future behavior and thought processes. (Lhyle and Kulhavy, 1987)

In terms of the analysis of knowledge acquisition, machine learning techniques have been used to identify patterns in the way that children with autism acquire new knowledge. For example, one study used machine learning to analyze the language use of children with autism and found that it was possible to use this information to predict language outcomes and inform language interventions (Linsted et al., 2015). The study included a sample of 726 children with autism who were between the ages of 1.5 and 12 years old. The researchers evaluated data on the children's language use using a variety of data points from US children in eight different states, including structured supervision and applied behavior analysis. They then used machine learning algorithms to analyze this data and predict the children's learning outcomes relative to behavioral therapy and supervision (see Applied Behavioral Analysis (ABA) in the Technology and Autism section). A linear regression model confirmed a 35 percent improvement relationship between therapies to learning outcomes.

The results of the study showed that the machine learning model was able to accurately predict the performance outcomes of the children with an accuracy of 60 percent. Although not an impressive number, the researchers found that the model was able to identify specific factors that were similarly associated with positive language outcomes, including the child's age, cognitive ability, and social communication skills. The study also highlighted the gender skew toward males, which reflects the autistic population split of 65 percent male to 35 percent female. Another concession of the study points to the importance of individual differences in the design of interventions between states, due to accessibility and cost to families (Linsted et al., 2015). This research reflects current practice generalizing differing practices and techniques of therapy, which is understandable, but inconsistent. In other education studies, machine learning has been used to personalize learning and adapt to the needs and abilities of individual students, making the variation between students less random. For example, one study found that machine learning algorithms were able to identify patterns in the way that students learned, and used this information to adapt the difficulty of the learning material to better match each student's abilities (Cen et al., 2018). The limits of AI studies on education often feature selection or variation in features to make predictions. Where Cen et al. research indicates the presence of patterns in learning strategy, a higher prediction rating is necessary for levels of confidence in the system to be adequate. Adding more features has promise to improve the predictive rates, but the question of which features and what impact they will have is still in question.

There is some evidence that machine learning techniques have the potential to facilitate the transfer of knowledge from one subject to another. For example, one study found that machine learning algorithms were able to learn to classify objects in one domain (e.g., animals) and then transfer this knowledge to classify objects in a different domain (e.g., vehicles) with high accuracy (Shin et al., 2016). In a study of human transference Ausubel et. al. (1969), suggested that prior learning is not transferable to new learning tasks until they are first over-learned. ‘Overlearning’ is a term to describe comparable ideas or concepts to reinforce retention and, at the same time, broaden the understanding of the same idea. Brophy and Evertson (1978), reported that mastery learning levels of 80 percent to 85 percent seemed to produce significant learning gains without negative student attitudes toward instruction when exposed to similar concepts. This would support the idea of overlearning which, according to Ausubel et al., is even beyond reflective or externalized demonstrations of knowledge. Compared to multiple exposures to an original text (rehearsal) Ausuble et al. found that “since overlearning of the original material involves both more complete and more explicit repetition than that involved in rehearsal, the facilitating effect of overlearning is accordingly much more pronounced" (Ausubel, 1969). To the end that similar concepts are being learned by using physical interactions as overlearning, the use of two different games that have similar concepts fits this definition.

Conclusion

Educational systems in the Western world are set up based on age and general categorization that follow Piaget's four stages of cognitive development of children including the sensorimotor stage (ages birth to 2 years), the preoperational stage (2 to 7), the concrete operational stage (7 to 11), and the formal operational stage (12+). Parallel to this, children progress through the aged stages of formal education.

Technology Literature Review

Technological advantages in education promise to be more efficient and accurate, and to thoroughly remake education. However, the lived experience of technology in the classroom has not fully delivered on these promises, often leaving educators to wrangle content and master platforms, and to troubleshoot technical issues. Some of these problems stem from technologies built to perform functions in a way that suits the platform rather than the user (Chauhan et al., 2022). Still other problems have occurred due to ever-changing platforms that become outdated, or to maintenance of systems that have not kept up, plaguing systems and software. Cloud-based systems alleviate some of these problems by pushing maintenance back onto companies that run and update them continually rather than through software versioning (Bulaghi et al., 2020). The promise of machine learning (ML) shifts even more interactions onto the Cloud, delivering content-based predictions of need, interest, or the likely requirements people have for systems (Bulaghi et al., 2020).

Educational technologies that use machine learning models can predict a student's general levels of understanding and offer material that engages and challenges them to improve their academic performance. In addition to predictive capability, ML addresses these challenges by providing personalized interventions tailored to each child's specific needs. (Oliveira, 2020) Machine learning algorithms can analyze data from various sources, including physical behavior, language use, and other indicators of developmental progress, and use this information to predict which interventions are most likely effective for a particular child (Seminara et al., 2019). Technologies like this would allow educators and therapists to design more targeted, efficient interventions tailored to each child's specific strengths and needs.

Given that children with autism have a variety of learning challenges related to their cognitive abilities, communication skills, social skills, and behaviors, a prediction at the appropriate content level in an ML model can address each student's issue. Previous studies that applied machine learning include the following approaches and results.

- Difficulty with language and communication: Many children with autism have difficulty with language and communication, which can impact their ability to understand and express themselves. This may include difficulty with verbal communication, difficulty with nonverbal communication, and difficulty with social language skills such as taking turns in conversation.
- Difficulty with social skills: Children with autism may have difficulty with social skills, such as initiating and maintaining social interactions, recognizing and responding to social cues, and engaging in cooperative play with peers. These difficulties can make it challenging for children with autism to learn in social settings and to develop social relationships.
- Repetitive behaviors: Children with autism may engage in repetitive behaviors, such as hand-flapping or pacing, which can be disruptive to learning.
- Difficulty with attention and executive function: Children with autism may have difficulty with attention and executive function, which can impact their ability to focus, plan, and organize their learning.
- Difficulty with sensory processing: Children with autism may have difficulty processing sensory information, such as sights, sounds, and textures, which can impact their ability to learn and focus.

Attention and Executive Function - Intelligent Tutoring Systems (ITS).

Intelligent tutoring systems have had varied successes in aiding neuro-typical children in learning. The difficulty in creating suitable lessons that are successful at teaching is the foundational structure of ITSs, that form a platform for teaching a variety of subjects and integrate content with the interaction. Kurt VanLehn discusses Intelligent Tutoring Systems (ITS) as a structure divided into an outer loop that coordinates the learning tasks to choose from, and an inner loop that manages the steps within the tasks with hints and feedback such as "correct answer" or " incorrect answer" (Reiner, 1999). Rowe (2011) evaluated six different systems based on their characteristics with schematic details, which correspond to the level of technology advancement and choices determined by the designers and creators of the system. Four common types of outer loops are listed below, the last two of which were formally recognized by VanLehn (Rowe, 2011):

- Ordered Learning - the student selects learning tasks from the menu of all tasks;
- Sequential learning - the teacher assigns tasks in a fixed order;
- Mastery Learning - the teacher sets tasks from the pool of one chapter until the student completely masters the knowledge contained in this chapter;
- Macro-Adaptive Learning – the teacher tracks the characteristics, both the fixed ones, such as learning styles, and the variable ones, such as correct and incorrect knowledge components.

Assigning learning tasks through each of the outer loops can be simple, such as sharing and changing tasks based on the level of performance of previous tasks. Of all the listed outer loops, only a few contain task generators that can be added to the list of tasks needed to complete a given section. The ultimate goal of intelligent learning systems is to create genuinely adaptive systems that dynamically generate new tasks based on the student's needs and interests.

Generating tasks is a complex process, which is discussed in the future research section of this study; however, step generation and analysis are two of the more difficult stages to consider. Intelligent tutoring systems use artificial intelligence (AI) tools to solve this problem based on the interactions and reactions of the students using them. Next-generation personal learning systems are model-based adaptive tutors (MBAT) that integrate AI elements in a range of features built specifically for education, such as:

- modeling cognitive and affective states of the student;
- the use of dialogue to engage the student in the Socratic learning experience of inquiry and discussion, questioning and answering;
- incorporating open learning models that promote reflection and self-awareness;
- adopting meta-cognitive structuring (for example, through dynamic aid or narrative structure) to increase learner motivation and engagement;
- the use of simulation models (for example, to facilitate learning a foreign language and enable learners to engage effectively with native speakers by understanding cultural and social norms) (Siemens, 2004)

Results of subsequent meta-analyses by James A. Kulik and JD Fletcher (Schank, 2006) out of 50 systems, suggest that properly deployed intelligent tutoring systems (ITS) are effective and can compete with flesh-and-blood teachers and outperform standard, unassisted education. Successful implementation was measured by the experience of teachers working with students in parallel with the system, not by letting students work with ITSs on their own. One of the more interesting findings of this study is that even students who receive extra learning support do not perform well on standardized tests compared to 'local' tests that consider the curriculum used in class (VanLehn, 2006). This seems strange when students who understand the material can be expected to use knowledge transfer as a strategy to achieve consistent results regardless of the test. At the same time, tests based on a given curriculum are probably more suited to the student's knowledge. In this respect, intelligent learning systems differ from educational video games in that teachers may emphasize the transfer of knowledge to another context as a sign of learning, while moving to the next level of play requires skillful action to demonstrate mastery of the material.

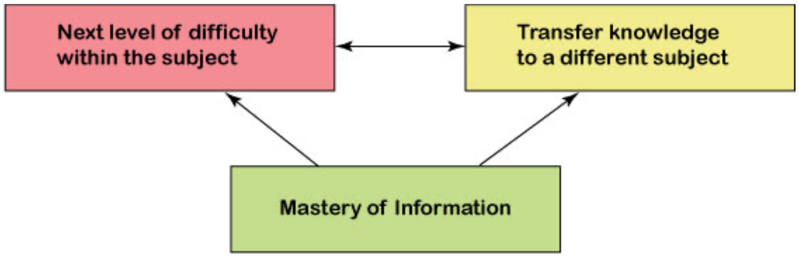

Next level of difficulty within the subject
Transfer knowledge to a different subject
Mastery of Information

Figure 2.3 - Influence of Difficulty and Transfer of Context on Mastery

Children with autism tend to focus on material they are comfortable with and practice the same or similar processes repeatedly. Klin et al. suggest that previous “psychological and experimental literature on the ASDs focuses on deficits in a wide range of abilities, some recent theoretical models pose that the style of learning of these individuals may also result in relative strengths within their profile, or even in absolute strengths relative to typical peers.” (Klin, 2007). ASD students have a less holistic view of their knowledge or configural processing and what the revised taxonomy categorizes as conceptual knowledge (Klin et al., 2007). Their attention advantage labeled circumscribed interest (CI), viewed from a positivist position, “would be at some advantage in aspects of knowledge that require featural, fragmented, or rote learning” (Klin et al., 2007). Conversely, there are clear implications of such focus, which can be difficult in social situations and if these repetitive behaviors become substitutive for attention needed during school. CI is conceptualized as the interests or preoccupations of individuals with ASD that become unusual in intensity and/or focus.

These interests appear to increase in intensity over the individual’s lifespan (South, Ozonoff, & McMahon, 2005). Boyde et al. found that CI behavior, when used as a stimulus in social behavior, garners the best results. Using an antecedent-based technique that integrates the CI into social situations instead of a reward (consequential-based technique) for participating in social behavior had much greater success (Boyde, 2007). Although it is unclear what makes a circumscribed interest a focus for any particular child, their use in educational settings bolsters social engagement, which has a cyclical effect when learning objectives are applied. (reference)

From a study using a tutoring system, Trafton et al.(2013) identify the difference between student outcomes that used the system and those who were taught in a lecture setting versus those who learned on their own, “The difference between the two conditions in which the students were tutored [was] not significant, but both were significantly faster than the students learning their own.” (Trafton et al., 2013). The students in this study were college-level, and none had reported having autism; however, the speed did have an impact on the number of questions they could answer on the test. This finding is not insignificant because it is precisely the condition in which we find autistic students struggling to recall or comprehend a task quickly. The more interactive tutors and systems become the delivery mechanisms for education, the more efficiency plays a role in evaluating performance. As long as performance measures include timed evaluations, autistic students will need additional practice to remain at grade level.

How Machine Learning Works to Mitigate Autistic Behaviors

There is some evidence that machine learning techniques can predict a person's level of any subject understanding (Wittman, 1998; Korkmaz and Correia, 2019; Shemshak & Spensor, 2020). Studies examining math education often evaluate performance, specifically those studies that identify patterns in how individuals have learned math in the past. For example, one study found that machine learning algorithms could accurately predict students' performance on math exams based on data such as their past performance on math exams and their learning style (Orcos et al., 2019). It is difficult to determine the accuracy of the prediction, even if past performance is a fair assessment measure for prediction. In part, it is because of the training set of the model that performance across any country or countries have several factors, such as formal education, economics, and access. These are compounded by a wide range of teaching quality with few standard competencies that either students do not reach or reach by narrow means (Cleveland et al., 2000; Orcos et al., 2019). Some other recent studies indicate that domain-general cognitive abilities, more specifically executive functions, may provide reasonable explanations for variability in early math learning (Bull & Scerif, 2001; Espy et al., 2004; Passolunghi, Vercelloni, & Schadee, 2007). These studies, however, focus on performance alone, or on performance and one other indicator. What is a benefit for future potential performance is only an indicator of outcomes, and not general learning behavior.

Ahadi et al.(2015) employed a different approach to try to predict students needing assistance using ML. This study bases predictions on past performance (grades) but also considers gender, age, focus of study, and previous experience in the topic. Features that have been shown to have little effect on grades, such as age (all the students in this study were of similar age), gender, and focus of study (all the students were in the same major), are independent and identify a current condition that is somewhat fixed during the study (Ahadi et al., 2015). Although an argument can be made that age, gender, and selected major may change, the duration is longer than behavior, for example, which may change from one moment to another. The result of the study uses the Random Forest classifier, which "was able to categorize students on the Exam Question, the Final Grade, and the combination of both with the accuracy of 80%, 73%, and 71% respectively" (Ahadi et al., 2015). As we can see, the prediction is still focused on performance and each student's grades but relatively early, within the first week of class. What is notable is the accuracy of machine learning compared to traditional methods of predicting students in need. Although their machine learning model outperformed the conventional method of identifying low-achieving students, it also identified high-achieving students, as well as students who fell in between these two difference poles. Ahadi et al. report inconsistent results between the fall and spring semesters, indicating context influences grades and performance measures. The study also suggests that the student's behavior is applied after the performance evaluation to remedy the problem of low-achieving students. Behavior considered during learning may be more effective, particularly with students with learning deficits, and supports personalized learning.

"Personalized learning with machine learning" by Cen et al. (2018) is a meta-analysis of machine learning in personalized learning, which involves adapting the learning material to individual students' specific needs and abilities. The authors review the literature on the use of machine learning in personalized learning and highlight several key areas where machine learning has been applied, including:

- Adaptive learning systems: These systems use machine learning algorithms to adapt the learning material's difficulty to better match individual students' abilities.
- Personalized recommendation systems: These systems use machine learning algorithms to recommend learning materials to students based on their interests and abilities.
- Learning analytics: These systems use machine learning algorithms to analyze data on student performance and learning behaviors to identify patterns and predict student outcomes.

The authors also discuss the challenges and limitations of using machine learning in personalized learning, including the need for high-quality data, the need to consider ethical and privacy issues, and the importance of considering the cultural and social context in which learning occurs. Overall, the paper highlights the potential of machine learning to support personalized learning and improve student outcomes, as well as the need for further research to understand its potential and limitations fully.

Adaptive Learning Systems

Adaptive learning systems are educational technologies that use machine learning algorithms to adjust the difficulty of the learning material to better match the abilities of individual students. These systems can use a variety of data sources, such as assessments, student performance, and learning behaviors, to continuously adjust the learning material to meet each student's needs.

Using adaptive learning systems in various contexts, including traditional classroom settings, online learning environments, and adaptive test preparation systems, can be helpful to students who may be struggling or need additional support to keep up with their peers.
There are several critical components of adaptive learning systems:

- Learning content: This refers to the material that students are learning, such as text, videos, or interactive activities.
- Adaptive engine: This is a machine learning algorithm that uses data on student performance and learning behaviors to adapt the learning material to better match each student's abilities.
- Learning management system: This platform manages the delivery of the learning content and tracks student progress.

Adaptive learning systems have the potential to improve student outcomes by providing personalized learning experiences tailored to each student's needs and abilities. However, careful consideration of their design and implementation ensures their effectiveness.

Learning Analytics

Learning analytics can be used in a variety of contexts, including traditional classroom settings, online learning environments, and adaptive test preparation systems. It can be particularly useful for identifying students who may be struggling or who may need additional support, and for identifying patterns in the way that students learn and adapting the learning material to better meet their needs.

There are several key components of learning analytics systems:

- Data sources: These are the sources of data that are used to analyze student performance and learning behaviors, such as assessments, student performance data, and learning logs.
- Analytics algorithms: These are the machine learning algorithms that are used to analyze the data, identify patterns and predict student outcomes.
- Learning management system: This is the platform that manages the delivery of the learning content and tracks student progress.

Learning analytics can improve student outcomes by providing insights into the learning process and identifying areas for improvement. However, it is essential to carefully consider the design and implementation of these systems to ensure their effectiveness and to avoid potential biases.

Personalized Recommendation Systems

Personalized recommendation systems are educational technologies that use machine learning algorithms to recommend learning materials to students based on their interests and abilities. Personalized recommendation systems typically use data on student performance, learning behaviors, and interests to generate recommendations for learning materials tailored to each student's needs. For example, a personalized recommendation system might recommend math videos to a student who has struggled with math or recommend reading materials on a particular topic to a student who shows interest in the subject. Learning analytics is the use of data and analysis to understand and improve learning and education. It involves the collection and analysis of data on student performance, learning behaviors, and other factors to identify patterns and predict student outcomes. Machine learning algorithms are often used to analyze these data and identify patterns that may be difficult for humans to detect.

Connecting Theory of Mind, AI, and Educational Toys

Education and treatment for children with autism typically involve a range of interventions designed to support their cognitive and academic development, including their ability to acquire new knowledge. Technology-based interventions include software and assistive technologies such as text-to-speech and communication boards that deal with single deficiencies. These tools can help autistic children access educational materials, aiding their communication with others. However, they need to attempt to improve learning efficiency and support the pace or pattern they naturally implement to learn to be more efficient. For many early ITS systems, theoretical frameworks such as Adaptive Character of Thought (ACT - changed from Adaptive Control of Thought) guided the structure of software programs and later machine learning. Programming this way was partly done because of the technological constraints of 1980s computers but also because of an incomplete understanding of cognition (Anderson, 1983). Anderson goes on to explain the theory using two principal factors "that affect human cognition and organize them into a complete cognitive theory, [that] consists of a set of assumptions about a declarative memory and a procedural memory." (Anderson, 1983).

Declarative memory deals with facts and long-term memory retrieval, whereas procedural memory operates using schema or, to use the psychological term, 'productions' (Anderson, 1983; Anderson, 1991;). Implications of this theory for tutoring systems center the problem objective or goal on loading the working memory by giving instructions in the problem-solving context. For example, to solve for C if $A^2 + B^2 = C^2$ and A = 5, B = 3, the goal is to find the answer to C within this problem-solving context provided by the calculation. Understanding the goal in this context is similar to the way game designers work to integrate the lesson into gameplay. The context of the interaction makes the content implicit to the user. Introducing context is akin to applying the physical world interactions into the virtual closer to simulation than some arbitrary interaction: "[t]his goal-directed character of cognition proves to be the key to much of the tutoring effort" (Anderson, 1983). The goal becomes a shared starting point for the ML model and humans. Another important implication of these principles is that students should be given immediate feedback about their errors (Norman & Neilson, 2010). It is not only to correct the student but also to help develop accurate schemas later to be used in other contexts: "[t]his will make it easier for the student to integrate the instruction about errors the new productions that they form." (Anderson, 1983). The theory offers a structure not just as a framework to build tutoring systems but also to recognize human interaction as a source of errors that occur, collectively knowing how to accomplish the goal.

More recent versions of the ACT theory of cognition have been extended to include rationality (ACT-R) as a complete theory of mind, explained more fully in the theory section (Chapter 3). It is noted here for its influence on computational thinking, particularly operationalizing machine learning in IoT objects (Anderson, 1995; Trafton et al., 2013; Ritter et al., 2018). The natural tendency to model computers to think like humans is understandable but clearly insurmountable in practical terms. We have recently seen the reflection of ourselves in artificial intelligence used in large language models. Although still developing and problematic, they function on limited data features (Chen & Rosenfeld, 1999), producing much less than what we would expect but still producing comprehendible cohesive texts and images. The addition of *Rational* to ACT is structured similarly where declarative and procedural memory and information, such as tactile, visual, aural, etc., are buffered together within a model but use limited data inputs to perform tasks. Trafton et al. describe multi-model systems to operate using "several limited-capacity buffers which, together, comprise its context" (Trafton et al., 2013). Granted, the broad scope of Trafton's research deals with the complexity of robotics, but still they emphasize that machine learning that accounts for human limitations has the best opportunity to succeed. Rather than mimicking the behavior and cognition of human beings, machine learning should understand these limitations that humans have, recognize them when they happen, and support error correction when needed.

Trafton et al. also extends their theory toward embodiment (ACT-R/E), recognizing the limitations of ACT-R to be bound to the constraints of a computer, referred to as 'screen-as-world.' Even though ACT-R uses visual and auditory data to inform the predictions, a full accounting for peripheral vision and ambient noise in these scenarios has not been done. ACT-R/E uses two models in addition to those briefly described here, enabling spatial reasoning in a three-dimensional (3D) world (Trafton et al., 2013). These models are not used for locomotion, motoric function, or robotic depth perception, "while useful to a robot during its own localization and navigational process; it is not at all useful when trying to understand why humans systematically think they are closer to landmarks than they really are" (Trafton et al., 2013). In several studies on haptics and education, touch is used to form as detailed structures in memory as vision (Srinivas, Greene, and Easton, 1997; Calik and Kargin, 2010; Eid and Osman, 2016) and together provide a deeper understanding but also the context of the subject/object in the world. In a literature meta-analysis about Haptics in Education, Monigue and Jones (2006) attest to studies that reference *active touch,* a deliberate manipulation of objects that engages students "in consciously choosing to investigate the properties of an object is a powerful motivator and increases attention to learning" (Sathian 1998). Active touch also informs users *how* to perform a task through manipulation, although Monigue and Jones (2006) could not reference any study that examined this proposition. More recent studies show that human haptic exploration is efficient and robust to noise yet adapts rapidly to changing conditions (Prescott et al., 2011; Seminara, 2019).

Later studies described by Zhacharias Zacharia (2015) in his paper "Examining whether touch sensory feedback is necessary for science learning through experimentation" were inconclusive, comparing physical manipulation and basic virtual manipulation (without haptic devices). Citing much of his work revealed physical interactions that inherently have haptic information available were more conducive to students' science learning than the use of virtual manipulations (Zacharia, Loizou, & Papaevripidou, 2012). Conversely, instances in which the use of virtual manipulations was more supportive than the use of physical ones (Finkelstein et al., 2005; Zacharia, 2007; Zacharia, Olympiou, & Papaevripidou, 2008), as well as instances in which the use of virtual and the use of physical manipulations was equally supportive for learning (Zacharia, 2015). Although active touch has generally been shown to support learning compared to non-constructivist approaches to education, it is unclear to what extent active touch or haptics' impact on expediting, easing, or committing information to long-term memory.

Machine Learning - Connecting Socialization and Education

Another potential application of machine learning in the education and treatment of children with autism is the prediction of intervention outcomes. By analyzing data from past interventions, machine learning algorithms can identify patterns associated with successful outcomes and use this information to predict likely future interventions. These applications could allow educators and therapists to optimize their interventions by focusing on the factors that lead to positive results, such as lower rates of disinterest and lower dropout rates (Costaet et al., 2017; Adnan et al., 2021). These approaches to ML use are often directed at larger populations and their interactions with MOOC's. Some interventions operate at a higher level of prevention, where students must stick with a course. Few studies show that attempts to address the issues at the point of content-specific engagement rather than course engagement.

Although successful generalization of social skills across adults or peers has been reported (e.g., Charlop et al., 1985; Krantz & McClannahan, 1998; Williams, Donley, & Keller, 2000), a number of studies found that participants' performance levels in the generalization context were below that in training (e.g., Charlop-Christy & Kelso, 2003; Delano & Snell, 2006; Leaf, Dotson, Oppeneheim, Sheldon, & Sherman, 2010; Thorp, Stahmer, & Schreibman, 1995; Woods & Poulson, 2006). Several factors could be responsible for a participant's level of response toward adults not being comparable to that directed toward peers. In particular, irrelevant stimuli in the training context (e.g., physical features or responses of the recipient), alone or in combination with relevant stimuli, may have gained control over the trained response (e.g., Rincover & Koegel, 1975; see also Kirby & Bickel, 1988, for discussion). The presence of an adult versus peer per se may also set the occasion for differential responses due to a child's different reinforcement history with adults and peers. Moreover, generalized responding may be rapidly extinguished if the type, amount, or schedule of reinforcement provided by the peer is not adequate to maintain the level of performance observed during training (Foxx, McMorrow, Bittle, & Ness, 1986).

Conclusion

Overall, it appears that machine learning has the potential to be a valuable tool for supporting the education of children with autism. However, the combination of technology, design and education is a complex issue particularly in the context of special needs education. Determining what variables are indicators of learning can be done isolating each data point, but machine learning calculates probability based on the interrelationship of data. This study will consider both the impact of the individual data and the veracity of the data in combination. For this reason, this research is needed to fully understand the potential of these techniques and to ensure the research responds to the participants in an ethical and responsible way.

Chapter 3 - Theoretical Frameworks

The theory of mind operates here as a meta-theory that ties together the first three cognitive frameworks that underpin this research. In the context of autism, the theory of mind captures the cognitive challenges addressed by the toy. Autistic children have been shown to have difficulty shifting to a perspective outside their mind to recognize the perceptions of others. In part, this explains the concept of mimicry often demonstrated in educational constructivist theory. Characterized by the internal construction of knowledge by doing, constructivist theory is not limited to trial and error methods. Direction and often physically guiding autistic children is a way of learning through interaction categorized as constructionist theory. For neurotypical children, constructivist approaches allow discovery, reinforcing what works and what does not. For autistic children, repetitive activity often reinforces learning, which resembles the flow state. The theory of mind also posits a perspective of oneself in the future or different contexts. Research suggests that this inability to see outside yourself limits the imagination to apply concepts within one context only to be transferred in certain conditions to another. Tailoring theory and theories around games, engagement, and flow are specific to the content interpretations and the resulting state of mind that trigger pleasurable experiences to the point of immersion, encouraging learning. These theories are more commonly applied to communications and gaming; however, overlaps in the frameworks suit the purposes of building an educational toy and, therefore, support the approaches to its physical design and the study research design. The application of probability theory using neural networks is mainly discussed when developing the machine learning model. Moreover, it connects to other concepts concerning thought and thinking about learning.

A Meta-theory for Autism Spectrum Disorder

Since the 1980s, discussion around the theory of mind reveals it is inadequate to argue as a complete theory because it cannot explain all the deficits that characterize autistic children. As a theory, it does not account for some assets that autism presents in some cases, which are absent in others. Neuro-typical children can view an activity, concept, or scenario from their perspective, knowing that others may have a different mental image from their point of view. It is a type of conscious knowing of our thoughts. Frith and Happe describe it this way, “to have a theory of mind is to be able to attribute independent mental states to self and others in order to explain and predict behaviour“ (1994). This definition would suggest that the social-communicative problems of people with autism are the consequence of an incapacity to attribute mental states to oneself and others (Baron-Cohen, 1995). It also infers that the attributions applied to oneself are predictive, or of the future. The theory identifies autistic behavior in experiments such as the Hanoi test, which examines learning strategies through mimicry. Without a concept of one's mind, observing others for social and communication cues fits a profile of many autistic children; however, not all. Frith and Happe, who are skeptical of the definition, characterize the theory as a mentalizing account. “The mentalizing account has helped us to understand the nature of the autistic child’s impairments in play, social interaction, and verbal and non-verbal communication. But there is more to autism than [this] classic triad of impairments” (1994). Instead, they propose an extension to the theory of mind to include executive function and central coherence to explain other characteristics of autism.

Noens and Van Berckelaer-Onnes (2004) explain additional characteristics of autism that are not covered by the theory of mind. Executive function and central coherence “…are much broader and also try to explain the non-social features of autism. The **executive functions theory** suggests that shortcomings in planning, flexibility, organization, and self-monitoring are the core problems in autism (Ozonoff, 1995; 1997; Ozonoff et al., 1991; Russell, 1997). Bailey et al. (1996) describes the executive function as an umbrella term for a constellation of mental operations, not all of which have yet to be delineated. Although we see the development of executive function as autonomy to organize, plan, and adapt to things at a high level, it is delayed or prolonged in autistic children, yet exists in some cases with objects. It manifests in compulsive behaviors, such as lining up toy cars or grouping Legos by color, which is externalized behavior rather than internalized regulation (Garretson et al., 1990). According to Garretson et al., ASD students prefer simple, repetitive, self-generated tasks, and their study on performance only showed significant changes in assigned tasks that happened late in the test. Performance measures have more to do with motivation than cognitive ability (Garretson, 1990).

The central coherence account rationalizes autism as a weaker drive for information integration. Central coherence is the natural tendency to process incoming stimuli globally and in context, pulling information together to acquire a higher-level meaning. People with autism, however, tend to process incoming information locally and piecemeal (Frith, 1989; Frith and Happé, 1994; Happé, 1999). Linguistics tests are often used to determine the deficit using two or more sentences; the first provides context, and the second is an ambiguous sentence to be deciphered using the first context. For example, ‘the roar of the crowd was deafening. The fans distracted the team’. The test question is, what happened? a) The football fans distracted the team, b) the cooling fans distracted the team, and c) the football fans helped the team (Jolliffe, T., & Baron-Cohen, 1999). These tests found that autistic people performed worse than the control group with rare uses of ambiguous terms such as homographs (i.e., read, lead, bow, close). However, they performed only slightly less well on the more common use of the term. The local or direct meaning of a given sentence dominates their attention. This attention can be seen in autistic children’s preoccupation with subjects and the ability to focus on a self-generated task for sustained periods. Integrating several contexts in one device provides a world, albeit smaller than the global context of one’s life, which allows one to explore different behaviors. Shifting contexts, however, requires interpretations such as the state of things like modes, which is an ambiguous concept. For example, a smartphone uses similar interactions on vastly different operations and features within the phone. KeayBrith and Howarth (2011) indicate that the direct response to interactive electronics encouraged more engagement than those interfaces that required a mapping of input to response. Participants in studies who are autistic are challenged by mapping different contexts onto similar spaces. This supports Garretson’s assertion that interactive tasks should be simplified (Garretson et al., 1990).

The meta-theory that provides an epistemological view of autism for the theory of mind, executive function, and centralized coherence points to the restricted development of an internal dialogue to acquire social information to control responses. The internal processes using some behavioral therapies for developing this are done differently. Yet the externalization toward objects that ask children to mimic behavior, organize their activities and plan activities, and transfer information from one context to understand another has been observed. At least in part, these externalizations are part of learning, even though the meta-theory deals with the internal awareness of our thoughts. What is unclear is if the externalization of these behaviors is eventually internalized through having rehearsal stand in for what is developed directly in neuro-typical children.

The choice of games for this study investigates the use of rehearsal to stand in for human mimicry. Color sequences displayed for the player to mimic like Simon Says removes the performance of the human and its social implications for the autistic child to recreate. In this scenario the player’s flexibility in thinking as games change also challenge executive function without the influence of others. The toy removes the social aspect in learning to focus on the cognitive behaviors guided by presence of mind while retaining the adaptability of teachers for each individual.

From Constructivism to Constructionist Learning Theory

Traditionally, we view cognitive theory as a set of schemas or mental constructions of symbols that change as we learn or alter schemas (Anderson, 2018). For the individual learner, Anderson adds, “constructivist theory supports the learner’s acquisition of skills and power, such that he or she can articulate and achieve personal learning goals” (Anderson, 2018). The epistemic view of learning focuses on the evolutionary preoccupation for curiosity, discovery, sharing, and understanding for the skillful use of tools. It is most closely associated with social constructivist learning theories (Anderson, 2016). For the ASD student, these motivations to learn are limited, if at all present in their minds. The ASD student is less preoccupied with relationships with other students and their surroundings, taking them at face value and not necessarily relating them to each other.

> "…in the industrial age, an educational system sorted students – it separated the children who should do manual labor from the ones who should be managers or professionals. So, the "less bright" students were flunked out, and the brighter ones were promoted to higher levels of education. This is why our schools use norm-referenced assessment systems rather than criterion-referenced assessment" (Reigeluth, 2016)

Constructivism is a theory that has its roots in psychology, starting with George Kelly (1955). He used the term "personal constructivism," which describes the relationship between the world and an observer of the world, forming a distinction between what is real and what is perceived to be real (Kelly, 1955). The observer adjusts their reality upon the failure of their perception when it does not comply with the reality of physical and system properties (Raskin, 2002). The perceived world of children can be particularly entrenched because of the child's lack of experiences (Kelly, 1955). Constructivists divide into a second analogous sub-theory that further separates the real and perceived worlds– radical constructivism. The difference between radical constructivism, defined by Glaserfeld (1989), and Kelly's theory (personal constructivism) is that it communicates with the real world. Glaserfeld contests that individuals only know their own reality, which may or may not align with actual reality, making viable "experiences to already existing sensorimotor or conceptual structures," which are also perceptions. This distinction separates the real and perceived worlds, even further isolating the observer who never fully knows the objective 'real' world. Glaserfeld also borrows from Piaget's learning theory, which advances the notion of cognitive adaptation (Ackermann, 2001; Raskin, 2002). Cognitive adaptation recognizes the evolution of thought and thought patterns children have as they mature into adulthood. But like Ackermann (2001), the combined view of constructivism of Kelly and Glaserfeld suggests that a motivation to adjust our worldview will gravitate toward the objective natural world but this only describes the motivation that fits with the natural world. When a motivation is confronted by a social or collective view that contradicts the real-world view, it is mainly due to the complexity of the natural world being beyond the individual's comprehension.

The social implications of the perceived reality have less influence on autistic children because any social norms inferred are not decoded. Garretson et al. (1990) suggest that tangible reinforcement is more effective than social expectations, "the efficacy of social reinforcement declined at a steeper rate than that of the tangible reinforcement." However, tangible reinforcement is not necessarily a motivator for prolonged engagement. It may also be due to a weaker theory of mind or unsolidified centralized cohesion needed to connect these perspectives and transfer their local memory as they understand concepts to social reinforcement, which supports long-term understanding. For this reason, a relevant learning theory for our purposes will focus on constructivism; however, the focus is on the toy as a substitute for a type of tangible reinforcement, given it operates as an educational game. In that case, a child would have to adjust their motivation to accommodate the toy's behavior, which may sometimes be unexpected. Observations of this influence the toy has on a player will be evident in the way they play the games and generally with the toy. Following the game according to the rules or norms the toy sets suggests a complicit or at least a contractual attitude toward the toy. How willing the player is to comply tests their willingness to mimic and, conversely, the toy's ability to influence a player to play along.

Piaget's definition of constructivist learning theory states, "...knowledge is experience that is acquired through interaction with the world, people and things" (quoted by Ackermann, 2001). This emphasis on interaction is predicated on the observations made by Piaget as children interacted with tools and devices and described their understanding of how they work. Piaget noted how strong the child's mind held these perceptions and did not see them as fragile or fluid as much as defensible and viable for that stage of learning about the world as they knew it. (Piaget, 1959). However, a third interpretation of constructivism arose from contributions by psychologist George G. Hruby (2001), who advocates for a constructionist approach noting the social construction of understanding the world, crediting Papert as hierophant of constructionist learning. Like constructivism, a building of knowledge occurs around learning activities; constructionist approaches are social in nature, whereas constructivism focuses on the individual. Hruby proposed the inclusion of community into the constructivist framework, raising the value of social influence on held perceptions. Most applicable of Hruby's tenets of constructivism states, "there is a coherent and dependably consistent reality that is the basis for our sensations, even if our sensations do not resemble the causative phenomenal bases, or 'onta' that prompt them" (Hruby, 2001). Hruby argues for a collective understanding of reality, which is negotiated between people or learned together. How much is learned through community and social interaction is debatable but is acquired through verbal negotiations and non-verbal assumptions made by community members.

The role of the community provides a distinction made by Raskin between constructivist and constructionist theorists who "favor using the term "constructionism" rather than 'constructivism.' This distinction reflects the social constructionist's aversion to the notion of an isolated knower" (Raskin, 2002). However, it is precisely this inability to read and decode social knowledge that autistic children miss from their education. They are the proverbial 'isolated knower,' contextualizing knowledge locally.

Although the constructionist theory is problematic for autistic children, it still highlights the importance of interacting with the world to explain the learning process. Papert (1980) describes this process as “building knowledge structures through progressive internalization of actions.” The unique experience each of us has with the world is an argument for customized learning, and building a record of that unique experience over long periods helps locate gaps, visualize preferred topics, and assess aptitude relative to time and quality of learning. From the perspective of the constructionist theory, we can see that there is a shared world understanding among neuro-typical learners – supporting the idea of patterns to acquiring knowledge; however, there are contending arguments suggesting a flexible understanding that adjusts to changes in the world we think we know. Both constructionist and constructivist theories assert that we attain knowledge through a unique physical experience of the world, reconciled with the social view, allowing for individual perspectives. ASD students have some understanding of these social views, but they are obtained individually and explicitly through their own experiences. The theories describe how knowledge is acquired, but the next level of cognitive development shows how it is applied, adapted, and repurposed for new experiences.

Although the changes in the physical interaction are the same, differences in the cognitive interaction are significant. The color sequences only require memorization of the correct pattern of colors where the math game requires the player to recognize the operation between numbers and the function of the operation. Multiplication is more difficult than addition but the sequence of button pushing is the same as the color sequence limiting the cognitive load to “what is the response” rather than ‘what and how to respond”. The player also recognizes their physical response triggers a response from the toy which is rewarding or not. The toys design reenforces behavior using lights as an indicator of correct / incorrect answers. A simple expression from the toy but effective in encouraging the player to answer correctly building knowledge structures previously described by Papert (1980).

Transfer of Learning Theory

According to David Perkins’ (1992) definition of transfer of learning, it “..is always at least implicitly contrastive: it assumes learning within a certain context and asks about impact beyond that context.” To understand the transfer of learning theory, we must first clearly define the variations of the theory's title and the confusion surrounding the concept as it relates to this research. What is often referred to as Transference of Knowledge describes the passage of information from one person or entity to another, which is complicated by the interpretation of the message between the sender and receiver described by Shannon and Weaver's communications model. An often-cited example happens in business, involving the changeover of staff or the shift changes that occur in hospitals, where vital information must be communicated accurately and in a time-sensitive manner. Transfer of knowledge, on the other hand, refers to the passage of information from the cognitive state to the active state. This shift happens within the mind of one individual; however, it deals with changes in focus and triggering approaches to a similar or even the same problem previously thought about, now acted upon. The familiar saying of 'putting theory into practice' best describes this transfer but is only related to this study and its value in understanding their interpretation of information (see education literature review section).

The confusion is not just a semantic issue in the word choice of 'transference' over 'transfer' or 'knowledge' over 'learning.' In the seminal text, "The Cognitive Basis for Knowledge Transfer," Glick and Holyoak (1987). describe knowledge transfer as "a phenomenon involving change in the performance of a task as a result of the prior performance of a different task." (Glick and Holyoak, 1987). It is a broad definition; however, they narrow and differentiate its meaning from various other theories listed here and simultaneously move it toward the more current definition of 'transfer of learning.' Perkins and Solomon formulate a finer distinction in their text 'Transfer of Learning,' which compares two somewhat different mechanisms. "Reflexive or low road transfer involves the triggering of well-practiced routines by stimulus conditions similar to those in the learning context. Mindful or high road transfer involves deliberate, effortful abstraction and a search for connections." (Perkins and Solomon, 1992) The reflexive mechanism uses the memory of the context triggered by the familiarity of entities, objects, systems, or relationships that allow for an action or thought to be reused. Mindful transfer of learning is a deliberate shaping of knowledge to fit an unfamiliar context, not intended for its original use. It is a form of lateral thinking that coerces information and context to comply and form new approaches. This study focuses on reflexive (low-road) transfer using short-term working memory. Learning the initial information deals with direct interaction, organization, and signals like LED lights, similar to the transfer domain learning process.

Glick and Holyoak (1987) use the term "forward" when describing low road transfer, highlighting the novel characteristics of the context. They observed the initial learning of the information or skill applied to a novel context, which can be tested shortly after the initial acquisition of knowledge. Longer durations between one task and another are described as 'far' transfers, a reference to their diametric positions of near transfer, which are recently learned and soon after applied. (Glick and Holyoak, 1987). In 1949, Osgood was critical of the 'retroaction' transfer studies, which he argued did little to change the empirical outcome of those tests that alternate tasks from the initial to novel and back to the initial task – which also measured retention of information. The importance of transfer of learning is that it recognizes the flexibility of knowledge and contributes to the retention from short-term to long-term memory (Gaines, 1987).

Memory naturally has a significant role in the transfer of learning, and the retention level in memory is the critical point. The notion of learning suggests a level of retention that is available for recall but does not reference expert or novice use of the information. The breadth of knowledge is as significant as the vertical depth of what Hiebert and Lefevre (1986) describe as conceptual knowledge, "[c]onceptual knowledge is important in relation to procedure selection, procedure monitoring, and the transfer of procedural knowledge to new situations (Hiebert & Lefevre, 1986). It is essential for the construction and flexible use of solution procedures." (cited by Blöte et al., 2001). Learning a game's interaction procedures is an example of the breadth of information that quickly becomes procedural (Gaines, 1987). We see this in game design found in the game objectives and the sequence of interactions that inform the player how to play. The specifics of what to do in any given scenario, often acquired through trial and error, lead to in-depth knowledge. The pattern follows what Osgood (1949) described earlier: initial (reflexive knowledge) to the novel (specific knowledge) and back to initial knowledge.

Theories for Games

Where educational strategies and technological mutations shift the ground for people who have autism, game theory promotes the use of a narrative to scaffold learning, an example of where technology can be more inclusive. The use of narrative is relevant to theories of learning through stories that can connect abstract concepts to real-life experiences. Drawing from simple examples, stories visualize social skills and fundamental cognitive concepts (Schank & Berman, 2006). Considering other research covering meta-analytical findings, the value-added aspects of educational relevance in games appear to be as prevalent in computer and game literature as in other multimedia forms (Slota & Young, 2017).

Conscientious game developers create narratives shared through multiplayer games, understanding the trend toward open-ended stories. Virtual environments dynamically populate, coinciding with the open-ended stories, using player contributions to build stories and game content. Children with ASD often cannot participate in multi-player games because their focus does not model others, a behavior known as joint attention (Bosseler & Massaro, 2003). The inability to follow a point of interest along with others may preclude children from using open-ended games; however, the toy can operate as an asynchronous delivery system for narratives created by others. A narrative can form a connection between the narrative's producer and end-user. A connection also forms with environmental circumstances between the producer's life-world experience, the end user's life-world, and the environment or medium in which the narrative is embedded (Slota & Young, 2017). The story is still shared between people but is told through the game's development as interpreted by the student/player.

Extending game technology further, in response to user input, we could capture and store stories created by ASD children to encourage them to relate concepts they have mastered. No single narrative could provide an ideal context for all learners. However, Slota and Young outline a strategic approach that could help. Their strategy uses a “generator set,” creating pairs of stories designed to highlight the invariance in math courses across varying scenarios (Slota & Young, 2017). One story in the pair describes the problem for students to solve, while the second story uses the same skill in a new scenario. Similar to Vygotsky’s activity theory framework (1978) and supported by the transfer of learning theory, game theory helps us to focus on the intersection between individual student/player ideas, the technology that mediates their play, and how their social interactions help generate and transform their ideas within the workings of a game. In this study, we use the toy as a social construct to guide the student and their interactions. Using the initial knowledge of how to play, the ASD student masters the initial problem – following the color sequence, for example – before being led to a second scenario, the math problems. Ideally, they do this using the same problem structure, but computationally, a new story is generated to demonstrate the application in a new context. For example, the color LEDs change to white LEDs on a button that highlights the numbers – signaling the new story being told by the toy. The math sequence also includes operators such as ‘+,’ ‘-,’ and ‘=’ that form a new context that connects the lights and emphasizes the importance of order but is different enough from the color sequences.

The game's design uses the transfer of learning theory but depends on the level of engagement students/players have shown while playing. Many studies have shown that successful educational games can engage students (Bandura, 1982; Garris et al., 2002; Filsecker and Kerres, 2014), a measure to evaluate the toy’s potential. Several factors lead up to engagement, such as novelty, interest, and motivation; however, keeping a player interested in a game requires a balance of challenges and skill (Csikszentmilhalyl, 1970). The state of mind that triggers pleasure sensors that keep us interested in an activity with intense interest (often called ‘flow’) also shares characteristics with ‘hyperfocus.’ Games capitalize on these two attributes of challenge and skill, offering small tasks just difficult enough so they are not dull, but in trying to accomplish the task, support skills that make it seem doable. A feedback loop is formed by thinking about the task, attempting it, considering what was done, and responding to feedback that indicates success or failure.

The process mirrors learning, according to Garris et al. (2002), which can be organized as an input-process-output (IPO) model (Garris et al., 2002). If games associate the learning objective with the interaction, the engagement is held within the game context – so long as the challenges are matched to the player's skill levels. Bandura (1982) describes this as “reciprocal determination," which occurs among the player's perceptions during gameplay, the player's actual actions and behavior, and the system's responses to those behaviors. Machine learning is well suited for responding in real time to the player's actions and the level of challenge they are ready for. Engagement is not a static state – it represents what Filsecker and Karres (2014) call the 'mediator variable' between an educational game and its learning outcomes.

Tailoring Theory - Customization is not Tailoring.

A benefit to tutoring systems discussed in the literature is the increased motivation that encourages students to engage in the subject. Early computer use in classrooms credit, "the increase in student motivation and effort appeared to be related to, among other things, a lessened sense of embarrassment at mistakes; one might hypothesize that AI programs that encourage the sense that one's mistakes are private might be more conducive to enhancing student effort" (Schofield et al., 1990). The protection of individual feelings from peers indicates the broader concerns surrounding public education but begins at a time when user-centered design also enters the conversation. From a commercial perspective, user-centered design is a process of understanding the consumer's needs, wants, and desires and giving them appropriate tools for solving their problems in a given context. The result, which again meets industry needs, tends to cluster groups of users together, offering a few customizable options rather than just one fixed solution. Tailoring theory comes from communications disciplines and is often used in health messaging to directly respond to identity, culture, and personal beliefs, or what Rimer and Kreuter (2006) collectively call theoretical dimensions and demographics. "In general, tailoring along dimensions that are culturally or theoretically derived tends to be more effective than tailoring along "surface" traits, (Huang & Shen, 2016) such as basic demographic characteristics" (Christy et al., 2022). Much of the literature demonstrates that tailoring significantly affects persuasion and motivation to engage in marketing messages. Although, little research has been done using this theory in pedagogical scenarios. Tailoring is directional, generated by the AI, and is distinguished from a user's choice of options. The theory runs parallel to traditional agile programming, which anticipates user interaction but is based on generalizing group members with similar interaction behavior.

Conversely, AI is a response to previous information that classifies the interaction. It also generates new classifications when encountered or is localized forming patterns uniquely identified by each user-generated action. An example taken from health messages describes the personal approach from systems that call you by name rather than you choosing features from a set of options (Christy et al., 2022) and includes search histories that include medication or symptom searches. The advantage is the patterns and influence of a data in combination that at least appears to be customized for each user.

Lastly, tailoring gives the impression of personalization experienced in human-to-human interactions. Machine learning communication needs to be identifiable and ‘feel’ like it is reacting to the immediate context in order for the participant to recognize responsive change (Winkler and Roos, 2019). Making the toy prototype in this study was more challenging because it has a ‘fixed’ interface where the physical buttons only changed with the color of light, unlike digital interfaces that have the potential to be symbols. The button's responsive nature can only be experienced over multiple sessions of play because the game is never the same between sessions, responding to the user's input. Therefore, the prescribed nature of tree-structured programming, although offering multiple paths to play the game, is still finite or fixed. The tree-structured program is, at best, tailored to one type of player deciding the difficulty level or changing the starting point within the game, whereas the ML game is genuinely tailored to the user’s level. Making predictions for the next set of questions is unique to each session and each player. How this effects player performance has yet to be determined however, machine learning provides a suitable platform for them to engage in the questions and adapting to their performance. If they sense the changes the toy makes based on their success or failure it becomes a tailored experience, and not just randomized to be different.

Technological Implications for Education

There are a number of challenges that students with ASD face, each having been studied extensively. The combination of these challenges makes it difficult to find solutions. Critical traits in autistic children identified by researchers reference their inability to recognize social cues, which slows their social growth and cognitive progress (Ingersoll & Schreibman, 2006). Children with autism also use strategies of imitation less than neuro-typical children, limiting their learning from others as well as through objects (object imitation). On one hand, some ASD children share a trait described as ‘joint attention,’ which teaches not only social skills by sharing an experience but builds language skills. On the other hand, keeping the attention of ASD children is difficult, which suggests that change might be helpful, and yet they also have difficulty with change. Transitions between activities should be slowly implemented since sudden changes in the environment or persons engaged in the activity may cause them to act out (Ingersoll & Schreibman, 2006). In addition to these difficulties, many external frameworks set out challenges for ASD students that could make learning more difficult. Several educational theories over the past decade and more portend a synergy between technological and educational shifts in favor of the prevalence of technology (Goldie, 2016). This prevalence has already begun for some, but the implications of the underlying theory often inform practice, and the practice can unintentionally exclude individuals who struggle, not with technology in general, but because of the expectations implied by its use. Technologies can be either inclusive or exclusive, depending on our understanding of the implications of theories that connect technology to education.

Connectivism proposes that learning may also reside in technology – where it may be stored and manipulated by human and non-human agents (Siemens, 2006). Mental models of these complex systems equip users to traverse the system for information gathering and organizing operational spaces that are highly personalized. Complexity theory helps build the mental models that enable us to conceptualize future impacts (Anderson, 2018), such as the impacts of economics on the political system. Both theories require understanding relationships and flows of related networks when the inevitable unanticipated disruptive technology destabilizes a system. We have seen many disruptive changes recently, only to have the system re-configured to meet the needs of increasing paradigm shifts. These changes can affect many people, demanding flexibility and agile responses to changing environments with retooled skills. Access to technology is not the only concern for education; learning to change with technology is also critical. If the changes are less suited to ASD students, the technology can potentially increase the speed at which they fall behind.

Contemporary views of education theory, such as Connectivism, "view knowledge as sub-symbolic, with meaning arising from interaction [between] sets of connections" (Downes, 2006). Goldie (2016) defines learning as occurring "through the construction and traversing of networks… participation in network activities results in the creation, removal or adjustment in strength of connections." Concepts such as networked relationships are intellectually complex; however, children learn about the interconnectivity of 'things' by handling, using, and mastering them. Though connectivism has yet to become widely accepted as the leading learning theory for the digital era, as envisioned by Siemens (2005) and Downes (2006), many education researchers and theorists actively examine the notions of interconnectivity as a pedagogical imperative.

A similar argument is being made by the serious game community, which advocates for educational games. Schank (2006) asserts that people create and use cognitive "scripts" to anticipate events and recall them based on story frameworks, planning actions around scenarios they prospectively play out in anticipation of them happening in the future. The connectivist employs similar scenarios with networked entities that are typically unrelated (Zheng & Gardner, 2016). The ASD student will likely have problems with connectivist learning paradigms, given their inability to generalize knowledge and make connections this way.

A proposed architecture to study observable knowledge in action can be understood using Adaptive Control of Thought-Rational (ACT-R), a framework for analyzing task performance.

Ritter et al. (2018) describe this as a cognitive architecture: “a fixed set of mechanisms that use task knowledge to perform a task, thereby predicting and explaining the steps of cognition that form human behavior.” The leap that research has to make is what is going on in the brain and what actions are being performed. ACT offers a connection between knowledge about an action and the inferred knowledge from the object or context of the activity. “The complexity of the human mind presents a significant “black box” problem, wherein researchers are forced to infer information processing mechanisms from behaviors. Rational analysis provides a solution through the assumption of mechanisms' optimality (i.e., rationality); if multiple mechanisms are possible, the choice of optimal performance suggests which mechanism is most likely.” This assumption of optimal mechanisms for human behavior offers a method for reducing the problem space by providing constraints on the gamut of plausible mental mechanisms (Anderson, 1990). The following sources provide further explanation: Chater and Oaksford (1999) give a retrospective on the first decade of rational analysis; Anderson and Schooler (1991) provide an in-depth description of its foundation and methodology; finally, Anderson’s (1993) explanation of ACT-R includes the integration of the rational analysis methodology. (Ritter et al., 2018).

Since the model also divides knowledge into two distinct kinds, it can inform teachers that the best way to ensure information is encoded and easy to retrieve is to make that information both declarative and procedural. Students should be encouraged to combine their knowledge with actions. For example, combining an action or activity with a learned fact means that information will be encoded as both kinds of knowledge and, therefore, deepen the level of learning (Anderson, 1991). The combination of physical interaction while playing, or the action, and the cognitive memorization, calculation and response, or knowledge encodes the information similar to constructionist learning theory. How well the system connections to the player are made is outside the analysis of this study however, considerations of these connections can build over time as the players use the toy, improve their skills as they play, and engage with the games.

Probability Theory and the Neural Network

The use of probability theory in computer science and programming generally reimagines the algorithm and the role of computer programmers. Simply stated, traditional programming relies on the programming team's anticipation of how a user will behave. Conversely, Andreas Holzinger contrasts this with ML programming: “Probabilistic programming… is different from traditional programming, in a way that parts of the program are not fixed in advance; instead, they take on values generated at runtime by random sampling procedures.” Deep learning– including neural networks – has only recently been applied to predict educational outcomes.

According to Ciolacu (2017), four areas of the educational system have seen increases in using artificial intelligence to understand: “Artificial Intelligence will play a key role in Education, identifying new drivers of students’ performance and early disengagement cues, adopting personalizing learning, answering students’ routine questions, using learning analytics and providing predictive modeling.” To understand these use cases and how they apply to this study, we must examine probability theory to follow how it is used in probabilistic programming.

We determine the probability of future events by considering the frequency with which an event has occurred. However, frequency only tells part of the story and would take much larger datasets to make the kind of predictions with high accuracy rates such as we are seeing today. Instead, we imagine that the data being collected is not simply a two-dimensional graph of two data points but rather multivariate data, each point being weighted and related to each other through their influence. We can use these relationships for the parameters of the event that are being predicted. Machine Learning uses the Bayesian approach to **prior** knowledge of the parameters before seeing data, the **likelihood** probability of the data given values of the parameters, and the **posterior** probability of the parameters given the data (Stefan Hrouda-Rasmussen, 2021). The complexity of weighting, combining these probabilities, and making the prediction is, at least in part, the unknown of machine learning. The values within the complex black box of neural networks also change as new data is introduced. For this study, the amount of data used in training the model is fixed in time, but as more data is introduced, we would expect that the model would get better at predicting, and higher accuracy rates would occur the longer the child learns with the toy.

To narrow the scope of this study, only relevant system designs of machine learning and their neural networks (NN) will be considered; for example, the selected type of NN is best suited for Internet of Things (IoT) devices and real-time calculation and feedback is critical to the overall experience of the toy’s performance and function. Feed-forward (FNN) and recurrent (RNN) neural networks, as well as a mix of these two forms, provide a choice for balancing input and output data to improve performance or establish memory of future events (Schmidhuber, 2014). This study uses an RNN based on labeled data as a classifier taken from the tree-structured study. Developing an ML model requires a thorough review of the data and its structure to inform the type of model to be used. The data collected results from the toy’s functions and focus on student performance and physical interaction as indicators of progress with learning. The data types are a mix of continuous and discrete values as follows:

Toy Data Types:
Input Data - Independent Variables

Right vs. Wrong (discrete, binary)
Duration (continuous from 0 – end, ordinal)
Pressure (continuous from 0 – peak – 0, ordinal)
Orientation (continuous from 0 – end, ordinal)
Current Difficulty (discrete from 1 – 8, discreet)

Output prediction - Dependent variable.
easier, same, harder (discrete from -1 – 1)

Of these data points, some explanation is necessary to understand the shape of the data and what is being collected. The correct/incorrect answer is straightforward – the sequence of events must be followed precisely for the correct answer response from the toy. All other input events are determined to be incorrect. The duration is when the button interactions occur, from the first button press to the last. Although the duration of the entire play session is available, it is only used to calculate categories of players and the mean of all player's time using the toy. Pressure is also straightforward regarding the values captured (0 min – 2000 max). The pressure is the trigger event in the toy to recognize which button is being pressed and which lights indicate color or number depending on the game being played. Orientation values captured from the accelerometer are the changes in speed from one position to another based on x, y, and z coordinates. Other data is available to the accelerometer, including gyroscope degrees from 0 using x, y, and z coordinates, motion values (a sum of all three-speed values), and location (a sum of all three-degree values). The motion values were determined to be more appropriate than the other three values relative to attention because rotation on the toy sampled every second in time could be due to body position or ease of viewing. The speed at which the toy is being moved – sampled every second – can distinguish between shaking, swinging, subtle adjustments, or laying entirely still. These movements directly affect visibility and the ability to interact with the buttons regardless of a user's motivation or intention. The current difficulty level may impact all the other data points, and the ML maps the difference, weighting this data to have a greater or lesser influence on the others – specifically duration – as questions become harder.

Using mixed data type analysis can take two forms. The first is analyzing each input and combining the analysis to understand better the patterns that emerge from the data; however, this approach negates the impact of one variable on another, blurring insights the data can reveal. It also requires a human in the middle, similar to a predetermined set of comparisons of the tree-structured programming method. Although machine learning classifiers can provide more accurate predictions, it is this 'hand-crafted' approach that is time and computationally expensive. The other approach uses mixed method analysis, which requires more data preprocessing to distinguish relevant and irrelevant data. This preprocessing is best described by Holzinger (2019), who identifies the relevance of data as a weakness in ML.

> The performance of any ML algorithm is dependent on the choice of data representations. Consequently, these data representations, aka features, are key for learning; hence, much ML development goes into the design of preprocessing pipelines, data transformations, and data mappings that result in a representation that supports effective ML. Current learning algorithms still have an enormous weakness: they are unable to extract discriminative knowledge from the data (Holzinger, 2019).

The statistical features found in the dataset for the toy are more easily formed because all the data is quantitative; however, the discriminatory determination of the data is less obvious. Two questions arise here as a caution when deciding that a particular feature indicates meaning. For example, the pressure value from any one button press or the toy's orientation values hold some meaning in relation to the user. The specific meaning – such as emotion or physical ability – is not discernible in real-time but can indicate attention or interest, which, in turn, impacts learning. The toy collects the subtle variation within the interaction, which calculates what can be extracted from the amount of discriminatory knowledge affecting the data in question. Machine learning is a balance between the features that correlate to the patterns of the player's input, such as a slow, progressively harder push versus an abrupt hard press. Categorically, these are different types of input, but their influence on the determination of the machine learning response is only one weighted relationship to the output prediction. Holzinger sums up the analysis of features this way, “a truly intelligent algorithm is required to understand the context and to be able to discriminate between relevant and irrelevant features – similarly as we humans can do. “What is interesting?” and “What is relevant?” are hard questions, and as long as we cannot achieve this grand goal with automatic approaches, we have to develop algorithms, which can be applied by a domain expert” (Holzinger, 2019). The interactions can be documented in the context in which they are happening by drawing from video-recorded evidence that captures the players' emotional states and physical attributes. We can map their interaction to what the toy captures to compare to similar interactions to see the delineation between the categories. For example, if a player is observed to be frustrated, the pattern of a button press may be short in duration and accelerate quickly from 0 pressure to 1100 units at its peak. We can compare this interaction with previous interactions and from other players to assess if the toy is predicting based in part on these interactions considering the player's performance and other data features collected.

Conclusion

These frameworks draw a complex relationship between the physical interactions of play, cognitive processes in children with autism, and computational response to calculate the right conditions for learning. At the core of this research is investigating the potential of machine learning to make predictions based on the data associated with the players unique behavior. In turn, each prediction is tailored to each player in a game context that both challenges and entertains while educating them. The toy provides a game context which tracks physical interactions encouraging constructivist learning principles storing the interactions to memory. The constructivist approach is described in social contexts using mimicry, theory of mind, and executive function to clarify learning among learners. Due to social anxiety these contexts do not promote learning strategies and may limit them for ASD students. The toy's amiable presence side-steps the social relationship needed for learning collecting data to respond in a tailored way that feels like human instruction. How much influence these particular data features have on determining the correct response to a child's mindset is an open question. It is one of many questions involving machine learning responsiveness and customization based on human interaction. The most important aspect of these questions is how humans respond to the predictions that 'feel' human are discussed in the next chapters.

Chapter 4 - Research Questions, Goals, and Related Theories

In this chapter the research questions are further developed with attached goals to guide the research toward building a toy system that begins to address the implications of such a system and what it can do for individuals but also the systems the toy is embedded.

Question 1 – Are there patterns in learning discernible from interactions with the toy?

What learning patterns emerge if we map the relationship between the input values?

Goal – Determine if there are patterns in learning behaviors of children with autism. The input values are potential indicators of engagement with the game and the subject of the game. Many factors, including emotional, behavioral, level of interest, or skill, can cause disruptions to a player's engagement. This goal looks at the duration of a play session to determine a pattern within each player and compare these patterns to the other players to see not only the interrelationship between inputs but also the interrelationship between players' inputs, such as duration to answer questions and if these durations extend or shorten during play.

Theory – This research question probes the value of the constructionists and how much physical interaction plays a role in learning. If the process of 'teaching' is delivered in a consistent way and interactions tracking behaviors demonstrate learning occurred, we could capture those interactions as a representation of the learning process.

Question 2 – How does a fixed decision tree compare to ML decisions in terms of successful predictions?

Does machine learning software outperform typical tree-structured programming in making predictions for appropriate question difficulty?

Goal—Determine if ML prediction produces a greater success rate than tree-structured programming, which predetermines the possible paths for a user. The next question's difficulty evaluates a player's readiness to be challenged. The value of challenging the player is to improve their understanding and ability to succeed in the game. The goal is also to compare the prediction success of different ML software that uses the same features.

Theory – Game theory described in Chapter 3 describes 'flow' and how challenge and pleasure are in balance. Tailoring theory details the importance of challenge in the flow equation. It is clear that the tree structured decisions are not specific enough to accommodate the variations of autism. If machine learning predicts the appropriate challenge level for each player, we would expect the players performance would improve.

The following research questions are specific to machine learning software and do not compare with the tree-structured version.

Question 3 – How useful are the data features in making successful predictions?

Are the input values—performance, motion, duration, pressure, and orientation—along with the level of difficulty the best features for predicting a level of readiness to learn?
Goal—To evaluate the impact on predictions using these data points and their significance. The cumulative effect can be framed in terms of the influence of these features on performance.
Theory – A measure of the features is prediction accuracy, but this could be contextual and a result of features specific to a particular game and its objectives. Transfer of learning requires the features to be generalized focusing less on player performance and more on the effects of the other features. We would expect that patterns between the games remain the same despite the players performance.

Question 4 – How well can ML predict suitable changes to game-level difficulty?

To what extent can machine learning accurately predict the optimal moment for subject transitioning?
Goal – Identify learning patterns that signal readiness to change subjects through the game-level data and the question difficulty. The research question assumes there will be patterns in the data, and the number of patterns is of a size that supports high confidence in the probability of the toy's selection of game to play. The question also infers that the color game is generally easier to play, and transitioning to the math game is not only a transition to a new subject but also to a new level of difficulty. Although this requires the same number of cognitive actions, there is a difference between games. For example, a sequence of three colors and adding two numbers together to get an answer, the third number have the same number of actions but the difference in cognitive load between these two games is in question even though they are at the same game level.
Theory – Probability theory characterizes machine learning and neural networks, its ability and value for the complex problem of teaching autistic children. Within the limits of machine learning probability is only the 'best' possible answer and not the right answer. Current teaching methods cannot provide a right answer but how close difficulty is tied to the next question is also part of the discussion.

Question 5 – Can ML improve performance scores?

To what extent can the toy help improve a player's performance?
Goal—Determine if the similarity of the games' interactions helps the player understand the subject and, as a result, learn math. The color game provides sequences to follow and allows the

child to memorize order; however, its primary purpose is to show players how to interact with the toy. Customizing the experience by predicting the levels of the next question should improve performance. The goal is to answer if and how much performance improves.
Theory – Technology in Education is theorized to improve the experience, amount, and quality of learning. The main objective of the toy is to demonstrate the ability to make these improvements a reality. It is dependent on the machine learning model and its performance to predict as well as how well difficulty relates to performance.

Discussion

It is essential to understand the purpose of measuring the player's performance which shifts between dependent to independent variable to measure toy's performance. This provides a perspective that defines the goals of machine learning embedded into the toy. Previous player performance informs toy performance in predicting difficulty. From this interrelationship of data, it is important to evaluate the influence of a particular data point as well as the influence between them. The project goals were conceived and evolved, ultimately shaping the research design and the specific methods selected for collecting, processing, and analyzing data. The high-level goals lead to the research questions specific to the issues discussed in the literature, which can now be addressed partly because of the available technology (ML) embedded into an educational toy.

One of the most significant advantages of technology is the notion of offloading, extending our actions and thoughts beyond what is typical or expected from human activity. Offloading, described by Aazam et al. (2018) as an advantage of computer networked technology, is "…where tasks are outsourced, and the involved entities work in tandem to achieve the ultimate goal of the application" (2018), emphasizing tasks as an opportunity or outsourcing. Offloading our tasks to technology has become commonplace for those things that are repetitive or mundane. Envisioning machine learning as a second level of outsourcing follows from human to device processed by machine learning and back to the human. Although not a specific goal, outsourcing some parts of education is a motivation for this study because of the difficulty in educating children with autism. Even for neurotypical students, repetition is part of learning on the way to becoming a master of a subject. The use of technology at the first level is often promoted as inexhaustible for retrieving, processing, and referencing information (Shemshack & Spector, 2020; Sierpinska & Kilpatrick, 2012; Krathwohl, 2002). Within the second tier of the game environment and machine learning, varying the depth, difficulty, and context of the information challenges the user in a way best suited to their current state of interest in learning. Contrast the two-tiered outsourcing system with the typical approach, where the teacher evaluates each child in a classroom, mapping their progress, and offering similar inexhaustible support is challenging to sustain. Can we offload this kind of support teachers typically provide onto technologies with evaluative capability? Many factors contribute to learning, and technology significantly advances the speed, stickiness, and extent of what can be accomplished (Bolding & Rugy, 2006).

This study reassesses the notion of learning from a strategic point of view that utilizes technology. Oxford's list of motivations for developing learning strategies notes "specific actions taken by the learner to make learning easier, faster, more enjoyable, more self-directed, more effective, more transferable to new situations" (Oxford, 1990). Motivations for implementing the strategy are internal and external, developed by the individual and provided to them. Still, other scholars consider learning strategies to have fewer specific motivations. For example, Bloom (1987) argued that "there were different strategies for mastery learning, including allowing students to learn at their own pace, guiding them to adopt proper paths of learning, and providing them with individualized tutorial and feedback."

This study aims to improve learning activity by developing a device that implements a new combined strategy from Oxford and Bloom, shifting it toward the individual to make learning more accessible, enjoyable, and effective at transferring knowledge since the exercises are set at their preferred pace. Gamification of education supports this internal shift, as seen in examples studied in the classroom (Granic, 2014; Klopfer et al., 2009; Becker, 2007). However, few games go outside the subject matter and attempt to make a cognitive link between subjects. Games are designed to move between levels that increase in difficulty, assuming the player has mastered the information or skill that has come before. Options are set out for the player to move or act within the context relevant to the game, whose challenges are predetermined for the player to achieve as they come across these options. Counter to the success of gamified education is the concern that individuals focus on mastering the skill of playing rather than the lesson (Klopfer et al., 2009; Becker, 2007). Studies have shown that closely connecting the gameplay with the information delivered has a greater chance of retaining the information (Becker, 2007). These connections can be achieved within the confines of a game's environment using typical game development tools. Applying constraints becomes more difficult in an artificially intelligent controlled environment where the model adapts to the user's approach, skill level, and strategy for play (Kurzweil, 2012). Therefore, this study aims to understand learning in the context of games that use machine learning to make each experience unique to the user by accommodating their pattern of play.

From this perspective, the goals have been narrowed and delineated into three parts;

- Use machine learning to predict when children with autism are ready to learn.

- Use physical measurement to indicate engagement and embodied learning during play.
- Aid the transference of knowledge between subjects by providing different games on the same toy.

The relationship between the goals is hierarchical. At the highest level of understanding, a connection is made from the premise that machine learning can make predictions of readiness. The theory is that physical and cognitive interaction patterns signal upcoming attention or interest, followed by behavior. Gathering data from users that reference the physical and mental interactions are the closest indicators to engagement, however influential they may be on the prediction. Often, autistic children have difficulty applying concepts of one subject to another as they create their patterns of knowing in their minds. If the patterns exist and the student is engaged in the content, transferring the concepts they have retained and applying them to a new context is part of the calculation in the predictive model.

Forms of Measurement

In the literature review, we see from the discussion of the constructivist theoretical perspective the importance of hands-on learning. However, measuring this interaction using a statistical correlation method indicates how much of an impact the hands-on approach has on learning relative to other inputs that support learning outcomes. The data collected, such as pressure (on the buttons) and motion (acceleration of the toy), originate from the user's motivation to pick up or push on the toy. This study captures the user's interaction by recording observable behavior — it does not aim to provide a behavioral explanation for their motivation. Contexts for behavior have been provided when it is specific to the physical data and its relation to other inputs, but an analysis of behavior from a psychological perspective is outside the scope of this study. For example, a user's level of engagement is one of these inputs, as well as their cognitive readiness, which the toy attempts to predict. This study aims to measure the effect of physical interaction as a measure of engagement and mental readiness to assess the significance physical interaction has in supporting learning outcomes. The significance of the interactions in making the correct prediction is part of the analysis, and the literature has shown it to be worthy of inclusion.

Designing Educational Games

Since the advent of video games, haptic controllers have tried to capture physical actions that mimic the real world. Understandably, the focus has been on the interaction to engage the player rather than the relationship between manipulating content and learning. Educational game developers have more recently understood the importance of relating content to game action to reinforce learning. The context of the game environment has two benefits that other interactive tools, such as electronic tutors, do not; the first is the potential for engagement in the subject to minimize effort and willingness to play rather than to learn for learning's sake. The second benefit is the ability to replay a problem as often as necessary for the skill to be learned and mastered. Although both types of programming studies here can replay the sequences of the two versions, we expect the machine learning game to be a better experience, given its ability to adapt to the user. Some variation happens in most traditional video games today, but it is not in response to the player's cognitive ability; it is only their physical skill. At the task level of interaction, the use of neural networks predicts human behavior referenced in the physical interactions in a learning process map. These maps are familiar to everyone and are not a monolith that resembles the current public educational system. In this study, artificial intelligent toys attempt to record the learning patterns, aiding students with autism to learn, mimic, transition, and apply new information. This study will capture the many variations of patterns found in the game interactions and responses, categorize them, and analyze the qualities of these categories. I have outlined some of the more exciting characteristics of the data, but more importantly, the questions from the study target the learning experience of autistic children. The aforementioned research questions include immediate goals but also underpin the project's larger goal: to determine the value of ML tools in education. Formal methods are also identified in the next chapter to assure the significance, correlation, and confidence in the evidence of their answers.

Significance or What It Means for Autism and the Larger Population

The significance of this research is its impact on education and understanding patterns of behavior that occur as children with autism learn. In a broader context, design, and technology have historically created products and services for underserved populations only to be extended to the larger population because of their benefit. The patterns of learning found in autistic students are unique to their strategic approaches to learning. However, machine learning could also be trained on input from neurotypical students. The study also measures the significance of physical input data and more typical measures such as speed and accuracy to make up the pattern. The constructivist theory has established the benefit of hands-on learning, but without the influence of machine learning to customize content matched with the learner. This study provides a framework for designers and developers when creating smart educational toys and games that take advantage of physical and digital interaction.

Chapter 5 - Study 1: Usability Testing

A usability test aims to gain insight on a prototype that can simulate functionality, strategies, preferences, and thought processes that are ambiguous or need validation for its design. Tests of this kind require a set of tasks that encourage potential users to work through the function, behavior, motivation, or actions they would encounter on the ‘real’ device. They are open-ended in that users are left to proceed through the task as they usually would outside the experiment. Observations of the tasks are video recorded to capture physical interactions, both seen and unseen, to interpret the rationale for the behaviors.

The first of three observational studies was intended to test the toy’s set-up, functionality, and general ease of use. It was conducted with 16 participants from various backgrounds. The selection criteria for participants included an age of 25 or older, given the median age of parenthood in the United States (National Center for Health Statistics, 2020), and at least one degree from a post-secondary institution. Generally, these criteria match potential teachers, physiotherapists, or parents of 6- to 10-year-old children. Of the three studies, this one used a paper prototype to simulate the future versions of the toy and focused on functional issues of setup and play of the games. The prototype is equipped with electronics that use lights and a microcontroller in order to deliver the games, similar to the second and third studies. The electronics are administered by the facilitator, who adjusts the order of the tasks and any changes to the speed and level of play. A usability test method was suitable to uncover functional and user expectations without developing a fully functioning toy. The 16 participants were divided into two groups; the first eight found technical problems easy to solve quickly. Their feedback would improve the toy's usability for the second group of eight participants. All participants were asked to perform the same five tasks using the think-aloud protocol: verbalize the actions and thought processes as you do them. These sessions were video recorded and transcribed. Tables 5.1 and 5.2 summarize tasks and responses between the two groups.

Participants – Inclusion / Exclusion Criteria

A random sample of qualified participants was selected, characterized by a persona built around typical physiotherapists, teachers, and caregivers of autistic children. Fourteen people had post-secondary education, mainly bachelor degrees, including two childhood therapists; however, 12% of people in this study did not hold any post-secondary education, reflecting the parent population who may not—of those participants who held a degree, only one had previously worked with children with autism. This study aimed to test the toy and observe the behavior of a typical guide to users who could help or explain the game. Performing well on this test does not require a degree but is a requirement reflecting the education level of people who typically help autistic kids.

All participants were over the age of 25, with 85% being over the age of 30. This was not a strict requirement because not all participants represented parents of the child age range of 6-10 years. According to the National Center for Health Stats (NCHS), the average age of first-time mothers is 24-26, depending on the year – statistics taken 2000-2014 (T.J. Mathews, M.S. and Brady E. Hamilton, 2016) setting the youngest mother with a six-year-old at 30, with 34 at the top of the range. Gender was also not a qualifying factor; interestingly, there were only four males out of 16 participants over 34. All participants spoke English as a first or second language and could understand the instructions to describe the game and its rules.

Study Design

Observational Method – Moderated Assessment and Comparative Method

Each group within the study was given the same tasks, with additional questions asked of the second group to interrogate changes made to the prototype provided to the first group.

Immediately after acquiring the participant's consent (see consent form Appendix E), the participants were asked questions about their experience with toys of this type, their use of technology, and their understanding of educational games (see Table 5.1). The participants began with the usability test, followed by an exit interview to provide additional feedback. During the usability test, participants were asked to follow the 'think-aloud' protocol that includes a discussion of what they are thinking and doing during play and identifying why they are doing it. They were given a scenario in which they would be playing the game to provide them context for playing.

Participants were asked to perform five various tasks during the test. These tasks will relate to the operations and general gameplay of the toy. They are as follows:

Task 1: 'Set up the toy so the child can play' – This test aimed to determine whether the toy can be turned on and connected to the Internet. Once connected, the toy should automatically find the network, and the first sequence will run.

Task 2: 'According to the instructions, a round of play is five color sequences and five math questions. Play two rounds with the toy.' – The purpose of this test was to determine if participants could recognize the start and end of each game sequence. It also determined if participants could recognize when transitions between sequences occur.

Task 3: ‘Play more rounds with the game in a way you suspect a child would play to determine the appropriate difficulty level” – The purpose of this test was to determine if participants could recognize the level of difficulty change, what it is that changes, and how much more or less difficult the game has become.

Task 4: ‘Toggle from the color sequence game to the math game' – This test aimed to determine how easy or difficult it was to switch between the two games.

Task 5: ‘Attempt to fix the toy's performance” – Connecting and reconnecting to the internet is a task for the caregiver to perform when needed. This requires the participant to reset the connection using the central button or turn the toy off and restart.

Participants were encouraged to provide input at any time during the test. Follow-up questions were asked after each task, as well as an assessment of the difficulty level for the task. Each participant was instructed to interact with the toy as if they were gathering information to teach children how to use it. From this perspective, they were to attempt to use the typical functions and features that a player would encounter, as well as some issues that may arise if the toy should malfunction or need assistance in its performance.

The tests took place in open office environments such as conference rooms or larger office areas with open discussion tables. Participants sat opposite the facilitator in front of the toy, which was tethered to the circuit board and computer via a ribbon cable (see fig. 5.1). Measures of participants' responses, both quantitative and qualitative, were collected – such as usability problems, errors, completion rates, task time, and satisfaction ratings (Lewis and Sauro, 2016).

Introductory questions included general experience with digital toys and games and comfort level with computers and technology. The qualitative data collected was codified and evaluated based on the values assigned to comments using Lewis and Sauro's (2016) method in “Quantifying the User Experience.” For example, the question concerning how often a participant plays video games was divided into three categories: never played, played in the past but not now, or currently played. Within these general groupings, a follow-up question asked how often you played or play per week, which was only relevant to the latter two categories. The weighting of these results provided a relationship between the speed at which the participants answered the questions and the amount of time they played per week. A weighting was given to participants who played more than three times per week in the past to those who only played one or fewer times per week currently (See Table 5.1 and Discussion section for detailed analysis).

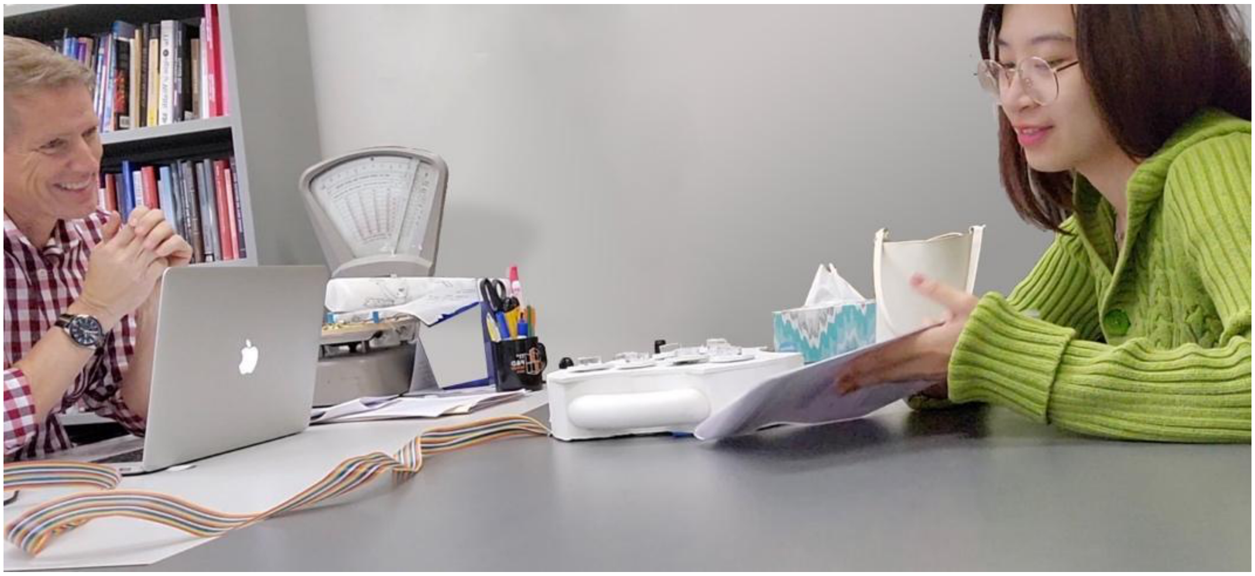


Figure 5.1 Usability test with a paper prototype.

After the introductory questions, participants were reminded of the think-aloud protocol and reassured that they could not provide ‘wrong’ answers. Five scenarios and tasks were given to each participant. Video recordings of each session captured their responses to the tasks, as well as their interactions with the toy. Correct responses to each color and math sequence were collected manually by following each mouse press and/or spoken action that followed the sequence displayed by the toy. These results are also provided in Table 5.3. Although not every participant pressed the buttons, their verbal and/or physical response was treated equally as correct or incorrect. Three of the 16 participants either only verbalized their responses or provided a combination of only button presses or verbal indications of their answers. A ranked value was given to each task based on the level of difficulty, 1 - for easy to 5 – difficult. Tasks were selected based on the variables of data to be collected. For example, the speed of task one, ‘setting up the toy,’ is the most relevant factor for caregivers preparing the game. Although the speed to answer each game question could have been collected, it was less important for adults who knew math. Task 2 collected right versus wrong answers, and Tasks 3 and 4 collected data on the difficulty level between the color and math sequences and the difficulty within each. The 5th task tested the participant's ability to recognize problems with the toy operation and their ability to repair the toy if needed.

The exit interview consisted of general comments about their experience during the usability test and any other information they could provide to improve gameplay that was not discussed during the test. The participants were asked to base their responses on other games or toys they have played with and their usability preferences. Once this was completed, remuneration for their participation was provided, and the test was completed.

Study Scenarios and Tasks

Quantifying the Responses – Codifying terms for Analysis.

Stage I – the first eight

The first group of the study tested half of the participants to establish a benchmark for the toy and the parts of its functionality that could be represented in a paper prototype. The prototype was built to include some electronics for the tests to be similar to the duration of gameplay in the final version. The paper prototype had indicator lights for the color sequence and math games (Figure 5.2). Each game provided five color sequences and five math sequences for the participants to follow. The toy changed from one game to the other automatically once the five questions were asked without repeating the questions. Three random colors were chosen and displayed, prompting participants to push each button after they lit up, similar to a 'Simon Says' game. Following the five color sequences, the game mode button (center) is masked with a '+' sign lit up to indicate the change in the game to math sequences. The setup for the game also included connecting the toy to the local WiFi and switching the game on. Changing the game mode from color to math sequences and back was also a task.

Modifications to the prototype were made in response to the first group test, which set the benchmark for the toy. The most significant change to the prototype after the first 8 participants was its software and the values associated with the duration of the lights. Initial settings of 600 milliseconds for both the light to be on and off were too long for adults to wait for the sequence to play out. The 'on' duration was reduced to 400 milliseconds, long enough for participants to register the light and short enough to move the game along at a reasonable pace. The facilitator determined durations between the color and math sequences to accommodate participants' comments and questions. Additional minor changes were made to the toy, such as adding the '+' mask for the central button to indicate the math sequences. The initial indicator was the central 'game mode' light; however, it became clear that more feedback was needed between sequences (see discussion section). The mask was placed over the game mode light and subsequently lit up at the start of each math question.

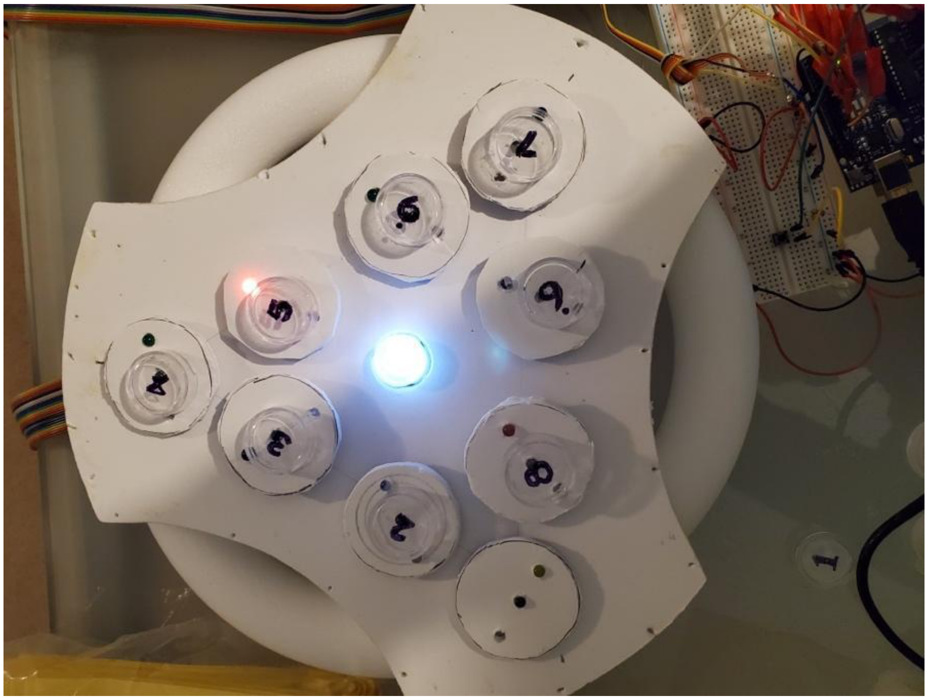


Figure 5.2 Paper prototype toy with lights for both color and math game

Prior to stage II of the usability test, a few changes were also made based on the feedback from group one. A follow-up question was added to task 3. Participants were asked to change the difficulty level back to the default once they identified the current level. This is achieved by pressing the game mode button for 3 seconds. In order to gain insight into their understanding of this function, task 4 was added after the first three participants. The function of game mode button operates based on the current mode of the toy. The starting default game is the color sequences at level one. If the game mode button is pressed for 3 seconds, the game first changes to the math game, then to the next level if pressed a second time for 3 seconds. Task 3 starts on the more difficult level (level 2), and the participant is asked to switch back to the easier level 1 (press the game mode button for 3 sec. once). Task 4 required the participant to understand what game and level they were in to make the change. For example, if they are in the math game, they must first switch to the color game by pushing the game mode button once. Once there, they must press it again to switch to level 1. The task was added because of the complexity of the game mode button.

Usability Findings

Data Collection and Analysis

Usability testing involves observing users interacting with products and services when given tasks. In this usability test, I follow the model of Jakob Neilson of NNg, implementing the think-aloud protocol and recording the users' thoughts as they describe them during the test. Each participant was prompted to use the protocol, and follow-up questions were asked after completing each task. The following tables capture four of the five criteria previously noted by Lewis and Sauro (2016): Tables 5.1 & 5.2 – Usability Problems and Errors of Group 1 and Group 2, Table 5.3 – Completion Rates and Difficulty Rankings, Table 5.4 – Time taken to perform the tasks. The fifth criterion of user satisfaction is discussed at the end of this chapter with references to participants' comments.

Ten participants reported needing feedback to indicate correct/incorrect responses after each sequence answer. The overwhelming response raises concern over gameplay and how to implement a signal that may slow down play, lessening the user experience. The paper prototype did not include one critical interaction: the motion of the button press and popup once an answer was given. The toy will wait for the correct number of button presses and popup once this number is reached. If the sequence of presses is correct, the pressed buttons will throb on and off in unison. If incorrect, the correct sequence will light up in the order they should have been pressed. In part, the absence of feedback was understandable, given the limitations of the paper prototype, but some aspects of keeping score were important for adult players.

The game mode button has three functions: it indicates that the game is on and functioning; if held for 3 seconds, the game will change from the current game to the alternate; and it indicates the operators for the math game (i.e., +, -, =). The button is placed in the center of the toy and flush mounted to diminish its priority relative to the other raised buttons on the toy. Six participants pressed the game mode button as part of the sequence of pressing buttons. They identified the center light as part of the sequence because it would light up to indicate that the math game had started. For example, the sequence would be "+, 3, 5, 8" where the + sign stayed lit while the other buttons lit up. The expression of the sequence was, "this is addition - three plus five equals eight." The participants who included the plus prior to the sequence suffered no consequence; however, the question of transference comes into question. If the game mode button is perceived as part of the button sequence and includes an "=" sign as it does on later versions of the toy, the expression at a minimum would be "3 + 5 = 8", for example. The sequence of button pushes may be enough to relate the color and math games; however, it was also clear that adults paid less attention to order during the math game because they understood that "5 + 3" is also equal to 8. For children new to math, this commutative property is just one concept to be learned.

Usability Problems and Errors

Table 5.1: Task Responses 1 – 8 Problems and Errors * Stopped pressing buttons but responded verbally or indicated following along. ** Pushed the correct buttons but not in order *** Intentional mistakes to see what would happen.

	Session Tasks Stage 1				
	Task 1: Set up the toy so the child can play.	**Task 2**: According to the instructions, a round of play is 5 color sequences and 5 math questions. Play 2 rounds with the toy.’	**Task 3:** Determine any change in difficulty and then change the game back to the default level	**Task 4:** Switch from the color sequence game to the math game to begin	**Task 5:** Attempt to fix the performance of the toy
	Difficulty Rank 1-easy, 5-hard Time: 0 min 0 sec. Timestamp: start - end	Possible Answers out of 20 Could the participant explain the game? Was the explanation accurate?	Can you identify the change? Can you change it back?	Can you change to the math game?	Can you identify there is an error? Correctly Fix:
1	I'm confused about the feedback. An icon was displayed, and the game mode button was on, but there is no relationship between the two.	Right Answer: 18 Wrong Answer: 1*** Non-attempts; 1 Can You Explain How to Play? Yes Was it accurate? Yes	Correctly ID Change: Yes Changed Back: NA	Changed to Math: NA	Correctly Identify Error: Yes Correctly Fix: On/Off
2	Misspelled the URL twice. Once on the site was able to input username, password, and toy description.	Right Answer: 17 Wrong Answer: 1 Non-attempts; 2 Can You Explain How to Play? Yes Was it accurate? No, they did not understand the difference between the color game and the math game.	Correctly ID Change: Yes Changed Back: NA Identified faster speed as a change as well but not accurate.	Changed to Math: NA	Correctly Identify Error: Yes Correctly Fix: Reference Guide, Connections, and Network
3	I was looking when it says flip over to the bottom of the toy, I was assuming that everything would be on the bottom.	Right Answer: 20 Wrong Answer: 0 Non-attempts; 0 Can You Explain How to Play? Yes Was it accurate? Yes	Correctly ID Change: Yes Changed Back: NA How do I know if it is a color sequence?	Changed to Math: NA	Correctly Identify Error: Yes Correctly Fix: On/Off
4	I need to go to… right now; I'm looking to see if it is on Bluetooth. Since I'm already on the network.	Right Answer: 20 Wrong Answer: 0 Non-attempts; 0 Can You Explain How to Play? Yes Was it accurate? Yes	Correctly ID Change: Yes Changed Back: NA	Changed to Math: Yes	Correctly Identify Error: Yes Correctly Fix: On/Off then Network.
5	QR codes would be good for people like me because I am not good. Already broke it. I typed it in.	Right Answer: 18* Wrong Answer: 1 Non-attempts; 1 Can You Explain How to Play? Yes Was it accurate? Yes	Correctly ID Change: Yes Changed Back: NA Identified light duration has changed when it has not.	Changed to Math: Yes	Correctly Identify Error: Yes Correctly Fix: On/Off, Guide
6	Yeah, it was pretty self-explanatory	Right Answer: 17** Wrong Answer: 1 Non-attempts; 2 Can You Explain How to Play? Yes Was it accurate? Partially yes	Correctly ID Change: Yes Changed Back: No	Changed to Math: Yes	Correctly Identify Error: No, just assumed faster and more numbers. Correctly Fix: Network, Game mode button

Table 5.1: Task Responses 1 – 8 Problems and Errors (continued)

7	I think this thing when I said, select the Toy to connect, and you are connected, I wanted more feedback than that. I just got a little text across the top, and I was like, Is that all I'm going to get?	Right Answer: 16 Wrong Answer: 3*** Non-attempts; 1 Can You Explain How to Play? Yes Was it accurate? Yes	Correctly ID Change: Yes Changed Back: Yes Turn it on and off, press button 3 sec.	Changed to Math: Yes	Correctly Identify Error: No Correctly Fix: On/Off
8	Interestingly enough, actually, because this is in front of me, I automatically saw that I needed to take an action here first, then the laptop. But the steps have you going to the laptop first and then this second.	Right Answer: 10 Wrong Answer: 8 Non-attempts; 2 Can You Explain How to Play? Yes Was it accurate? Yes	Correctly ID Change: Yes Changed Back: Yes Press button 3 sec.	Changed to Math: Yes	Correctly Identify Error: Yes Correctly Fix: Game Mode 3 sec.

Table 5.2: Task Responses 9 – 16 Problems and Errors * The student Stopped pressing buttons but responded verbally or indicated that he was following along. **The student Identified a fourth light in the sequence but attempted to add it together.

	Session Tasks Stage 2				
	Task 1: Set up the toy so the child can play'	**Task 2:** A round of play is 5 color sequences and 5 math questions. Play 2 rounds with the toy.'	**Task 3:** Determine any change in difficulty and then change the game back to the default level.	**Task 4:** Switch from the color game to the math game	**Task 5:** Attempt to fix the performance of the toy
	Difficulty Rank 1-easy, 5-hard	Possible Answers out of 20	Can you identify the change? Can you change it back? Could the participant explain the game? Was the explanation accurate?	Can you change to the math game?	Can you identify there is an error? Correctly Fix:
9	For this one yeah, I don't know why I didn't follow this one (pointing to instructions). I just ignored this one.	Right Answer: 18* Wrong Answer: 0 Non-attempts; 2	Correctly ID Change: Yes Changed Back: Yes Can You Explain How to Play: Not sure Was it accurate? No. I don't understand	Changed to Math: Yes	Correctly Identify Error: Yes Correctly Fix: Game Mode 3 sec.
10	Actually, it's not so hard. But I am a bit confused at first because I don't know am not sure what I am supposed to do. Like at which point am I finished I would like more feedback to follow the steps.	Right Answer: 17 Wrong Answer: 0 Non-attempts; 3	Correctly ID Change: Yes Changed Back: Yes Can You Explain How to Play? Yes Was it accurate? Yes, Not sure if I know, but I understand the color vs number indicators.	Changed to Math: Yes	Correctly Identify Error: Noticed a difference but thought it was part of the game. Correctly Fix: On/Off
11	...when I see that there's six steps even before I start reading, that looks like a lot of steps. Whereas if I have this picture, which is excellent, but if it says game mode button, press here to begin,	Right Answer: 19 Wrong Answer: 1 Non-attempts; 0	Correctly ID Change: Yes Changed Back: Yes Can You Explain How to Play? Yes. Was it accurate: Yes, for color sequence but not math	Changed to Math: Yes	Correctly Identify Error: No Correctly Fix: No (On/Off when prompted)

Table 5.2: Task Responses (continued)

12	I am a bit confused...I assume that I need to put the battery in, and then whichever is, so I assume sounds something already set up for me, and I don’t know where to jump in.	Right Answer: 5 Wrong Answer: 15 Non-attempts; 0	Correctly ID Change: Yes** Changed Back: No Can You Explain How to Play? I don’t know how. Connected the color to the number. Was it accurate? No	Changed to Math: No	*Noticed the math game Correctly Identify Error: Yes Correctly Fix: Game Mode Button 3 Sec. On/Off
13	But I can imagine somebody like typing or not knowing the password, username or password. But I guess for myself, then it was it was pretty easy.	Right Answer: 20 Wrong Answer: 0 Non-attempts; 0	Correctly ID Change: Yes Changed Back: Yes Can You Explain How to Play? Yes Was it accurate? Yes, however added prime numbers as part of the sequence which was not.	Changed to Math: Yes	Correctly Identify Error: Yes Correctly Fix: Yes Game Mode 3 sec. Network or On/Off
14		Right Answer: 20* Wrong Answer: 0 Non-attempts; 0	Correctly ID Change: No Focused on numbers repeating Changed Back: No Can You Explain How to Play: Yes Was it accurate: Yes initially ignored the math portion added later in task 4	Changed to Math: Yes	Correctly Identify Error: Noticed it was different but not an error with the toy Correctly Fix: On/Off when prompted
15	I think I’m connected. It says I’m connected and I touched the button and it went out. I don’t know if that was right.	Right Answer: 18 Wrong Answer: 2 Non-attempts; 0	Correctly ID Change: No At first then correctly identified, as well as repeated lights. Changed Back: Yes Can You Explain How to Play: NA Was it accurate: NA	Changed to Math: Yes	Correctly Identify Error: No Correctly Fix: No (On/Off when prompted)
16	Okay. But this one is not on for now. When I say that I should press this button, I suppose and it's a signal for the child to be able to put it. Okay. I think it's at half-stages. Finished.	Right Answer: 19 Wrong Answer: 1 Non-attempts; 0	Correctly ID Change: Yes Changed Back: Yes Can You Explain How to Play: Yes Was it accurate? Yes but followed light sequence did not explain the addition aspect	Changed to Math: Yes	Correctly Identify Error: Yes Correctly Fix: On/Off then network.

The startup sequence (task 1) requires the participant to turn on the toy and wait for the indicator to tell them it is connected to the internet. Once this is complete, the participant is required to start the game by selecting the central game mode button. The color sequence game will begin immediately following this action. The gameplay consisted of following five color sequences and repeating them in order. After the color sequences, the game switches to simple math questions and answers, which must be repeated for a correct response. The difference between the two sequences is merely colors vs. numbers; however, the central game mode button informs the user it is asking for math answers by displaying a ‘+’ sign. The indicator button does not need to be pressed in order for the sequence to be answered correctly. For example, if 2 and 5 are displayed, followed by a 7, the participant only needs to press 2, 5, and 7, respectively. Because the '+' sign is lit, participants may wish to include it in the sequence (2, +, 5, and 7). See task descriptions in the section below.

For task 2, the mean score for each group was 17 out of a possible 20 correct responses in both the color and math games. Participant 12, who received the lowest score, added all three numbers together during the math game and responded with a two-digit total. The sequence provided was "+, 3, 5, 8," and they responded by pressing buttons "1, 6," assuming the answer was 16. Where this makes logical sense, it assumes the child understands how to add and add three numbers together. Seven of the 16 participants had no incorrect answers, 5 others were incorrect only once, and 1 had two incorrect answers. Two participants had 8 and 15 incorrect responses due to their misunderstanding of the math game; however, this would account for the ten math sequences each player received. The participant who scored 8 only understood that the numbers should be added together for the last two math questions. The participant with 15 incorrect responses (participant 12) continued to use the strategy of adding all three numbers together, even during the color sequence game. Although this error was only made by 1 participant and was not an issue that warranted an adjustment to the game, they were not the only players to follow the numbers on the buttons during the color game. Although players were aware of the math game and how it worked, instead of referring to the color during the color game, they would often recite the number they were pushing. It indicates how powerfully the symbol of numbers, as opposed to colors, is ingrained in the minds of adults. "I know it's red, but I prefer the number 5. It's easier." participant 7. Two participants identified this preference and consciously included the color in their verbal response, such as red 5, blue 3, or orange 9.

Participants found Task 3 easy to identify the problem with only one person unable to notice the change in difficulty level and could not change it back to the default level. Only one other participant was unable to change back to the default level. Similarly, task 4 only had one participant unable to complete the task, who also ranked it as the most difficult. All other participants were able to complete task four with two people, indicating a 3 (neither easy nor hard) for the level of difficulty in doing the task.

"Can you explain how to play with the toy as if you were describing it to a child?"

All participants were asked to describe how the toy works as if they were explaining it to a child. For the second group, this question was asked after the third task, while the first group was asked after the second task. The question order was changed to give participants more time to familiarize themselves with the game. After the 2nd task, the participants only played two rounds of the game. Of the 16 participants (both groups), 15 responded to the question with an answer. However, the second group needed to be able to describe the difference between the games accurately or add details absent or unrelated to the game. The only other modifications were software changes previously described.

According to participants, task 5 was the second most challenging task to complete. Four users could not complete the task, and two more struggled to find the answer in the guide or find the button to turn it off and on but managed to complete the task. However, one of the four users who did not identify an issue with the toy did indicate the correct response as a matter of 'standard procedure.' Turning any electronic device off and on again "usually fixes the problem," according to three participants.

Generally, the participant's description of the games and toy highlights one of the main findings of the usability test. In group one, 6 of 8 responded correctly, and one correctly answered but could not distinguish between the math and color games. In contrast, only 2 out of 7 who answered in the second group could fully describe how the games worked. Three of the seven could describe it correctly but could not distinguish between the color sequence and the math sequence game. Just over half (8/15) understood the toy well enough to explain it to a child, with 4 participants unable to describe the color sequences from the math sequence. Surprisingly, the more time spent with the toy, the less clear the math game was for the participants. Other tasks reveal the same problem with the math and color game distinction. Often, participants requested more feedback from the toy to indicate where they were in the game. "It would have been useful to tell me that the reaction to pushing the game mode button."

Completion Rates and Difficulty Rating

Table 5.3: Completion Rate – 1 completed, 2 completed with issues or delay, 3 did not complete. Difficulty Rating – 1 the task was very easy, 2 easy, 3 neither easy or hard, 4 difficult, 5 very difficult.

	Stages 1 and 2									
	Task 1		**Task 2**		**Task 3**		**Task 4**		**Task 5**	
	Completion 2 > 5 min.									
	Comp.	**Diff.**	**Comp.**	**Diff.**	**Comp.**	**Diff.**	**Comp.**	**Diff.**	**Comp.**	**Diff.**
1	2	2	1	1	ID 1 BK N/A	1	N/A	N/A	ID 1 ST IO	1
2	2	3	1	N/A	ID 1 BK N/A	2	N/A	N/A	ID 1 ST RG	1
3	1	1	1	2	ID 1 BK N/A	1	N/A	N/A	ID 1 ST IO	1
4	1	2	1	3	ID 1 BK N/A	2 – 2.5	1	1	ID 1 ST IO	1
5	2	1	1	3	ID 1 BK N/A	1	1	1	ID 1 ST IO	1
6	1	1	1	3	ID 1 BK 3	2	1	2	ID 3 ST 3	2
7	1	2	1	5	ID 1 BK 1	1	1	1	ID 3 ST IO*	1
8	1	2	2	1	ID 1 BK 1	1	1	1	ID 1 ST GM	1
9	2	5	1	4	ID 1 BK 1	1	1	3	ID 1 ST GM	1
10	1	2	1	3	ID 1 BK 1	1	1	2	ID 2 ST IO	4
11	1	2	1	2	ID 1 BK 1	1	1	3	ID 3 ST 3	5
12	2	1	2	2	ID 1 BK 1	3	3	5	ID 1 ST GM	2
13	1	1	1	1	ID 1 BK 1	1	1	1	ID 1 ST GM	1
14	1	1	1	3	ID 3 BK 3	2	1	1	ID 2 ST IO	4
15	1	3	1	2	ID 2 BK 1	2	1	1	ID 3 ST 3	5
16	2	2	1	1	ID 1 BK 1	2	1	1	ID 1 ST IO	2

Tasks 3 and 5 had two parts: identify (**ID**) the error and change it back (**BK**) to the toy default in Task 3. Task 5 lists their strategy (**ST**) for how they would change the toy: **IO—**turning it on and off, **GM—**pushing the game mode button, and **RG—**checking the reference guide. Participants indicated referencing the guide as well but not as a first choice.

Completion Rates: Task 1: 100%, 2: 100%, 3: ID**: 93%** BK **82%, 4: 92%, 5:** ID **75%** ST **81%**
Difficulty Rating Average: Task 1: 1.93, 2: 2.4, 3: 1.5 – 1.53, 4: 1.76, 5: 2.06

The most challenging task, according to the participant's ranking, was task 2, with an average rating of 2.4 out of 5. This is due to the introduction of the game and toy interaction, the need for feedback concerning the different games, and the game mode button previously discussed.

The fifth task ranked second most challenging, with a value of 2.06 out of 5. The toy was made to operate chaotically with too many lights flashing out of sequence. This rating would be higher for troubleshooting electronic devices if they are spotted. This rating was not as high as it might have been, considering 6 of the 16 did not identify a problem with the toy and assumed it was 'just the way the toy worked' according to one participant. Changes made to the gameplay were drastic for each participant, including speeding up the duration the lights blinked on and off from 600 milliseconds to less than 200 and prolonging the delay between blinking from 600 milliseconds to 1200 or more. Color and math game lights were intermixed, and five or more lights were part of the sequence. The toy's strange behavior indicated that people will tolerate malfunctions or assume electronics do not always function as expected.

Participants were prompted to play more rounds of the game. They were unaware this was part of the next task. It was given to them only when they noted a problem with the game, and they were asked to repair the game. This technique of allowing users to interact with the game and interjecting the task was effective in getting users to reflect immediately on what they witnessed and rewarded their assumption that the game's unusual behavior was correct. Of the six users who did not identify a problem with the game, four ranked the problem difficulty at 4 or 5. One of these participants ranked it a 2. The highest ranking of difficulty for task 5 was in the first group of eight. The other participant ranked the difficulty level as 1 out of 5, explaining: I don't know if what I did worked or not, but rechecking the connection would be 1 [extremely easy]. They are focused on the functional engagement of the interaction and, therefore, found it easy. Rechecking the connection is easy, but if we take the task to include identifying the problem and thinking of a fix, they may have increased the ranking.

Table 5.4: task times during the usability test and time stamp location on the video recording.
*After task 2, the first 8 participants were asked to explain the game; the same was asked of the last 8 participants after task 3.
**Completed task but not correctly. The participant perceived their own solution.
*** Did not recognize the problem within the task.

Task Time

	Stages 1 and 2			🟦 Slowest Time	🟨 Fastest Time
	Task 1	**Task 2**	**Task 3**	**Task 4**	**Task 5**
	Duration **Time Stamp**	**Duration** *first 8 descriptions of the game not included	**Duration** *last 8 descriptions of the game not included	**Duration** Changes made to toy after first 8.	**Duration** Time started when problem was recognized.
1	Time: 7 min 48 sec. Timestamp 4:21 - 11:59	Time: 8 min 25 sec. Timestamp 14:00 - 22:25	Time: 1 min 48 sec. Timestamp 25:53 - 27:41	N/A	Time: 2 min 45 sec. Timestamp 29:07 - 31:52
2	Time: 7 min 20 sec. Timestamp 5:20 - 12:40	Time: 7 min 20 sec. Timestamp 5:20 - 12:40	Time: 0 min 0 sec. Timestamp 15:55 - 23:20	N/A	Time: 1 min 48 sec. Timestamp 25:53 - 27:41

Table 5.4: task times during the usability test (continued)

3	Time: 3 min 18 sec. Timestamp 5:17 - 8:35	Time: 2 min 9 sec. Timestamp 9:36 - 11:45	Time: 1 min 49 sec. Timestamp 13:22 - 15:11	N/A	Time: 2 min 2 sec. Timestamp 17:37 - 18:39
4	Time: 4 min 41 sec. Timestamp 4:50 - 9:31	Time: 3 min 24 sec. Timestamp 11:31 - 14:55	Time: 1 min 44 sec. Timestamp 19:24 - 20:58	Time: 4 min 22 sec. Timestamp 25:17 - 29:39	Time: 2 min 47 sec. Timestamp 31:30 - 33:17
5	Time: 5 min 5 sec. Timestamp 5:37 - 10:42	Time: 9 min 42 sec. Timestamp 13:22 - 23:04	Time: 3 min 15 sec. Timestamp 28:44 - 31:59	Time: 2 min 46 sec. Timestamp 35:44 - 38:30	Time: 1 min 45 sec. Timestamp 42:24 - 44:09
6	Time: 2 min 13 sec. Timestamp 4:31 - 6:44	Time: 3 min 58 sec. Timestamp 9:20 - 13:18	Time: 0 min 41 sec. Timestamp 16:23 - 17:04	Time: 2 min 39 sec. Timestamp 19:43 - 22:22	Time: 2 min 28 sec. ***Timestamp 24:33 - 27:01
7	Time: 4 min 26 sec. Timestamp 1:34 - 6:00	Time: 3 min 27 sec. Timestamp 7:03 - 10:30	Time: 0 min 40 sec. Timestamp 12:06 - 12:46	Time: 1 min 20 sec. Timestamp 13:10 - 14:30	Time: 0 min 55 sec. Timestamp 22:24 - 23:19
8	Time: 3 min 45 sec. Timestamp 4:19 - 8:04	Time: 3 min 37 sec. Timestamp 10:01 - 13:38	Time: 1 min 21 sec. **Timestamp 15:37 - 16:58	Time: 1 min 2 sec. Timestamp 18:49 - 19:51	Time: 1 min 27 sec. Timestamp 25:35 - 27:02
9	Time: 7 min 13 sec. Timestamp 3:43 - 10:56	Time: 7 min 43 sec. Timestamp 13:10 - 20:53	Time:1 min 8 sec. Timestamp 23:03 - 24:11	Time: 0 min 16 sec. Timestamp 29:08 - 29:24	Time: 1 min 06 sec. Timestamp 35:09 - 36:57
10	Time: 2 min 48 sec. Timestamp 4:31 - 7:19	Time: 7 min 20 sec. Timestamp 9:11 - 15:10	Time: 0 min 33 sec. Timestamp 17:04 - 17:37	Time: 0 min 38 sec. Timestamp 21:53 - 22:31	Time: 2 min 11 sec. Timestamp 30:58 - 33:09
11	Time: 2 min 04 sec. Timestamp 3:43 - 5:47	Time: 4 min 12 sec. Timestamp 8:17 - 12:29	Time: 0 min 55 sec. Timestamp 13:57 - 14:52	Time: 0 min 24 sec. Timestamp 17:57 - 18:21	Time: 1 min 18 sec. ***Timestamp 24:59 - 26:17
12	Time: 6 min 01 sec. Timestamp 5:57 - 11:58	Time: 5 min 8 sec. Timestamp 17:11 - 22:19	Time: 2 min 34 sec. Timestamp 24:38 - 27:12	Time: 1 min 27 sec. Timestamp 36:48 - 38:15	Time: 3 min 52 sec. Timestamp 42:28 - 46:20
13	Time: 2 min 19 sec. Timestamp 2:08 – 4:27	Time: 2 min 50 sec. Timestamp 5:54 - 8:44	Time: 0 min 45 sec. Timestamp 10:36 - 11:21	Time: 0 min 46 sec. Timestamp 15:13 - 15:59	Time: 1 min 15 sec. ***Timestamp 20:38 - 21:53
14	Time: 2 min 32 sec. Timestamp 5:01 – 7:33	Time: 3 min 29 sec. Timestamp 8:50 - 12:19	Time: 1 min 04 sec. Timestamp 13:42 - 14:46	**Time: 0 min 48 sec. Timestamp 17:09 - 17:57	Time: 2 min 13 sec. ***Timestamp 21:26 - 23:39
15	Time: 4 min 07 sec. Timestamp 5:06 – 9:13	Time: 2 min 49 sec. Timestamp 11:58 - 14:47	Time: 0 min 41 sec. Timestamp 17:09 - 17:50	Time: 0 min 7 sec. Timestamp 23:59 - 24:06	Time: 1 min 37 sec. Timestamp 30:19 - 31:56
16	Time: 6 min 15 sec. Timestamp: 4:59 - 11:14	Time: 4 min 56 sec. Timestamp 15:04 - 20:00	Time: 1 min 08 sec. Timestamp 23:10 - 24:18	Time: 0 min 45 sec. Timestamp 29:04 - 29:49	Time: 3 min 35 sec. Timestamp 32:20 - 35:55

The duration of the first task varied due to the type of device the participant was using to log onto the site and connect with WiFi in the facilities where testing took place. All participants used their own devices and were familiar with the platform and functionality. The quickest time was 2 minutes and 3 seconds, which is fast for a 6-step process. The slowest completion rate was 7 minutes and 48 seconds, which is very long, although two other participants also took more than 7 minutes to complete the task. Six of the 16 participants took more than five minutes to complete this task, indicating it was laborious. A new strategy for connecting the toy to the internet needs to be developed so that setting up the toy doesn't stand in the way of playing. The average difficulty rank was only 1.93 out of 5, which suggests that participants did not find the task difficult but rather time-consuming, which is supported by a 100% completion rate even though three of the 10 participants who ranked it above 1 ranked the difficulty level 3 or more – not to mention the negative comments.

The overall duration for gameplay is similar across the tasks that require the user to interact with the toy as it was meant to be played. We would expect, however, the duration to drop, particularly between tasks 2 and 3, because in Task 2, the games are novel to the users, and Task 2 requires two rounds of both the color and math sequences for a total of four games. In contrast, Task 3 only requires one game – a color sequence. Overall times were recorded, as well as the time to complete each task in the timetable (Table 5.4). We would expect Task 2 to be at least four times the duration of Task 3, which was correct: on average, Task 2 was completed in 5 minutes 1 second and Task 3 in 1 minute 20 seconds. Taken individually, the games took a similar amount of time: for Task 2, the average was 1 minute 27 seconds per game, and for Task 3, it was 1 minute 20 seconds per game.

Discussion of Changes and Implications

Prioritizing the Problems – Iterative Design Process

The changes made to the toy were prioritized using IDEO's Feasibility, Desirability, and Viability scorecard, which was modified by the Neilson and Norman Group priority matrix to split out cost and fixability vs. the impact on users. The scale provides 15 possible points for each issue and a guide to what will affect the user.

- Feasibility: the degree to which the item can be technically built. Do the skillset and expertise exist to create this solution?
- Desirability: how much users want the item. What unique value proposition does it provide? Is the solution fundamentally needed, or are users otherwise able to accomplish their goals?
- Viability: if the item is functionally attainable for the business. Does pursuing the item benefit the business? What are the costs to the business, and is the solution sustainable over time?

Table 5.5—Each issue found during the usability test is given a weighted priority. This method is often used to determine priority fixes but also offers insight into the importance of 'impact' over the balance of 'Feasibility' vs. 'Viability.'

	How Feasible – Easy vs. Difficult	How Desirable -– Impact on Usability	How Viable -– Time and/or Expense	Weighted Priority
Usability Issues	**1 Easy – 5 Difficult**	**1 Low – 5 High**	**1 Inexpensive – 5 Costly**	**3 – 15 Scale**
Right / Wrong Feedback	3	5	3	11
Level of Difficulty Indicator	5	2	4	11
Connection Instructions	4	4	2-3	10-11
Game Indicator	3	4	3	10
+ = game mode button	3	2	4	9
Handle Use	1	3	2	6
Light Variation	2	2	1	5

During the usability test, the level of gameplay did not impact the players or the number of rounds they played. The second task did increase from level one (3 colors in a sequence and two numbers added together along with the answer for the math sequence) to level two (4 colors in a sequence). Participants were asked to indicate the change and if that change was clearly understood. Fourteen participants noted it was clear, and one thought they should add them together but indicated that four numbers were more difficult. Two did not notice the change in difficulty. These results suggest the indication through gameplay needs to be more significant, and modifying the toy or adding a level indicator might be necessary. What was clear from participants is that difficulty had less to do with the number of digits and more with the cognitive effort of the math question being asked.

For this reason, the difficulty level between the two games does not change at the same rate. For example, level one uses three colors and three numbers as equivalent button presses. Increasing the number of colors would increase along with the number of digits being added together. We can see the pedagogical difference in this scenario and the practical issues that arise. It was important to teach math using single-digit examples throughout the levels (Kamii et al., 2001). Mostly, this is restricted by the number represented on the button. Therefore, the highest possible answer is 9. At level 5, the toy adds three digits, limiting the possible questions to 220. Questions can use duplicate numbers, as well as reverse order of numbers as unique questions. However, if addition and subtraction are used together, values larger than 9 are part of the equation, although not the final result (e.g., 6+8-5=9, where 6 plus 8 equals 14). It is arguably more difficult to use 3 or more digits than just using a single operator (Kamii et al., 2001). According to Mauhibah and Karso (2020), the cognitive ordering of simple math questions by elementary school teachers is: adding 2-digit numbers, subtracting 2 digits, adding 3 digits, and adding and subtracting 3 digits together. Although the specific level of difficulty and its corresponding color sequence are beyond the scope of the usability test and this study, it does alter how to frame the research question of transference. To retain equivalency between games, the number of pressed buttons was less critical than cognitive difficulty. The following game levels were programmed into the game.

Table 5.6 – Game Level of Difficulty equivalency of color and math sequence games.

Game Level of Difficulty	Color Sequence	Math Sequence
Level 1	3	
Level 2		Add 2 digits equals 3rd
Level 3	4	
Level 4		Subtract 2 digits equals 3rd
Level 5	5	
Level 6		Add 3 digits equals 4th
Level 7	6	
Level 8		Add and/or Subtract 3 digits equals 4th

The variety of colors available using single hue 5mm mini-LED bulb was limited to 8 when constructing the paper prototype. A second blue light was added to fill in the 9th color. The purple light tended to blink on – which confused the participants – however, their responses were not included in the error totals due to this anomaly. The design of the paper prototype buttons added a clear bubble with the digit printed on the top in the center, where the color light was set outside the bubble on the rim flush with the button.

Although each button was visible to the participants, five misinterpreted the white lights under the bubble and the color lights around the button's rim. All the participants followed the colors of the first sequence, and 14 understood that the math game followed. The remaining two assumed the light was enough of an indication and followed the sequence of lights. Three others who understood that the lights indicated numbers did not know that the sequence consisted of the first two numbers adding up to the third. As a result, they added all three numbers together and, rather than push the numbers in sequence, provided the total, typically a two-digit number. The order of math light sequences was (+ AB) C), where the plus sign stayed on versus (A+B=C). Some participants suggested just providing the first two digits and letting the player add them together. This change would have altered the general concept of gameplay to follow the sequence, and it is determined to be too difficult for this audience that is beginning to learn math. Playing the sequence using the central lights to display '+' and '=' will make the simple addition clearer but also demonstrate the toy is providing the answer along with the question. Changes to the next version of the toy will include this central indicator and more lights to indicate the colors, separate from a single white light to indicate the number.

Change made from group 1 to group 2 in task 4 – a clear improvement
Each participant was asked to explain the toy to a child. This task was done after Task 2 for Group 1 and after Task 3 for Group 2. Participants in both groups provided an explanation phrased as if to a child within the target audience. Group 1 played both math and color sequence games but only twice. All eight participants in this group indicated they understood the games and were able to explain them correctly. Of the eight participants in group 2 who had the advantage of playing one more round of the color sequences, only 6 indicated they could explain it, and only four could do this correctly. Although not definitive, the second group, which played an additional round, had more difficulty than group 1 partly because the task was more difficult. Two participants expressed the need for concentrating more during task 3, "...wow! I need to pay attention" and "now I have to concentrate," while others gestured by sitting up or focusing on the toy – indicating they were more attentive. The same participants in group 2 who misunderstood the math game also indicated a need for feedback after each sequence and their attempts to add all three numbers together as previously described.

5-ness: considering the representation of numbers

For this reason, the dots' or pips' orientation is the same for numbers 1 through 6 on a dice, and 7, 8, and 9 replicate the orientation of domino tiles. The pips are also raised to distinguish the dots on each button physically. The design was made to provide a tactile quality to the button top and to assure the students could participate and accomplish the goal of the math game. Because students can count the dots in order to find the answer to the math questions, physical activity eases emotional anxiety or fear when playing. This also alleviates the issue of semiotic understanding of numeric symbols. Although subtle, the impression of the number is less important to the educational contribution the dots make than the inclusive nature of helping the children who have not mastered their numbers.

Design for User Satisfaction

The paper prototype tested adults who are typical caregivers – parents, therapists, and teachers – who would assist children with autism. Connecting the toy to the internet and preparing it for play was tested to determine the ease with which this can be done. Using a smartphone to set network preferences and pair the toy was a six-step process. Each participant was asked to locate a website, select the WiFi network closest to them, submit credentials to connect, pair the toy, and connect it to the network. This process was completed by 14 of the 16 participants, which is not a severe problem, but it does represent a problem for children who should be able to set up the toy themselves. A smartphone or other computer device is necessary because the toy is not equipped with a screen. That said, pairing devices has become more common since the advent of smart speakers, earbuds, and other wireless devices.

Conclusion

Usability testing often uncovers technical issues that are known even before testing, but these issues are more pronounced because of the limitations of paper prototypes. Confirming the problem is still worth noting and can help get information from participants. The first of these known issues and the most common (14 of 16 participants) comment was a need for more user feedback. Five participants proposed a sound indicating the user answered the question correctly or incorrectly, and two suggested a flash of all the lights on the toy, changing the pattern for correct and incorrect answers. Both sound and lights need to be considered with care because some children with autism dislike sharp or shrill noises or flashing lights. To technically accommodate light feedback on future toys, all nine lights will fade on and hold for 3 seconds, then fade off for a correct answer, and individual lights will fade on and off for 1 second each. The second major issue found in the user study was the function of the central button. Changes were made to the process of connecting the toy to the internet to remove the need to push the button. The center light must only indicate the '+,' '-,' and '=' operands, taking them out of the sequence. If players press the center light, it will not change their answer; however, it may add fractions of time to their response. The design of the light is flush mounted to the toy body and significantly smaller than the buttons. These design changes intend to lessen the importance of the center light and, at the same time, reinforce the sequence concept of pushing buttons in both games.

Lastly, times during play will be compared and used based on the machine learning training set for each difficulty level. Compensating for the difficulty level is not easily accomplished based on conditional statements of more traditional forms of programming. For example, we see a range of speeds for Task 2, which averages 1 minute 27 sec, and Task 3, which is 1 minute 20 sec. If any time estimate between these two tasks was made, one might guess the more difficult task would take longer. Statistically, they are the same; on average, the easier task takes slightly more time. However, this does not consider novelty (the newer the player is to the game, the slower they will be) or education level (the more a student knows, the faster they answer).

Chapter 6 - Experimental Research Design Overview

Using data collected from the paper prototype, design changes and interactions with the toy were refined and implemented in the next phase of the design. User interaction data continued to be captured with results reported in the previous chapter. Evaluating the toy's performance provides a basis for selecting a predictive approach over a predetermined one and insight into the learning process. The following experimental research studies compare two strategic approaches to developing digital educational games: a tree-structured approach and machine learning. The study examines the learning experience to address criticism of Goosen and Wabash (2004), who identify three critical shortcomings of earlier studies which they excluded from their meta-analysis because "none assessed experiential exercises in terms of (a) objective learning measures, (b) an observable behavior, and/or (c) the use of a pre-post or control group design." This study addresses these shortcomings by collecting, analyzing, and evaluating the game's computational performance and assessing the impact of the design and technology based on observation of the participants and a pre-test study.

Observation of the Participants

Participant eligibility

A checklist was developed to determine the participants' eligibility for the screening criteria, including verification of the child's autism diagnosis, their current education level (specifically in math and color), and their ages between 6 and 10. In addition to these criteria, parents or teachers provided information concerning the participants' performance of these pre-requisite skills, such as (a) following written and oral instructions, (b) counting rhythmically one by one and two by two up to 20, (c) matching and writing numbers between 1 and 20, (d) recognizing the addition sign, (e) counting pictures of objects using count-all strategy and telling the total, (f) count-all from the largest number, and (g) having the skills to count the dots on the numbers prepared. (Calik and Kargin, 2010) In addition to verification from the Hope Academy, a school for autistic children, and discussions with administrators and staff, a pre-test was taken one day before the sessions.

Computational Performance Comparison

Separate evaluations of each software's performance are done in the following chapters. However, we also compare the performance of the predetermined program for user input versus the adaptive perspectives of machine learning. One could argue that depending on the level of granularity in the decision tree software, the reactions of machine learning can be simulated. After all, there are only three possible 'decisions' for the machine learning algorithm to make: increase the difficulty, decrease the difficulty, or remain the same. If, for example, software developed using a typical decision tree measures pressure values read from a button sensor. The sensor could have multiple ranges to emulate a type of weighting system; we could portend 10 out of 100 possible values to base this decision. For a possible nine values, one could also imagine the toy's orientation using X, Y, and Z coordinates of 0 degrees, > 30 degrees, and > 45 degrees. Three categories indicate the speed to answer: slow, moderate, and fast times. Lastly, the game's challenge level being played out of a possible five is also quantifiable. We can see how this modest amount of granularity would make the decision tree incredibly complex, challenging to maintain, and costly in terms of processing power, which could slow the game and, in turn, lessen the user experience.

A standard multivariate research design is a 5-factorial design (see Table 6.1); however, the complexity and management of these many variables becomes untenable. Using the same values described above, pressure, orientation, speed, and 10 x 9 x 3, respectively, requires 270 possible conditions multiplied by the 3 to understand the effects on each dependent variable.

There are three possible outcomes to assign the question: easier, the same level of difficulty, and a harder question. log-linear analysis.

Table 6.1: Factorial Research Design of independent variables that affect dependent variables to challenge the user’s ability and keep them engaged.

		Independent Variables				Dependent Variables	
		Pressure	Orientation	Speed	Question Difficulty	Sequence Difficulty	Performance
Sessions	1+	Group 1			easier	1 - 5	0 – Wrong 1 – Right
					stay the same		
					harder		
	5+	Group 2			easier	1 - 5	0 – Wrong 1 – Right
					stay the same		
					harder		

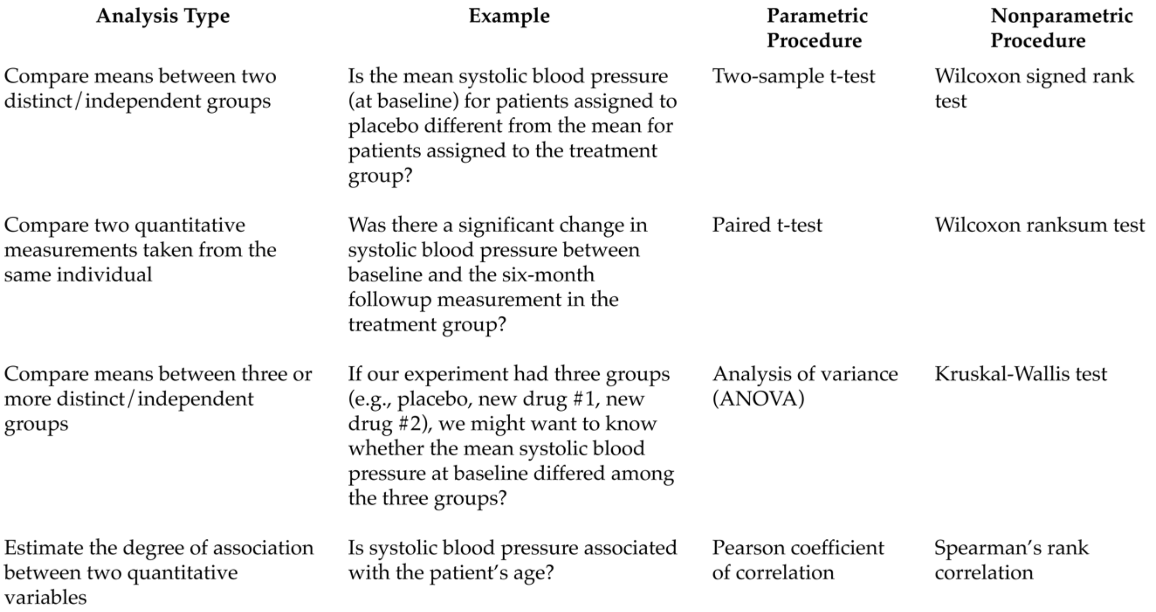
A Determination of Distribution – parametric or non-parametric

Selecting multivariate numerical variance analysis (MANOVA) has the advantage of assessing the relationship between variables. The requirements are the same for ANOVA with some additions. Although MANOVA appears to fit the criteria for analysis, the dependent variables must also be binary. The player's performance will meet this definition; the answer is either correct or incorrect. What makes the design more complex is when this variable is used to inform the prediction of the toy. The player performance becomes an independent variable, and the prediction is no longer a binary but rather an ordinal variable - 'easy,' 'stay,' or 'hard.'

"It is well known that the general class of analysis of variance (ANOVA) tools frequently applied by educational researchers…, include at least three key distributional assumptions. For all cases, the outcome measure Y*ki* (or score) associated with the *ith* individual within the *kth* group has normality and is independently distributed, with a mean of *u* and a variance of o^2" (Keselman et al., 1998)

To use MANOVA, validation methods, and confidence tests are employed to affirm the statistical significance of the results. Before we can select a statistical method, the assumptions about the data need to be understood. "Concretely, what this means is that an assumptions-violated test of group effects might yield an F ratio with a corresponding significance probability of p = .04, which (based on an a priori Type I error probability of .05) would lead a researcher to conclude that there are statistically non-chance differences among the K groups" If these assumptions are set aside interpretations are not on a solid footing. In an effort to be thorough, an additional method for analysis will include a multivariate multiple regression (MMR).

Table 6.2 Determination of Research Design (Hoskin, 2012)



Analysis Type	**Example**	**Parametric Procedure**	**Nonparametric Procedure**
Compare means between two distinct/independent groups	Is the mean systolic blood pressure (at baseline) for patients assigned to placebo different from the mean for patients assigned to the treatment group?	Two-sample t-test	Wilcoxon signed rank test
Compare two quantitative measurements taken from the same individual	Was there a significant change in systolic blood pressure between baseline and the six-month followup measurement in the treatment group?	Paired t-test	Wilcoxon ranksum test
Compare means between three or more distinct/independent groups	If our experiment had three groups (e.g., placebo, new drug #1, new drug #2), we might want to know whether the mean systolic blood pressure at baseline differed among the three groups?	Analysis of variance (ANOVA)	Kruskal-Wallis test
Estimate the degree of association between two quantitative variables	Is systolic blood pressure associated with the patient's age?	Pearson coefficient of correlation	Spearman's rank correlation

Interaction Between Independent Variables

The results from the statistical software provided a level of difficulty for the following questions.

The assumptions for Log-Linear Analysis include:

1. Random Sample
2. Independence
3. Mutually exclusive groups

The first and third assumptions have been met and discussed in earlier chapters. The measure of independents is done using Pearson's correlation coefficient in Chapter 8.

Justification for log-linear analysis:

1. To test the difference between two or more variables
2. Variable of interest is proportional or categorical
3. Two or more options

The variables of interest are determined at two stages of the study. The first is the performance-dependent variable (right vs wrong) relative to speed, motion, difficulty, duration, and pressure. This analysis tests the child's performance but depends on the toy's prediction performance. The second dependent variable is the prediction accuracy obtained by the same input variables, including performance. The complexity of this study is evident from the number of independent and dependent variables, notwithstanding the change from player performance to toy performance as a variable of interest.

There are two considerations in determining correlation between independent variables. The first is the impact they have through the analysis of significance. Using data from the TS and ML groups there is significant difference in all the independent variables where the tree structured data only indicates difficulty, duration, and performance are significant, but motion (p=0.5365) and pressure (p=0.0.2382) are not. The second consideration is the correlation between each other, this is tested with Pearson's corollary coefficient. The coefficients indicate there is a relationship between the independent variables however weak and negative in direction. There should be some overlap between them because we expect some relationship exists due to their existence related to playing the game. The negative direction indicates they are inversely related in that when one increases like difficulty then player performance goes down. Although this analysis does indicate the type of relationship it does not determine dominance of variable. Using the same example if player performance goes up then difficulty goes down. This seems counter-intuitive but Pearson’s test does not eliminate this possibility. A more detailed description and other validation tests are applied to determine the relational dominance are found in Chapter 8.

Chapter 7 - Data Collection

The methods for data collection supports which next set of questions to ask, making the process of collection critical to the performance of the toy. Collecting here describes the TS sessions with the participants and preparing it for training the ML model. Data is collected at one second intervals identifying any changes from any of the inputs. All independent variables and dependent outcomes are stored in real-time in a connected database, automatically setting an identification number for each child. Data is then shaped to expedite cleaning and organizing for cluster analysis and expedite the speed of gameplay. Figure 7.1 shows that the data is noisy but has a structure that divides into chunks based on the player, sequence (a prediction made after 5 questions), and each answer (Figure 7.1, performance).

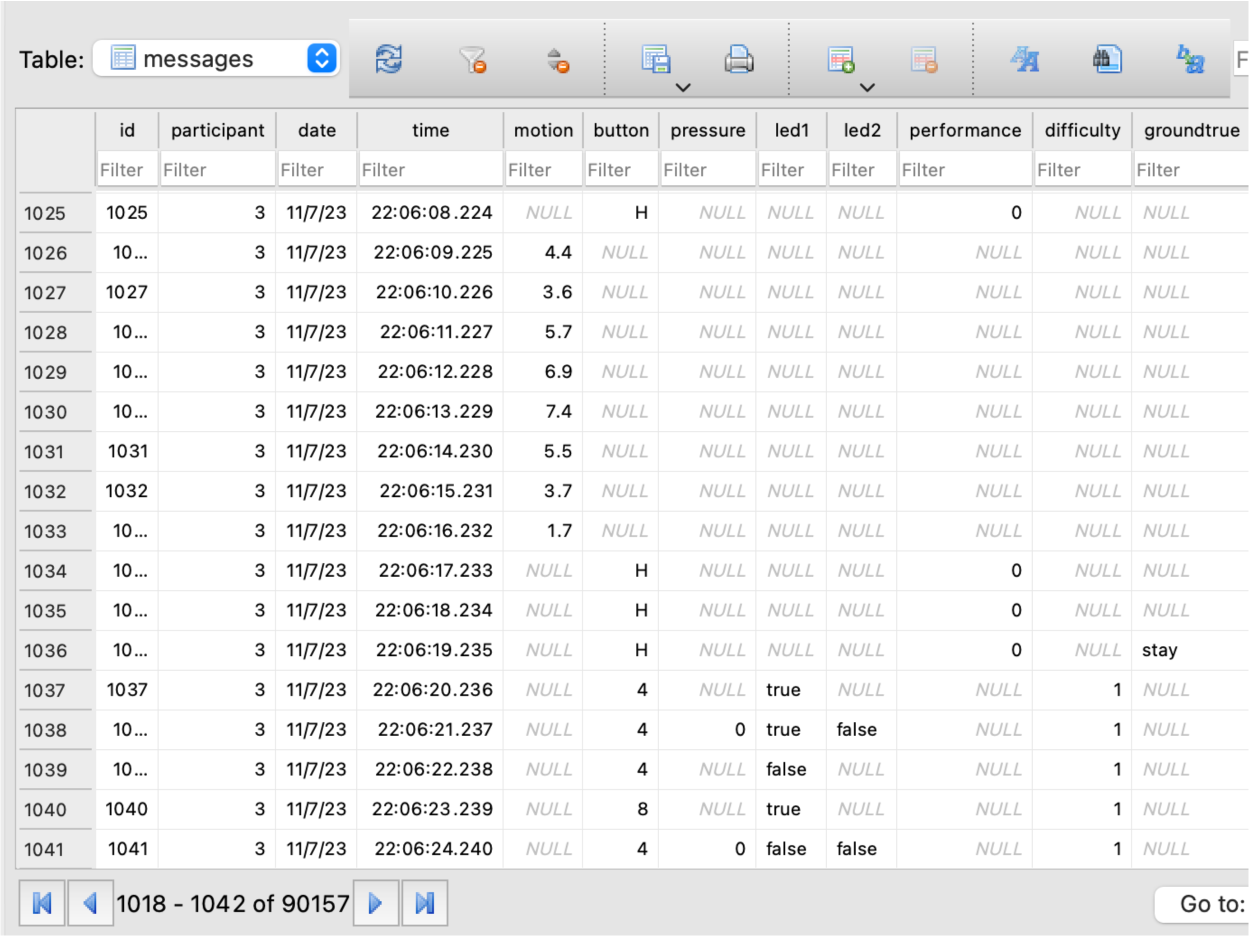


Table: messages

	id	participant	date	time	motion	button	pressure	led1	led2	performance	difficulty	groundtrue
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1025	1025	3	11/7/23	22:06:08.224	NULL	H	NULL	NULL	NULL	0	NULL	NULL
1026	10...	3	11/7/23	22:06:09.225	4.4	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1027	1027	3	11/7/23	22:06:10.226	3.6	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1028	10...	3	11/7/23	22:06:11.227	5.7	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1029	10...	3	11/7/23	22:06:12.228	6.9	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1030	10...	3	11/7/23	22:06:13.229	7.4	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1031	1031	3	11/7/23	22:06:14.230	5.5	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1032	1032	3	11/7/23	22:06:15.231	3.7	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1033	10...	3	11/7/23	22:06:16.232	1.7	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1034	10...	3	11/7/23	22:06:17.233	NULL	H	NULL	NULL	NULL	0	NULL	NULL
1035	10...	3	11/7/23	22:06:18.234	NULL	H	NULL	NULL	NULL	0	NULL	NULL
1036	10...	3	11/7/23	22:06:19.235	NULL	H	NULL	NULL	NULL	0	NULL	stay
1037	1037	3	11/7/23	22:06:20.236	NULL	4	NULL	true	NULL	NULL	1	NULL
1038	10...	3	11/7/23	22:06:21.237	NULL	4	0	true	false	NULL	1	NULL
1039	10...	3	11/7/23	22:06:22.238	NULL	4	NULL	false	NULL	NULL	1	NULL
1040	1040	3	11/7/23	22:06:23.239	NULL	8	NULL	true	NULL	NULL	1	NULL
1041	1041	3	11/7/23	22:06:24.240	NULL	4	0	false	false	NULL	1	NULL

1018 - 1042 of 90157

Figure 7.1 Database from Tree Structured study used for training and testing Machine Learning model.

The importance of sequential order is worthy of note because each participating player behaves differently – particularly at the start and end of their time with the game. Often, players miss the first question because it is their first encounter with the game or settling in. These questions were removed from the data set and instead acted as a demonstration for the game. When a player determined they wanted to stop playing, it would typically be in the middle of a sequence during the game. The questions within these first sequences were used to analyze the data for patterns of play but not in the training set for predictions. Other modifications from the raw data were lengthy periods between each player when some motion sensor data was collected from the accelerometer. Excessive motions were identified as movement in the floor, a threshold for noise was set to exclude values below 0.3 from the entire data set. This only effected the motion data but not the duration. For each question, the number of data rows varied, particularly for more complex questions. Values above motion thresholds were included and added to the duration of these questions however small. The time scale is evident in the performance/duration plot in Figure 7.3.

The total number of player interactions represented by each row in Table 7.1 in the training data is 90,517. These interactions are associated with 1001 questions (excluding first questions), making up 212 sequences. The training data for the model was collected from the n=16 participants who played the tree-structured game.

Table 7.1 Tree Structured Group Player Performance Totals

TS Group	Questions	Sequences			
		Totals	Easy	Stay	Hard
Right	496	109	12	51	46
Wrong	505	103	44	54	5
Total	1001	212	56	105	51

Cleaning, preprocessing, and manual labeling were done according to the constraints of the machine-learning version of the game, following best practices for the highest model performance. The test set of data is not included in these totals.

Table 7.2 Machine Learning Group Player Performance Totals

ML Group	Questions	Sequences			
		Totals	Easy	Stay	Hard
Right	331	94	20	30	34
Wrong	390	84	21	38	15
Total	731	178	41	68	49

Data Values and Distribution

In Table 7.3, mean and distribution from the mean (standard deviation) are provided for each feature except for the player's performance. Performance is discussed separately in Chapter 10 based on questions, sessions, and the game (See Figures 10.5 and 10.6 for detailed plots). The physical data of motion and pressure are not universal measures such as inches/second or lbs./square inch. The units per measure and unique to the toy but consistently collected based on calibration of each force sensor for pressure and a single accelerometer collecting motion data.

Table 7.3 Range, Mean, Variance and Standard Deviation of Independent Variables
*A threshold of 0.3 and 40 was used to limit noise from the motion and pressure sensors respectively

	Range	Mean	Variance	SD
Motion	0.3* – 141.5	**6.5** = 303123/46626	48.36	6.6
Pressure	40* – 3615	**489** = 6237880/12748	179256.7	423.4
Duration	7 – 516 sec.	**23** sec. = 1290823/56141	13036.52	114.2
Difficulty	1 – 9	**2.3** = 1802/779	5.03	2.24

On average the toy did not move much with a 6.5 mean however, the highest force exerted on the buttons was 6 X that of the mean between players. The duration of each question is calculated based on the start of the question and stopping once completed answers were given over 15.5 hours of questions. Despite the low average difficulty, the standard deviation is also low meaning the average player is playing at a very similar level to their peers.

Data Cleaning and Preprocessing

A row of data produced for each interaction with the toy, left many cells within the row as nonvalues (NaN). This is because the data is collected on a one-second timer, logging any change in the state of the toy, including LED's turning on and off, motion of the toy, etc. Using a fill-forward technique, the database copies the previous cell values until a new value is encountered. The new value would be used until the next value was discovered. The sensor data determined the number of rows, and the timestamp of each interaction uniquely identified each row. The fill-forward method keeps each question chunk intact because the pressure values consistently registered 0 before NaN values were recorded, and motion values alternated with pressure data recordings. To observe the shape of the pressure readings all pressure and motion data was preserved rather than flattening it by calculating peak pressure or motion or mean values. Figure 7.2 shows the patterns of pressure data over time, with motion data overlaying the gaps between pressure readings. Both values are captured only during high moments of activity. As we can see in Table 7.3 some standardizations or normalization methods (feature scaling) are necessary to scale data proportionately. This is also necessary to moderate any influence of one feature over another achieved in the TS software through weighting as well as using feature scaling methods. Again, this is applied consistently across all variables in the TS version in contrast to the ML which changes during the training process.

Feature scaling for ML is the proportional reduction of the range of data in any one feature profile. It is a required method for neural networks or gradient-based models and a helpful method for distance-based models such as k-NN and k means. Standard feature scaling shifts values to center around the mean, with a unit of standard deviation, which means the ranges between features can differ. Using normalization methods each feature was set within a range of 0 and 1. After plotting, normalization of each feature gave the distributions a distinct right lean, whereas standardization provided a more expected outcome. Standardization is preferred to maintain the relationship with the original data and is also useful for Gaussian distributions. The data used in the tree-structured model was weighted; however, it was a fixed weight and consistently applied across all data features. For supervised models in the machine learning version of the toy (such as k-NN), the data was manually checked sequence by sequence categorized as the training target results. Target training is done using the labels assigned to increasing, decreasing, or retaining the difficulty level of the set of 5 questions. Normalization is used when the dependent variables are "known" for these models and a label is assigned. From the normalized data, the sum of each feature was calculated and evaluated by ranges between 0 and 5 based on the total number of features. The minimum value is 0 X 5 features, and the maximum is 1 X 5. *The "known" labels are determined to be correct when a player performance corresponds to the following set after the prediction is made.* The labelled groups were calculated as follows: if the total correct answers were ≤ 2, it was given an 'easy' label, if > 2 and < 4, a 'stay' label; and if ≥ 4, a 'hard' label was attached to the sequence. A corresponding performance with a proceeding set is a subtle determination between the score of the current set and prediction of the next. For example, if a player on level 2 achieves a 2/5 the prediction based on all the features is made to increase, decrease, or remain at the level for the next set. If the player, then scores 4/5 in the next set the prediction is determined to be correct. This assessment is not as straightforward as it may seem. The model could predict any of the choices followed by a 4/5 performance result. However, the probability that the player will score well after scoring poorly on the previous question(s) is less likely than if the difficult level is dropped. Training of the model is making the determination is based on two factors; the first is that it learns from more than just one question but rather the pattern of responses throughout the game, and secondly from more than player performance data. Discussion of the ML model can be found in Chapter

9 however to clarify the data process for the model, manual adjustments were made to 10.5% of the target labels based on performance across multiple sets and not calculated in the same way as during the TS game. If the player demonstrated 3 or more performance scores over 4/5 and the difficulty level was not raised the label was changed to 'hard'. No labels that were changed moved more than one level, for example no label moved from 'easy' to 'hard' and most moved from 'stay' in either direction.

Processing Time Data

Duration calculations capture the timestamp when the question was answered (end-time) minus the difficulty level assigned (start-time). This difficulty data is used as the start time because it was the first available indication a question was asked. The end of the question was taken from the toy's first performance (answered question) indication, identifying that the player's interactions were completed. Subtracting the start time from the end time was stored in the data as duration. The mean value for each question asked and answered was within 23 seconds across all 1001 questions.

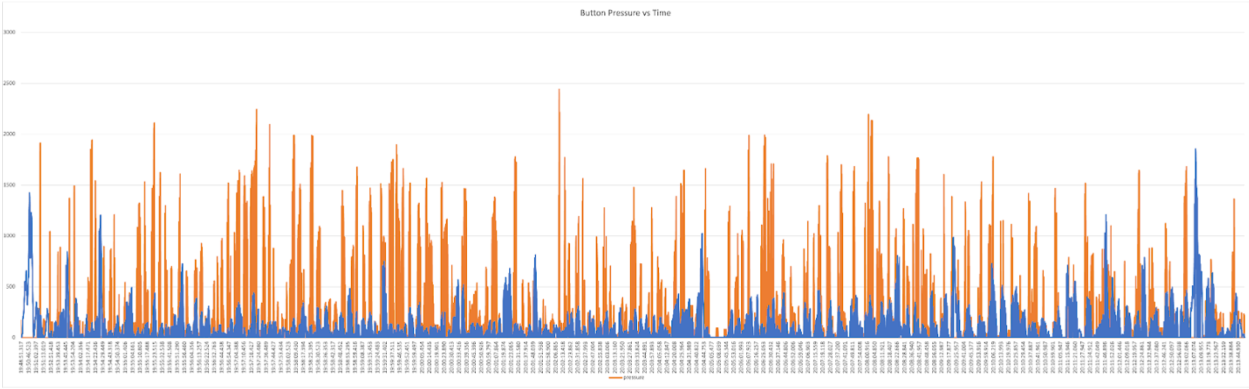


Figure 7.2 Pressure and Motion over Duration

Each light is set to turn on and off in 1-sec intervals, with a minimum of 2 LED lights for the easiest and 5 LED's for the hardest difficulty level 8. The response indicator of performance was also set to 1 sec, giving an 'asked' duration minimum of 3 seconds and a maximum of 6 seconds for the most difficult questions. Answering the questions varied a great deal between players. In some cases, players averaged 100 sec (1 min 40 sec) per question. According to Blanche (2006), you can conduct experimental research in situations where time is a vital factor in establishing a relationship between cause and effect and where invariable behavior between cause and effect not only exists but understanding the importance of the behavior is measured. We can see the relationship between the duration of the sequence of questions and child performance data in Figure 7.2. Across the performance of any sequence there is a slight increase when players score

3 out of 5 but generally there is an equal distribution over time. Notably the players who score 5/5 do take less time than the other scores with an average of 50 sec per set, compared to players who score 0/5 who average 64 sec. per set.

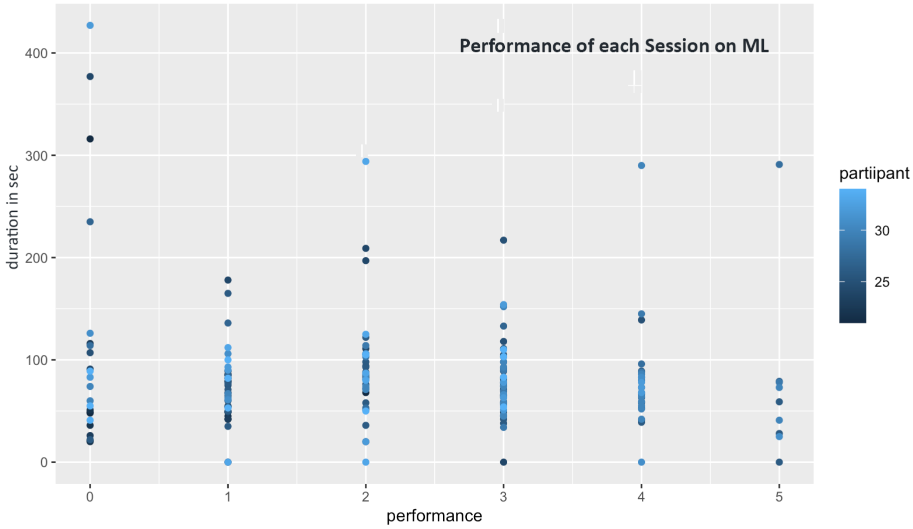


Figure 7.3 - Performance Set of Five Questions over Duration

This is not an indication of struggle with the game, nor is it an indication of engagement for all players. Some autistic children do fixate on activities they like, but longer duration times during questions, sequences, and total amounts playing can best be described as a willingness to play. The extremes at either end of playing duration range may suggest interest; however, there is no clear indication of where the distinctions lie between these ends. At what point do players disengage? This is a floating value and not distinguishable in time alone. Similarly, struggling with questions is notable over time but difficulty level also is influencing performance. For this reason, an accurate measurement and consistent treatment of the data provides a clearer picture of any set of questions to base the prediction.

The Shape and Processing of Data for ML Models

The ANN model combines numeric continuous values such as pressure, motion, and duration, as well as categorical data, such as performance (nominal) and difficulty (ordinal). Using classification models for machine learning requires one or both data types and necessitates cluster algorithms over other deep learning models. In the case of k-NN and k-means, the analysis and prediction of the models use low-layered neural networks where ANN leverages multiple layers to determine the weighted relationship between data features. The low neural net levels of cluster models favor pre-weighted values before assigning to a cluster; in contrast to the ANN model, which uses gradient descent or the rate of prediction of the dependent variable as x is nearing 0, the actual value of the independent variable (See Appendix L). Predictive models where x=0 are no longer making predictions but are overfit using actual values from the training set. Alternatively, too little data and the model is making assumptions about the data that are not reasonably supported. As we will see in Chapter 8, the k-NN and k-mean prediction performance is not as high as the Neural Network model. This is likely due to variations in features that are so unique to individuals that predictions are difficult to make given the two dimensions of k-NN – even using normalized data and weighted relationships between data points. ANN models have the advantage to using less data and still make a greater percentage of accurate predictions.

Data analysis to this point considers the comparisons between features and their statistical considerations. The machine learning considerations require decisions to be made about the data's form. If the output for the model were simply a binary choice, a logistic regression model could be implemented. Binary logistic regression models analyze the relationship between a set of independent variables and a binary dependent variable (this, not this). It is useful when the dependent variable is dichotomous, like death or survival, absence or presence. (Penman, 2022). Here, the multidimensional dependency of the difficulty prediction (harder, easier, or stay the same) can be done using multi-classifier models. Between the different models, a strong argument can be made for k-NN, in part because it handles non-parametric data, the importance of which is discussed in Chapter 10, and the simplistic clustering method. Clusters of data are grouped using the closest neighboring trained data to calculate and classify a new observation location within the group. The label assigned to a new observation is the predominant label given to the nearest training data to classify the unknown example. The main detractor to k-NN as a model for games is the slow response rate due to its measuring of all the training data rather than some sub-classification function that would make it more efficient. It is intentionally simple and often called a lazy learner but, in this context, a more efficiently trained model is needed.

Artificial Neural Networks (ANN) the Best Choice for Toy Data.

Selecting an appropriate machine learning model is often bound by the data types, scale, and quality of the data. In this dataset there are two categorical data inputs that can be encoded as integer or float values, and the remaining three data types can be expressed as multinomial logistic regressions. Given the scale and processing to improve the quality of the data an argument for ANN is fitting. The data is fed through the ANN assigning a weight and bias once it is determined to be 'activated' or important enough in calculating the prediction. A more detailed description of the model is discussed in Chapter 9 but developing an ANN for this data and this study to a larger degree, is the advantages of speedy responses and potential for improvement. Collecting data at 1 second intervals produce a reasonable amount of data for predictive models that are efficient in processing for game play. More importantly, improvements in future training and testing models are more easily adapted to the game as a function that can be ‘plugged-in’. In anticipation of future studies this favors models that have flexibility both for larger datasets and shortened processing time. Together these advantages inform the choice of model to use but the collection, processing and ultimate quality of data determines at least half of the model’s performance.

Chapter 8 - Study 2: ‘Tree Structured’ Programming

Research Design and Methodology for Analysis

Following the paper prototype study design considerations as well as software objectives and functionality were developed. The first version of functioning software was ‘Tree Structured’ (TS), based on the conditional statements used to generate questions which is a typical approach to programming many types of software. The purpose of this study was set a benchmark for the two forms of performance evaluated in this study: the participants’ performance and the software performance as it relates to question difficulty. Student performance is measured using correct answers, difficulty, speed of answering, toy orientation, and button pressure. Student performance will also be tested using the pre and post-test results. This is a comparative study between students in the first group, who use the tree-structured software, and students in the second group, who use the machine learning software to measure improvement in answering questions correctly. Play session performance and duration are more influential than the other collected data from participants. Although important, player performance is not the only measure, the other features are tested for influence on performance as well as correlations between each other and software performance. Software performance being the second performance consideration, shift from student’s performance as an independent variable to dependent variable, using the accuracy of the predetermined calculation for the next question, answering RQ5.

To start, the relationship between variables is evaluated using Pearson’s correlation test to answer RQ2. This will indicate the existence of a correlation, the strength of that relationship, and the direction of one toward the other. Lastly, the session data compared between participants indicates loose patterns that reoccur, such as a preference or performance of one game over the other (see RQ1). The purpose of a correlation test is “... to investigate the extent to which differences in one characteristic or variable are related to differences in one or more other characteristics or variables” (Leedy and Ormrod, 2010). In this study, the correlation method is used, where no manipulations of the variables to observe a change in other variables. Rather, variable changes will be measured based on their relationship to each other to determine the statistical significance effect. The directional change of these variables expresses the relationship, “A correlation occurs if one variable (X) increases, and another variable (Y) increases or decreases. A study with a correlation coefficient of 0.00 signifies no association between the variables investigated” (Curtis, 2016). However, the users who interacted with the toy provided the independent variables' values. The value inputs (independent variables) are collected from physical interaction, toy orientation, speed to completion, and game level to determine the influence of the right versus wrong response variable. These variables are collected by the toy and, at the same time, used to make predictions. More specifically, the

Pearson product-moment correlation coefficient, r, describes the strength of a linear relationship between these variables (see diagram X). Grove et al. identified three types of correlational research designs: descriptive or explanatory, predictive, and model testing. In this study, I focus on a predictive design that “attempts to predict a dependent variable's level from the independent variable's measured values.

Pearson’s Correlation Coefficient

The relative comparison method, Pearson’s correlation coefficient, accommodates different units of measure. For example, we can correlate a person's height with his weight. It is designed so that the unit of measurement can’t affect the study of covariation. Other distinctions of the test include the following:

1. Zero order coefficient (bivariate correlation X, Y, Z – XY, YZ, XZ)
2. First order coefficient (partial correlation controlling for one dependent variable)
3. Compare the values (if they are similar/identical, they are positive)

Pearson’s correlation coefficient (r) is a unitless correlation measure. It does not change in the effect of origin or scale shift measurement, which is a matter of direction between variables that cannot be determined using this test. Nor does the test consider whether a variable has been classified as a dependent or independent variable. It treats all variables equally. We want to determine whether a player’s performance correlates to their physical interaction with the toy. However, to determine whether a person’s pressure on buttons influenced their math performance (which makes little sense), the result will be the same as their math score influencing how hard they push which, may have some correlation. In this scenario, the predicted difficulty level is informed by the previous difficulty levels, but which difficulty level has the impact will need to be verified. The statistical analysis explained later in this chapter will cover the direction or the effect of an independent variable on the dependent variables.

Pearson’s Correlation Test

$$r = \frac{n\sum xy - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}$$

Where:

$cov_{x,y}$ = covariance between variable x and y

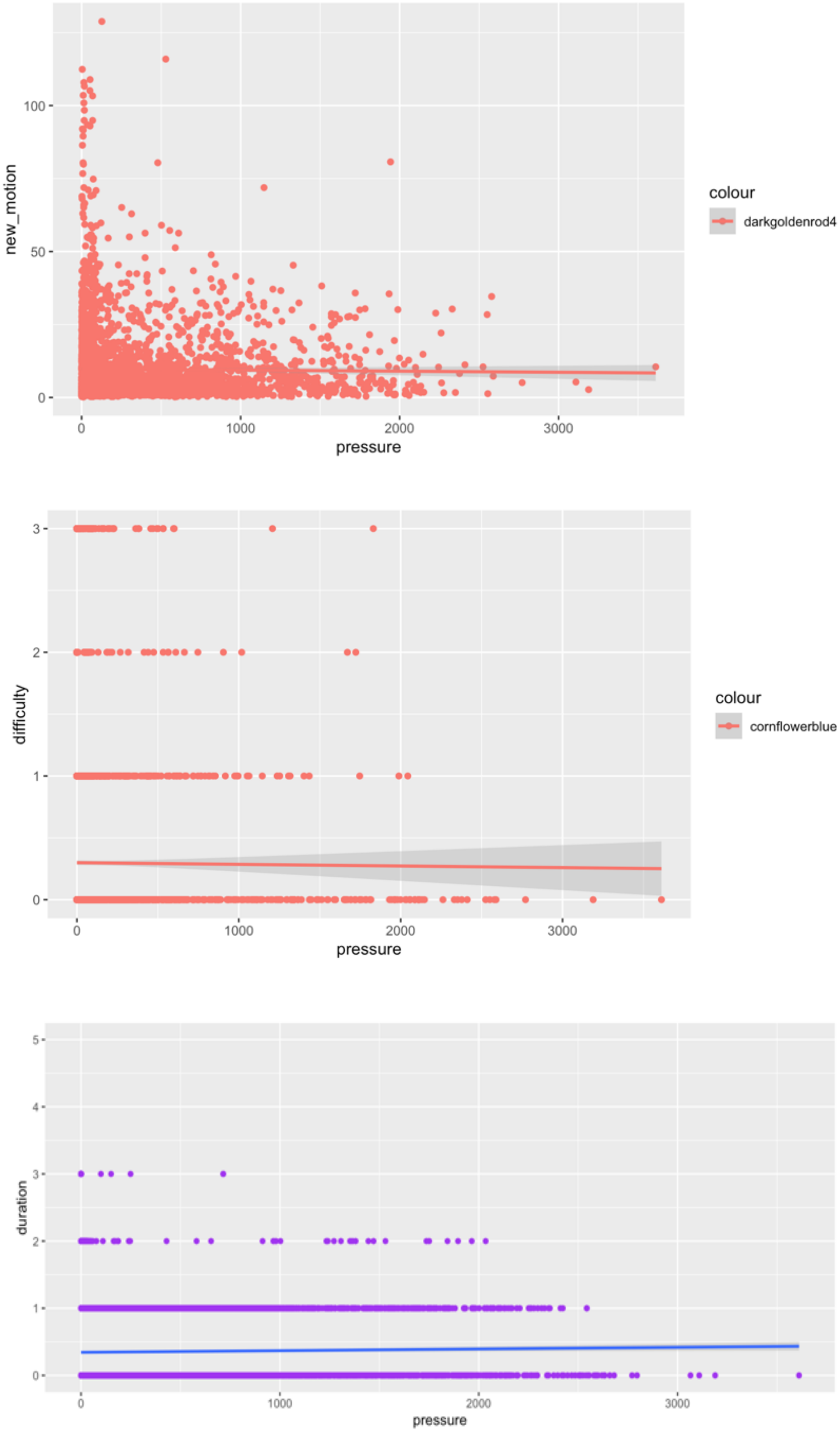
x_i = data value of x

y_i = data value of y

$\bar{x}$ = mean of x

$\bar{y}$ = mean of y

N = number of data values

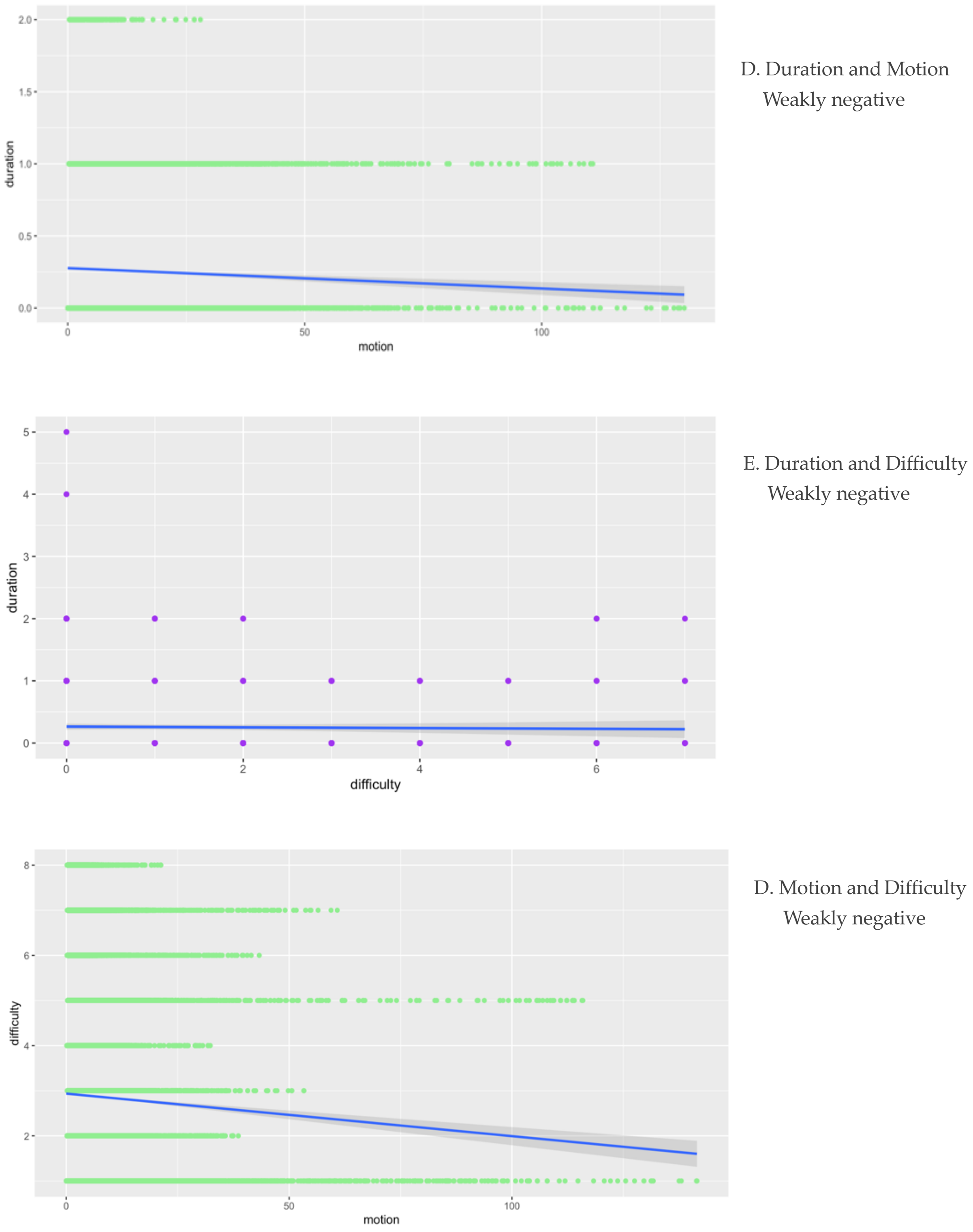


Pearson’s correlation test
A. Motion and Pressure
Weakly negative

B. Difficulty and Pressure
Weakly negative

C. Duration and Pressure
Weakly positive

Figure 8.1 Pearson’s correlation coefficient results show a weak negative correlation between all variables except between pressure and duration, which is weakly positive.



D. Duration and Motion
Weakly negative

E. Duration and Difficulty
Weakly negative

D. Motion and Difficulty
Weakly negative

Figure 8.1 Pearson's correlation coefficient (continued)

The results of the Pearson test are promising because of the strength of their relationship as well as their overlap. When selecting features, it was important to find those things that could inform the output of the toy to correspond to the performance of the player. The collective of features is described by Ghiselli (1964) as a composite:

> "When we develop a composite which we intend to use as a basis for predicting an outside variable, it is likely that the components we select to form the composite will have relatively low inter-correlations. When we seek to predict some variable from several other variables, we try to select predictor variables which measure different aspects of the outside variable."

Table 8.1 – Pearson's correlation coefficient scale for strength and direction.

Pearson correlation coefficient (r) value	**Strength**	**Direction**
Greater than .5	Strong	Positive
Between .3 and .5	Moderate	Positive
Between 0 and .3	Weak	Positive
0	None	None
Between 0 and –.3	Weak	Negative
Between –.3 and –.5	Moderate	Negative
Less than –.5	Strong	Negative

Ghiselli goes on to say, if the correlation between each of the components in a test and the output variable is known, and the inter-correlation between each pair of components is given, then the correlation between a composite test consisting of the summed components and the output variable can be predicted. (Ghiselli, 1964)

The data used to determine the correlation, such as pressure, motion, and the current difficulty readings, is directly taken from the TS database (Figure 8.1). Duration is determined by the toy response to the answer input (previously completed answer) to the frustration score response of the current question (calculated after the question is asked and responded to by the player). Of the players included in the machine learning study, 735 questions were completed – which is approximately 147 sessions (not all 5 questions were answered in the last session). Each player answered an average of ~62 questions and each answer took 3 button presses on average, totaling 2205 interactions with the toy. This does not include picking up the toy or counting the dots on the buttons, for example, but provides some general concept of the scope of data.

The Pearson's data shows the toy is consistent for both TS and ML, providing confidence in the comparison between the two types of software. It also indicates the limited amount of overlap between features, adding to the confidence in the model and limiting overfitting the data.

Table 8.2 – Pearson's correlation coefficient results from the four of five variables of machine learning.

	Pressure	Motion	Difficulty	Duration
Pressure	NA			
Motion	-0.0141097	NA		
Difficulty	-0.005979271	-0.01824944	NA	
Duration	0.01896174	-0.0578797	-0.008756353	NA

From the test, table 8.0 shows the correlation between the different variables and their strength. As we saw in the Pearson's correlation graphs for TS $p < 0.05$ which are similar for ML between the variables and in general there are only weak relationships between the variables, offering a high level of independence. Interpreting the results of the Pearson's tests we see that most correlations are also negative in their relationships. This indicates an inverse relationship, the more a player presses on a button, for example, the less the toy moves. The longer the player takes to answer, the less the toy moves, and the harder the questions, the less time it takes – however only slightly. Only pressure and duration had a positive correlation where both values increase, which was somewhat expected, since the more pressure increased the longer it took to answer the questions. What is more notable are the relationships between motion and difficulty, and between pressure and difficulty. One way to interpret this finding is players pressed harder and moved the toy more when the questions were easier. Conversely, the lighter the pressure and less motion the more difficult the questions. These correlated relationships may be an indication of engagement, for example the former scenario could suggest greater physical interaction with easier questions because they are excitedly answering correctly. Or the more still the toy and players pressure the more a player is cautiously considering the question. A closer look at player performance relative to these variables may offer insight into the direction of influence in the correlation results. Clearly this is one interpretation, but the relationship is only weakly correlated regardless of its meaning. More broadly, the influence of each variable on player performance is predetermined in the tree-structured software because it is set at the time of programming. The degree of influence on games difficulty is a subtle balance to keep players engaged and at the same time challenged.

Predetermining Player Performance

At the heart of comparing between the tree-structured program to the machine learning model is the issue of anticipation. Can programmers anticipate the state of a player and accommodate for every child, striking the balance between engagement and challenge? Conversely, can the machine learning model make accurate predictions in every scenario for every child as they play? To be clear, the answer to both these questions are tentative however, let us consider the first question specifically; to what degree can we predetermine an appropriate level of difficulty? The following code snippet (code 8.1) is taken from the current tree-structured version to explain its function, nuance, and implications of this approach, as well as discuss what is and is not included in the way it functions. In the function, we see the logic of tree-structured programming to anticipate student readiness for the next question. The function determines the frustration score, which is a slight misnomer in that it does not calculate the level of frustration. However, it considers the five data points being collected and the calculation made to determine the next question offered to the game player. It captures the input events and the time of each sequence to make the calculation.

Code 8.1 snippet calculating frustration score to determine the following question to pose to the player.

```
// inSession is true when amid a series of challenges
let inSession = false;

//This is where the answer is added
let answer = '';
// frustration level
let frustration = 0;
// number of seconds since the start of the session and challenge
let sessionTime = 0;
let challengeTime = 0;

 const weights = { pressure:1900, motion:0.0001, correct:-100, incorrect:100, time:0.01 };

  const updateFrustration = options => {
    if (inSession) {
        if ('pressure' in options) frustration += (options.pressure - 100) / weights.pressure;
        else if ('motion' in options) frustration += (options.motion - 100) * weights.motion;
        else if ('correct' in options) frustration += options.correct ? weights.correct * (difficulty + 1) :
  weights.incorrect;
       else if ('ctime' in options) frustration += options.ctime * weights.time;
      getElement('frustration').innerHTML = Math.round(frustration);
    }
  };
```

To begin, either game (math or color sequence) uses this same function to determine the next question; however, as previously discussed, the game levels are unequal. Level one for the color game is widely perceived as easier than level one for the math game. In the scenario where the next question should be more difficult if the game is also changed from color to math, the next question would increase in both question difficulty and the perceived game difficulty. A player of the color sequence game on level one who answered four of the five questions correctly would move to level two in the math game. In part, this is so the player experiences both games that we can observe the knowledge transference between games to assess its implications.

After five questions in each game, the *updateFrustration* function is called, and it passes the values of all the options collected by the toy. These include the five data points of button pressure, orientation speed, player performance answers, level of question difficulty, and speed with which the player answers the question. These are defined in the 'const weight' variable after the two weighted data points and before being added to the frustration score: the pressure sensor data and the performance data. The other variables are best described as normalized using a single multiplier to determine the range. Typical pressure data ranges between 100 – 2000 units of pressure, which is reduced by 100 to provide a value representing the vector of a press, which peaks at 1900 and is normalized between 0 and 1.

```
frustration += (options.pressure - 100) / weights.pressure;
```

The above calculation is for each button press in the sequence, replacing the last value when updated and resetting each response to each question. Samples of the current pressure are taken at one-second intervals, typically updating four or five times per button press. For example, a typical button press looks as follows: [0, 183, 376, 892, 1250]. Although the first value is 0, the toy does not recognize values under 100 as a button press due to noise in the pressure sensors. Using these values the normalized frustration score for the button press would be rounded to the nearest two decimal points: [.04, .14, .42, .6]. The argument for weighting the pressure data was to distinguish the different players' abilities and strengths, which adds more to the score if a child is pushing hard. This can be interpreted in several ways but only slightly favors a player's negative response to the game. Clearly, this is a suggested interpretation; however, it does indicate the difficulty in anticipating the player's response to the questions. Some assumptions are made, knowing full well that they do not represent everyone pushing hard or softly on the buttons. This is mitigated somewhat by the other data points and the pressure values leading up to the peak score, as well as other button presses and follow-up questions. In other words, no single button press value during a given second will determine the next question.

Similarly, player performance results are weighted because of the unique predictive value performance suggests. Although children may press the buttons vigorously and move the toy in space aggressively, they may also be answering the questions correctly and correctly and should be challenged more. The overall frustration score is weighted by the number of correct answers out of the five in the game sequence. The weighting of performance has a significant influence on the difficulty of the next question. Although the range of frustration scores is more easily distinguishable for the individual player, it is not the only factor in selecting the next question.

All other data points are added to the frustration score – with two exceptions. The motion value from the 'accel' data is often negative due to the direction in which the toy is moving based on x, y, and z coordinates, lowering the overall score. The absolute value of motion is provided to *updateFrustration* function in options. Time is also a reverse order value in that the faster the player answers the question, the less is subtracted from the frustration score, and the higher the number, the more likely the difficulty level will increase. We can imagine this approach to the frustration score as a running total for the sequence in the game. Other more sophisticated techniques, such as normalizing the data first and equalizing each data point before weighting, are possible; however, the ultimate determination is between three possible outcomes, emphasizing the thresholds between the outcomes rather than the outcomes themselves. For the programmer, the best efforts are placed on determining the calculations of the thresholds for each player to change based on their input, using median values as a benchmark at the beginning and adjusting them as the player plays.

Still, other approaches have been considered, such as making the game level more granular a choice rather than having just three possible states for the next questions. Because there are two games and three potential outcomes (easier, stay the same, or harder), there are six possible choices among difficulty levels. If, for example, a player is on level two in the color sequence game, one possible increase in difficulty is to add to the number of colors to remember. To lessen the difficulty, reduce the number of colors or keep them the same. In addition, the game mode could switch to math at the same level, which is a slight increase in difficulty given the complexity of math compared to simply following the color sequence, or lower / raise the question difficulty in addition to switching the game mode. One reason to avoid this approach was to make the game more predictable and draw a more significant distinction between levels. After five questions, the game mode changes when a harder or easier question is considered. If a player is in the color sequence mode, the difficulty will increase both in-game to math and in the type of question given a high frustration score. The transition then between math to color and color to math is different, allowing for a closer examination of the player's ability to transition in the context of difficulty. Suppose the game mode changes were not predictable before playing. In that case, it feels too chaotic as each question could change games, or the player is stuck in one game mode, making play too repetitive and lessening engagement. Still, other approaches are not cited here, and refinements of the frustration score could be nuanced further, providing more options for difficulty level thresholds; however, these same categories defined here are also constrained in the machine learning version to allow for comparison. The critical difference will be the strength of the probability of the frustration score to predict the following questions in real time, freely moving the thresholds depending on the player.

Order of difficulty

Level.	Game.	Button presses in the correct order.
1 –	color -	2
2 –	math -	3 addition (4 + 3 = 7)
3 –	color -	3
4 –	math -	3 subtraction (4 – 3 = 1)
5 –	color -	4
6 –	math -	4 addition and subtraction (4 – 3 + 1 = 2)
7 –	color -	5
8 –	math -	3 multiply (2 * 3 = 6)

The questions were generated randomly; only the numbers or colors needed to be pressed since the operators were displayed in the center light to indicate the question type and game level.

Questions About Programming

Adding weights to the variables in the tree-structured version is in anticipation of the nuance between variables like performance versus pressure, such that performance is more important than pressure in predetermining the next question, giving it more weight. Within the software itself, it looks more like normalizing the data because only one value is applied, for example, to all the performance data. Performance, therefore, has more influence on the outcome, but this, too, is nuanced because the other data points could make up the difference between a correct answer versus an incorrect answer if the sum of all the different variable weight influences were larger than that of performance. The outcome, calculated in the *updateFrustration* function, is relatively predictable, even in the tree structure. This leads to the importance of the tree structure data in training the machine learning model. A plausible question comparing the two software is, 'Wouldn't the AI model just be as good as the tree-structured (TS) version, given it is trained on tree-structured data?' In a supervised model, labeling the data indicates whether the program responds as intended; however, a more significant distinction is made between each variable and the comparison between variables. This ability to make the distinctions implies that AI could outperform the tree-structured version. If a player is given the sequence blue, green, and yellow, according to the rules of play in the TS program, blue, yellow, and green are incorrect. However, it is less wrong than blue, green, red, or even less than purple, red, or orange. While we can imagine accounting for all of these possible answers and giving them more or less weight in the tree-structured program, subtle variations using order or proximity between buttons make the weighting process more problematic. Using this same example, is blue, green, and orange more or less wrong than blue, green, and white? We could say that white is the absence of yellow, whereas orange includes red, and therefore white is less wrong. The locations of yellow and red on the toy are immediately next to each other, where white is on the opposite side of the toy, making red more correct. This example shows how machine learning can make more subtle changes between weights, affecting each feature to determine the next question. Answering our posed question, machine learning models have the potential to outperform the TS program because of the nuance of weighting and their ability to learn the variance of correctness, which is more difficult in traditional programming.

Appendix K provides a general look at the motion plots of each participant during their session suggesting characteristics of some common patterns in how they play. Motion is calculated from the accelerometer built into the circuit board, which measures the motion from any given state every second. Many of the sessions show the toy is moved around more vigorously toward the end of the session, and many, but not all, also move at the start. Interestingly, eight unique patterns occur, and three repeat more than once, even in this small sample (the top two patterns on the left and the top pattern on the right). Looking at the motion variable, it can only be said that some period of a higher movement happens at the start of each session, followed by a period of 'settling' ending with an increase in motion.

This cannot be explained by players picking up the toy at the start of the session and then putting it down at the end because the durations go beyond a single question or the moment they decide to stop playing. In addition to motion, the pressure on the buttons over the same duration produced the graphs in Appendix J. The graphs indicate groups of participants that can be split out into groups based on touch: 'light, 'medium,' 'hard,' and 'super.' The pressed values range from 1700 – 2100, 2100 – 2700, 2500 – 3300, and 2700 – 3600, respectively, for the peak values and the number of presses over time. This calculation explains the range levels that overlap within the groups; however, there are clear distinctions between them. Comparing the two variables, no distinct correlations exist between light touch players and the patterns in moving the toy. In addition to sample size, the correlation metrics across all five variables show a similar but weak relationship between them. Therefore, little can be drawn from the data other than that there seem to be data clusters within the larger group. More studies are needed to determine how many more patterns emerge and whether dominant and recessive patterns exist. However, it is of value in supporting cluster analysis as a method, and if the patterns are distinct enough, they can also benefit machine learning classification models.

Each student was selected at random by the autism center administrator based on the student's schedule in class and meeting the requirements of the study. The first 16 students were placed into the tree-structured group (TSG). Any selection bias was minimized because student selection was based on availability and who met the requirements forming this first group. The other 16 participants were placed into the machine learning group (MLG). A post-test was planned for the study; however, controlling for maturation between the pre-and post-test varied greatly, and some students were enrolled in a math class, which would have skewed the post-test results. Students who met the requirements were somewhat limited, and forming a control group without either experience with the toy was not feasible. There is further discussion on the pretest in Chapter 9, but all students met the requirements for the study in terms of math skills and their ability to identify color. The average play session with the toy was 13.5 sequences, which is approximately 67 questions. Players may have stopped in a sequence, and others restarted the game in the middle of their session. The fewest sequences were 4, and the longest was 37, answering 20 and 185 questions, respectively. Participant 18 (figure 8.6) played the longest; however, participant 5 (figure 8.4) was one of three players to win the game by answering 3 or more questions correctly at level 8.

Patterns in Performance

In both studies, there are two perspectives on performance; the first is the toy's performance and its ability to make predictions or, in the case of the TSG, pre-predictions made during game software development. The second perspective is the play's performance, measured in correct versus incorrect answers. If we consider the toy's performance as a reaction to the player’s input, we expect to see the patterns shown below in the samples of play sessions. Players 5, 9, and 16 are typical students who played between 10 and 37 sequences of both color and math – listed here as ‘sets.’ The pattern between easy (green), stay (yellow), and hard (red) questions is expected since when a hard question is asked, the performance of the player goes down, and fewer correct answers are given. When easier questions are asked, performance increases – albeit more slowly. Players often performed well once the first increase in difficulty happened, dropping only one question on average. The toy would register this as improved performance considering the difficulty level increase, and if the duration of answering questions did not lag, it would continue to increase the difficulty. Other than player 9, we see a second increase in difficulty level impact on player performance, averaging 2 incorrect responses per player. If a player answers 3 questions correctly, the frustration score does not increase nor decrease, which is not to say the difficulty level does not change. The level changes based on cumulative scores

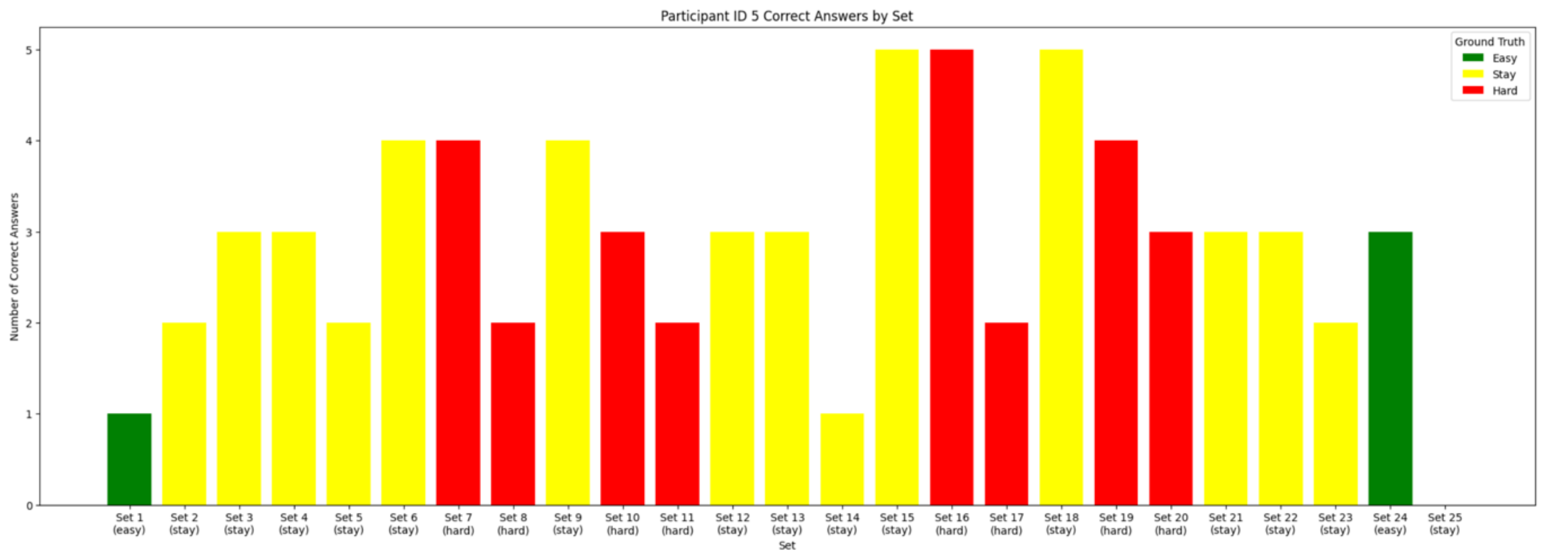
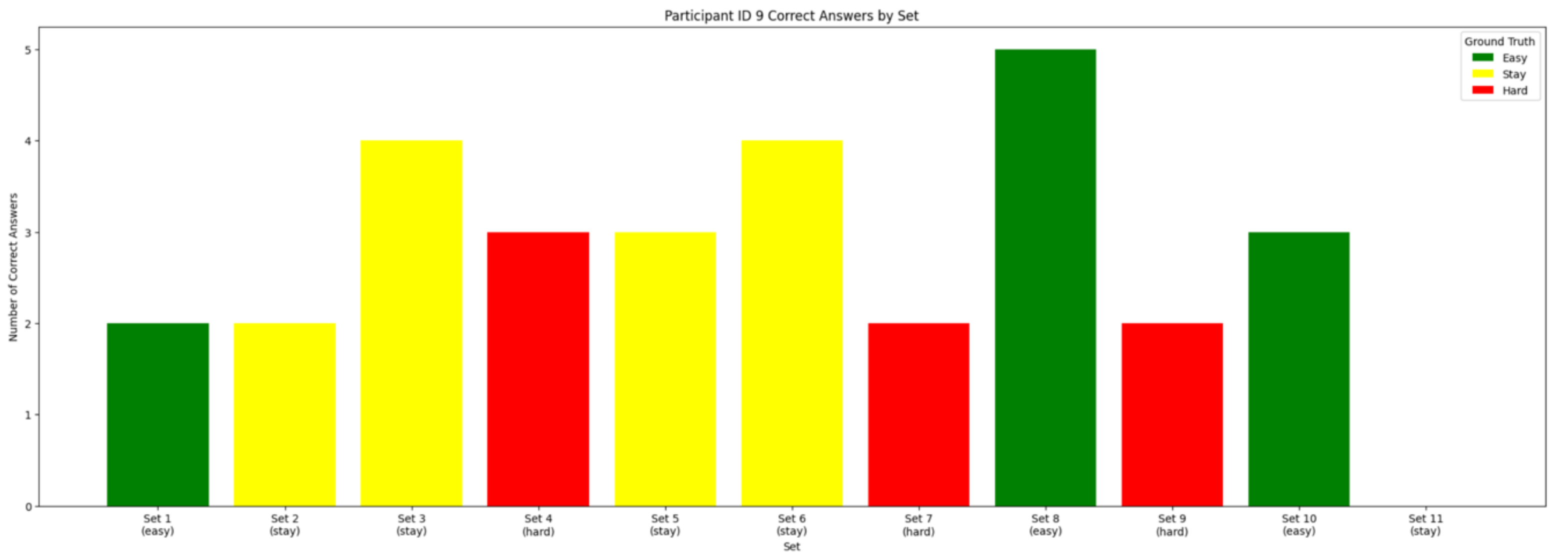


Figure 8.2 - Participant 5 averaging 3/5 correct answers over 24 sequences (sets)



Figures 8.3 Predictions for each sequence (easy, stay, hard) and the performance of participants 9

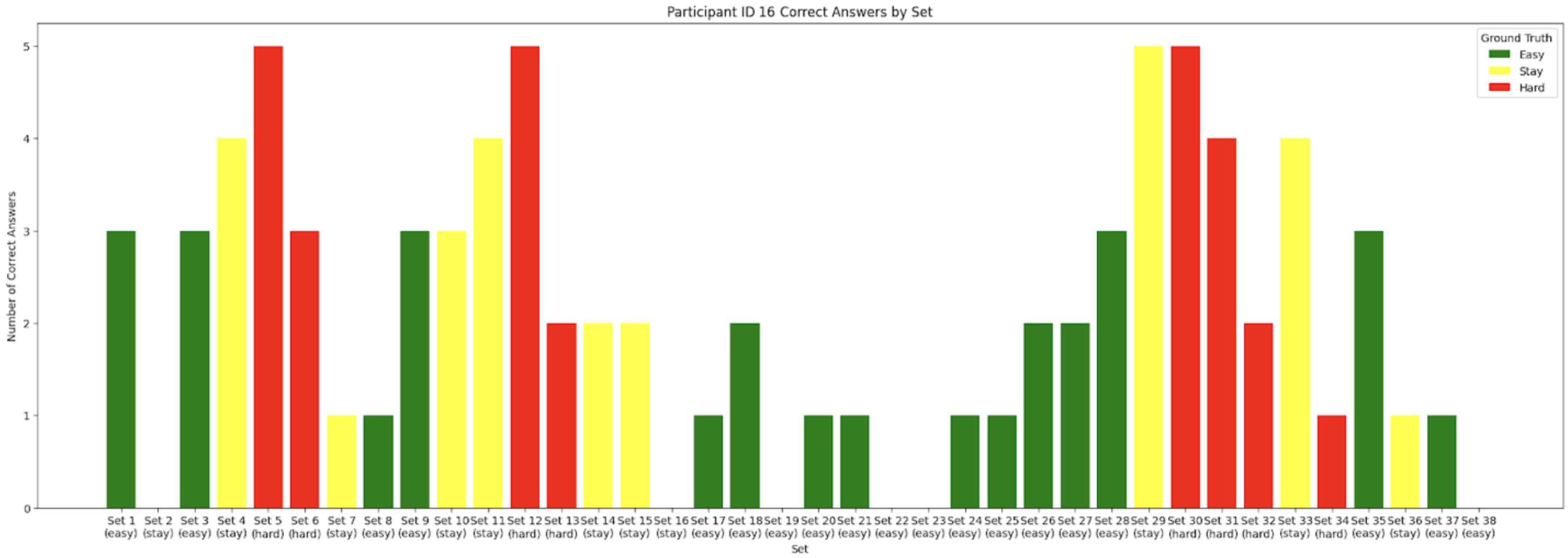
explain the long easing period for player 16. Earlier player performance was high, scoring 5 out of 5 on a high difficulty level; however, poor performance at low difficulty levels (0 out of 5 during sets 16, 19, 22, and 23) accounts for the continued easing of the difficulty levels. Comparing the performance of all three players, we see a contrast in the range of set scores. For example, player five only experienced an easing of difficulty at the beginning, the default level 1, and at the end of their time spent playing with the toy. The average score per set was ⅗, keeping most of the game at the same or increasing in difficulty level.

Other indications of the toy performance can be observed in player 9's performance during sets 7 – 10. The toy operated between the middle thresholds of 200 – 400 as a frustration score dramatically changed the difficulty levels after the player answered between 2 and 5 correct responses. The toy responded as expected, making the questions easier, and as a result, the player answered 5 out of 5 questions correctly. Conversely, the harder the questions, the poorer the player's performance. This scenario is central to the study because a pre-prediction can only respond to a player's past performance. Truly predictive performance of the toy should use past performance and the other inputs to anticipate drops after two increases in difficulty. The other two players fluctuated more between increases and decreases in level, which is typical of all players in the study (see Appendix 10 for all participant results).

Table 8.3 – Each participant's session, correct answers over total questions within the total number of sequences

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
Correct Answers	49	23	20	26	72	31	15	9	30	21	7	18	34	8	44	81
Total Questions	100	31	55	36	120	85	35	20	50	55	15	30	60	20	70	185
Total Sequences	25	8	11	8	24	17	7	4	10	11	3	6	12	4	14	37

An encouraging aspect of the data shows a high level of engagement, with an average of 12.5 sequences played by each player answering 967 questions for an average of 100 seconds per play session over a five-day period.



Figures 8.4 Predictions for each sequence (easy, stay, hard) and the performance of participants 16

Questions about Pressure and Motion

Each button on the toy was equipped with a force sensor to measure pressure on the buttons with the intent to answer Q2 and the influence of these features, including pressure, on predictions. We could imagine that physically larger or stronger participants could exert more pressure on the buttons irrespective of their intent or emotional state. This would be the natural state of each player and their tendency to push buttons based on how they perceive the button and what the button affords in terms of design. Figure 8.7 plots all the participants' pressure values throughout the first question, a two-color sequence. The duration is a complete cycle, starting at the first light of the question, through the answer being pressed, to the resulting confirmation of a correct or incorrect answer. The plot shows that some participants waited before answering but pressed the buttons within 100 seconds. Some noise in the buttons and low-level pressure from button caps are represented on the plot, leaving a trail of pressure readings. Therefore, a threshold of 40 and above was implemented to be counted as a legitimate press and included in the frustration score to make predictions. The threshold accounts for participant 6 and the delayed response, which is over 200 seconds. After viewing the video recording, the player missed the first question and timed out.

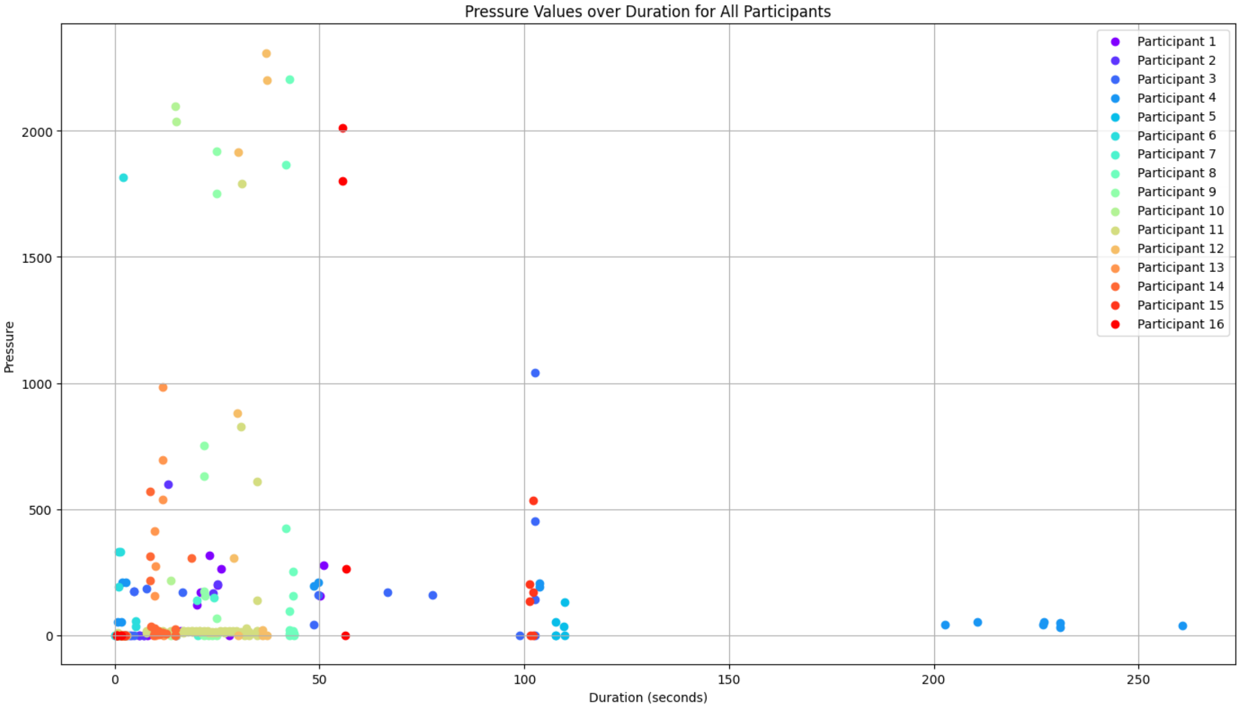


Figure 8.5 - Pressure on buttons to answer the first question in the first session by all participants in the TSG

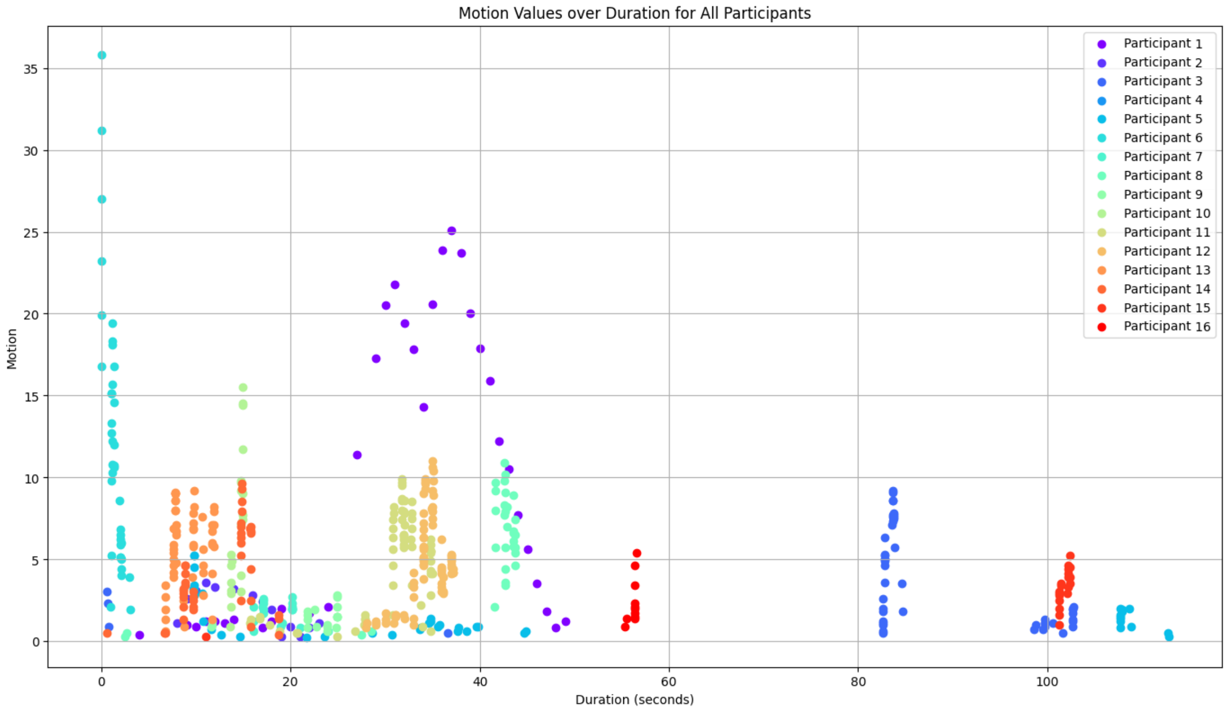


Figure 8.6 - Motion of the toy during the first question in the first session by all participants in the TSG

Overall, a similar pattern of rising sharply and falling after the button is released is seen in Figure 8.6; however, nearly half (7/16) pressed the buttons for the first question well over the maximum value of the other participants, ranging from ~1650 to 2500 between 1 and 3 seconds in duration. These same participants continued to press buttons above 1200 during other questions, forming a group of super pressers. They do not correlate to performance, nor is there a relationship to motion, as we see in the corresponding colors in the two figures above.

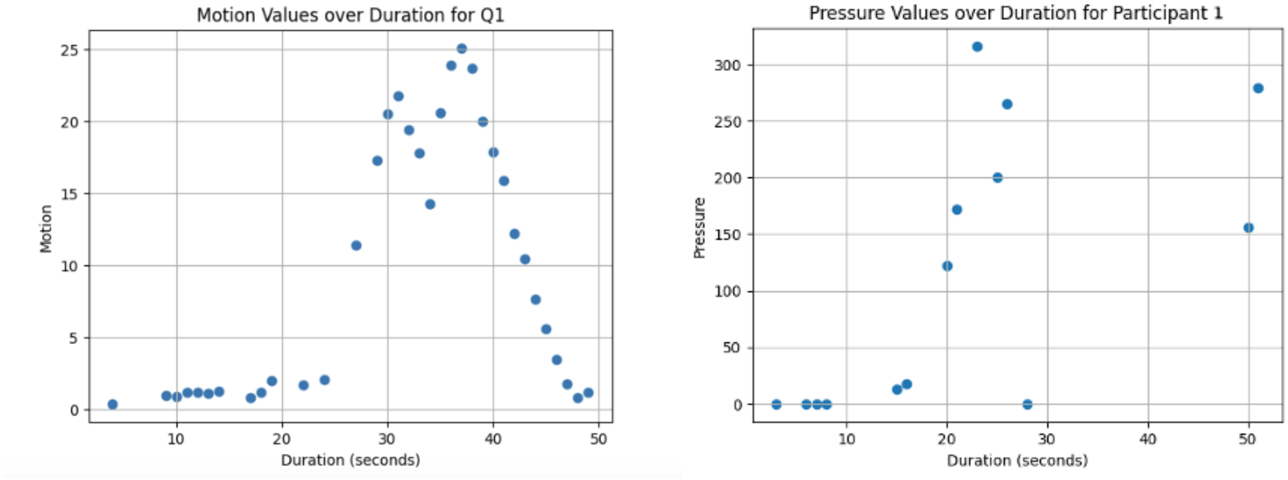


Figure 8.7 - Pressure and Motion readings from Participant 1 over time.

Range Values and the Line of Determination

Most modern approaches to programming are iterative and often follow an agile strategy, meaning refinements to code are done continuously. In industry, initial software is usually written and tested with a smaller sample of users and sent out as a minimum viable product. This approach adopts a mindset of organic growth that evolves, marked by significant updates and structural changes as its infrastructure of supporting hardware or software changes. The initial software deployment sets a trajectory of software that has implications for future versions. In this study, the software written for the toy is its origin, recognizing that it is just setting the trajectory for future versions. Lines have been drawn based on the current infrastructure and resources available. That said, the comparison between the tree-structured version of the software and machine learning is between origins, recognizing the process of both to be iterative towards improvement.

An essential aspect of the tree-structured software assessment is the definition of the structure and the relational analysis of the data collected. Within each decision of the tree, ranges of values were defined by the likelihood of a distinction between the input values. Capturing correct and wrong answers is distinct (categorical), even if no answer is part of the decision tree. The game difficulty is also categorical but ordinal, using the difficulty level to distinguish a ranking between them. The problem with the other three variables is at the core of the issue with the decision tree structure of the software. Continuous data is impractical to program because that degree of granularity cannot be predicted, and even if new unique values could be used to form a continuous scale, it is unclear how this would affect the next question difficulty output of the toy. Statistically, using categorical or ordinal data is the only feasible way to analyze it, but where the lines are between the groups and how they are drawn needs to be defined. Speed is categorized here as ordinal because there is a hierarchy of values – the faster the user answers, more is understood the assumption is an immediate response. They take more time if they are less sure of the answer. The same is true for different reasons if they answer incorrectly with a fast response, which may be carelessness but is still more highly ranked than a prolonged response that is still incorrect.

Table 8.4 – Data types for each feature

Variable	Right v Wrong	Duration	Motion	Pressure	Game Difficulty	Session Level (training target)
Data Type	Categorical Variable	Continuous Variable	Continuous Variable	Continuous Variable	Ordinal Variable 5 levels	Ordinal Variable 3 levels
Decision Tree Type	Categorical Variable	Ordinal Variable 5 levels	Categorical Variable	Categorical Variable	Ordinal Variable 5 levels	Ordinal Variable 3 levels

The function for pre-predictions in the tree-structured software was dubbed the 'frustration score' function, taking values of pressure and motion every second while collecting duration and performance data, adding in difficulty set in the previous frustration score. The calculation for the adjusted levels increased or decreased from the current score, such as with plus/minus grading. It operates as a sliding scale – adding or subtracting – which can build or deplete with each session of five questions. This approach ensures that the level more accurately reflects the player's interactions but also assumes a slow change in response to the toy and the player's own performance. The contention with a sliding scale is that each set of questions is not evaluated in isolation. A player could struggle with a sequence, but the toy would stay or go harder because the previous sequences were highly successful. It is important to note that within five questions, performance and engagement are difficult to measure from one minute to another. This approach allowed for a gentle fluctuation of change, giving players the opportunity to retry a level or be challenged further despite a poor performance in a single sequence. At the start of any game, the frustration score starts at 0 and is incremented by 100 points based on the five input features. Any player could reach the first threshold to jump a level after 2 sessions. In the opposite scenario, a player could be 'stuck' in several low-level sequences if their frustration score was meager because of multiple sequences that were unsuccessful. The mean duration of sequences for the tree-structured version was exactly 100 seconds for all players, which is fast enough not to hinder play or diminish the game's challenge to the point that a player might lose interest.

One Size Fits All and the Failure of Pre-determination

Behavioral software engineering researchers Per Lenberg et al. (2015) articulate a type of software developer who considers user perspectives when programming. Today, we think this is standard practice among developers; however, there are more factors than user needs when programming commercial software. Economically, it makes more sense, for example, to develop as broad an audience as possible to reach as many customers as possible. Historically, the approaches of some software developers have referenced the common assumption acc, according to Burnett and Meyers (2014), "if we build it, they will come." If a new tool or feature becomes available that can improve users' correctness or efficiency, then surely conscientious users will use it (Burnett and Meyers, 2014). This sentiment takes the position of pre-determination: if it is not only possible but more efficient and a less error-prone feature, why wouldn't people use it? This approach also appeals to a broader audience of users, but it assumes a range of skills and familiarity interacting with digital machines. Still, other methods for open-source platform software allow developers to customize tools built on top of base functionality, providing a measure of customization suitable at an organizational level. The benefit of this approach is often motivated by competitive advantage for the group of users but does not accommodate each user meeting their expectations (Gorton, 2011). Developing software genuinely customized to each user is the promise and advantage of machine learning models. Decisions anticipating the user's behavior have always been a balance of best practice, efficiency, and functionality, which varies between developer teams. Although the decisions for the tree-structured software attempt to incorporate the insight from the paper prototype, ease of use, and awareness of this audience, it is still one approach that attempts to serve many people. The notion of spectrum suggests autism is a range of skills requiring a level of customization for each individual.

Chapter 9 - Study 3: Machine Learning

To begin building a machine learning model, some considerations concerning types, fit and expected output need to be made in order to address the research questions. Much of the success of machine learning begins with the data and how it is preprocessed for the model to ingest. As discussed in chapter 7 the data to train this model comes from the TS software study which has been modified to align with the training targets. This chapter includes how the data is processing in training, testing and making predictions for the ML version of the toy. The types of models available for making predictions about the level of difficulty for the player of the toy are also considered for their benefits and drawbacks in informing the interpretations of their results, as well as explanations about their behavior. In addition to the model descriptions a comparative analysis of player performance patterns in the data between TS and ML software studies will also be covered. Lastly, a review of the results will examine the methods and toy performance output that partially answer the research questions 3 and 4 described along with their methods below.

Research Questions

In the context of educating children with autism, these machine learning techniques have been used to analyze a variety of data sources, including observations of behavior, response data, and other indicators of developmental progress. These algorithms can then be used to predict outcomes, identify patterns in the data, and design personalized interventions. An analysis of fit and configuration that is appropriate for both game play as well as prediction accuracy were the main considerations for implementation.

From RQ: 1 Can we see patterns in learning?
Method – Perform a cluster analysis from both training and test sets. Across all 2000+ questions we may never reach high prediction rates because the questions are not only determined by previous right / wrong answers but rather the factors of inputs to the toy from the training set. If there are large numbers of pattern clusters, predictions become more difficult. The kinds of cluster models tested for this study are, ANN, KNN, and K-means.

From RQ: 2 Are these the data features to use?
Method – Collect participant inputs from randomized machine choices and run regression models to determine significance of the variables selected. Using corollary determination analysis, a relationship value will be generated to indicate the impact of one value on another. This will be done once the training set data is collected. (see section How the Machine Learning Model is Trained and Tested pg. 134)

From RQ: 3 How does TS's pre-prediction approach compare with ML in prediction of question difficulty?
Method – Kruskal-Wallis statistical model to determine the significance of the independent variable effect on the toy performance.

From RQ: 4 How well can ML predict game level difficulty? (i.e. Toy Performance)
Method – Observation analysis using MANOVA to determine significance of player input relative to right / wrong variable. This is validated using a confidence matrix.

From RQ: 5 Can ML improve performance scores over tree structured programming?
Method – Comparative analysis between toy performance in both software environments.

The Toy is a System

There are a number of technical set-ups and dependencies that need to be created when considering an artificial intelligent (AI) system that uses sensor input from a data collection device. Like any technical system, it is a constant change of firmware updates, new module specifications, language compatibility, frameworks, and other evolutions in the development of software. As of this writing Tensorflow, MQTT, Python and MySQL are some of the links in a software chain to capture data, store and retrieve it to be used in in the game controlled over the internet using a machine learning model (ML) to predict outputs (Figure 9.1).

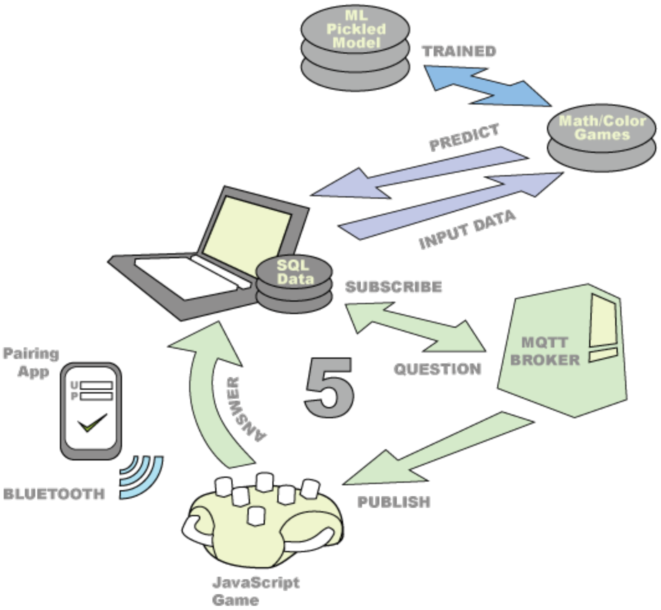


Figure 9.1 Network connections between the software (on computer), MQTT broker, the toy, and mobile app to connect toy to the network.

The use of ML configurations versus other AI tools will be explained more fully, but for the internet of things (IoT), implementing ML and more specifically cluster models, TensorFlow was determined to be the most ideal tool to choose, because a standard language could be used across the different platforms and services needed. Dependency on the internet to pair the device with a network, run the MQTT broker, the game, and store data, requires compatibility making the system rigid and susceptible to breaking. To unplug one part of the system to introduce another part causes the system to fail from incompatibility and therefore is not always practical to fully test the ML model in the real scenarios of playing the game. An additional problem with compatibility is the TS software was originally written in JavaScript and the MQTT broker and TensorFlow for the ML model are written in Python. This required rewriting software but retain its functionality while making it more efficient for real-time game interactions.

Rebuilding the Game in Python to Implement ML Model

The original game was developed in JavaScript, which was best suited for the paper prototype that used an Arduino board as a micro-controller. The new board for the tree-structured toy is agnostic in structure, relying on an MQTT protocol for publishing and subscribing to and from devices. The firmware for the board needed an upgrade to accommodate python, but the game functionality and structure stayed the same. The main modification within the program was the frustration score function which was replaced by the 'pickled' machine learning model, a compressed ingestion model made to predict using the five features without retraining (Appendix 3 Math Color Game in Python). A main concern when switching languages in software is the speed at which the software is processed, keeping the response and delivery times of the questions and answer responses similar so that game play is not affected. Both games average 250 milliseconds from the time the player presses the last button for the answer to the performance response from the center light or buttons indicating the correct answer. The machine learning model was even slightly faster in sending responses but had a wider range between slowest and fastest response times. Rather than manually weighting each feature, the trained model only needed to indicate the next difficulty level. The model was trained on the response data from the tree structured version with the data that best reflected the model accuracy. Each game started players on the easiest level for the first sequence. A difficulty level needed to be included to set a benchmark for all subsequent predictions. The first sequence for every restart of the game was removed because this would artificially inflate the 'easy' values. How difficulty levels are processed for the model determine the accuracy of its predictions which is to say, how the model uses data to derive at a label – in this case 'easy'. Two categories of model were considered: supervised and unsupervised or labeled and unlabeled data. Although the TS data was labeled, groupings of like data such as clustering or classification, also related to labeled and unlabeled data, needed to be determined.

Clustering Models Versus Classification Models

Two approaches to ML models are clustering and classification. Clustering models group data together based on a single or shared characteristics, and classification uses labeled data that is categorical for the model to assign the predictions. Clustering models like K-mean assign new observations to categories without labels, but rather use Euclidean geometry to adjust the centroids of the categories as it is trained (Milligan & Hirtle, 2012). K-means is often described as a classification model due to its assumptions associating data. The second clustering model worth consideration is k nearest neighbor (k-NN), which does not use categories – making it popular for labeled data. A simplistic description is k-NN uses plurality voting to determine the grouping of a new observation or how often a label is used to cluster the nth observation. The implication is that there is a ranking order of data rather than classes where the data belongs.

Although some literature use cluster and classification interchangeably here classification models use logistic regressions that apply weighted values to the data that detail the characteristics of the data both in relationship, influence, and directional value or factors. Logistic regressions assume a single dichotomous dependent variable. In this study, however, the predictions are categorical, and the independent variables are also multivariate. The type of models best suited for this type of data (multivariate multiple regression model or MMR) use neural networks to formulate a simultaneous response of the factors present in variables to the changes in others. Selecting between these types of models begins with an evaluation of the data and its shape.

Feature Selection for Prediction

Like the interchangeability of cluster and classification, two terms are used in machine learning circles to describe predictions. The term '*target for training*' references the categorical group which is labeled 'easy', 'hard' and 'stay', to determine the frustration score discussed in Chapter 8. Changes to the predictions are made manually to assess accuracy and are used in the test set as well as in the studies with participants. In these cases, the term 'Ground Truth' is described for ML by Kang (2023) this way, "…this approach of qualitatively tracing ground-truthing processes as ground truth tracings (GTT)… [that] foregrounds the broader socio-material practices of data collection, preparation, and maintenance" as well as the more common reference to logical processes of analyzing features to make predictions; a reference used henceforward.

“Features are relevant if their values vary systematically with category membership.” (Genari et al., 1989)

After the feature extraction phase, a total of 20 features were collected by the toy. To reduce the amount of overlapping features, possibly causing over-fitting, and in an effort to improve predictive accuracy, the feature set was selected based on the data type and the simplest value representation such as whole numbers. A correlation-based feature subset selection was used (Hall, 1999), where individual predictive ability of each feature along with the degree of redundancy between them was evaluated using the Recursive Feature Elimination method. Results of the feature selection phase confirmed the five variables which ranked within the top 10 out of 20. The x, y, z values of the accelerometer out ranked the motion variable, however, to reduce processing time motion was preferred and still rated within the top half of all features. After measuring information gain for each of the features and predicted value, the lowest contributing features were removed. The five remaining features were retained to train the 3 selected ML models. An examination of these three different models, K-mean, k-NN and ANN was done to determine the best fit for this study. The first two utilize clustering strategies, the difference is k-NN is unsupervised, and k-mean is supervised. ANN uses a neural network to evaluate inputs based on training data labels necessitating supervised learning. Of these three, ANN had the best accuracy rating and selected for its potential in future research; however, there are valid reasons for considering each model.

Machine Learning Models in Detail

K Nearest Neighbor (k-NN) Model

k-NN is a supervised algorithm that clusters data into a single group that allows multiple features to be used to locate the point on a multidimensional grid. The benefit of k-NN for this study is that the classification of the prediction can also be multi-dimensional rather than a binary output. The drawback is the need for the training set to be manually coded to assure a Target model for each set of five questions. Seen as a feature, there is no training to do with k-NN. Often called the lazy machine learning model, each observation is calculated once seen, which would take significant time in comparing each test data point. Hence, this technique is not efficient on big data; also, performance does deteriorate when the number of variables is high, due to the curse of dimensionality (Jiang et. al., 2012). The curse of dimensionality is a sequence of model characteristics that lowers performance with larger data input. k-NN performs best with a low number of features: when the number of features increases, then it requires more data. When more data is introduced, the potential for overfitting becomes a problem because it is not known which piece of noise will contribute to the model. The value of k which has no label is predicted to belong to the subset within the group of the majority of labeled neighbors, typically a square root of the total number of observations. In this case N = 16 of 1055 questions times the five features for 5275 calculations to make a prediction. Note that because k-NN involves calculating distances between data points, we must use numeric variables only. This only applies to the predictor variables. The outcome variable for k-NN classification should remain a factor variable. Despite the drawbacks, relative terms such as 'large' number of features may not apply to this study, and response times could also vary depending on the volume of the dataset. Evaluating the fit of k-NN is based on prediction accuracy with an eye on efficiency suitable for a game environment.

Labeling and Processing Data

A separate single column from the data are the prediction labels provided by the tree structured data. For every fifth question, a label indicates the three categories of prediction (hard, easy, or stay). These labels were determined based on the actual performance of the toy during the tree structured study and were verified manually for accuracy. The criteria for each set of five questions is based on a frustration score that is less than 200 average of the harder label, greater than 300 for the easier label and any score between 200 and 300 is labelled 'stay.' The labelled output corresponds to the softmax activation layer on the neural network.

In addition to labeling the classifiers for the model, the data must be scaled to accommodate the features on different metrics. For example, the "pressure" variable is on a much larger scale than "difficulty," which could be problematic given the k-NN relies on distances. The model uses a 'scalescdale' function here, which means pressure values are scaled to a z-score metric, the distance from a value to the mean value of that group. The data is split into two partitions, 70% of the data into the model set and the remaining 30% into the test set of new observations. There are several rules of thumb to determine the number of neighbors (k), one being the square root of the number of observations in the model set. In this case four neighbors, which is the square root of the sample size N = 16, is conveniently a whole number. From the labeled data, which is taken from the dependent variables in the model set, the nearest neighbor is measured to the test data, which is unlabeled or is predicted relative to the location to its neighbors.

$$\text{dist(p1,p2,p3,p4)} = \sqrt{(w1 - w2)2 + (x1 - x2)2 + (y1 - y2)2 + (z1 - z2)2}$$

The results of the Cross Table indicate that our model did not predict difficulty very well. To read the Cross Table (Table 9.2), we begin by examining the top-left to bottom-right diagonal of the matrix. The diagonal of the matrix represents the number of cases that were correctly classified for each category. If the model correctly classified all cases, the matrix would have zeros everywhere except on the diagonal. In this case, we see that the numbers are quite high in the off-diagonals, indicating that our model did not successfully classify our outcome based on our predictors. To examine the success of the classification given a certain category, one reads across the rows of the matrix. For example, when reading the second row, we see that the model classified 2 of 10 "L2" cases as "P1", 6 of 10 "L2" cases correctly, 2 of 10 "L2" cases as "P3," and 0 of 10 "L2" cases as "P4." Where L is the label and P is the prediction category.

Validating k-NN

Using Kappa as a validation method, we see the peak of predictions versus observed outcome. The kappa statistic is a measurement of the agreement for categorical items (Thompson, 2001). Its typical use is in assessment of the inter-rater agreement between predictions and observations. Here kappa can be used to assess the performance of the k-NN algorithm, formally expressed by the following equation:

$$kappa = \frac{P(A)-P(E)}{1-P(E)}$$

where P(A) is the relative observed agreement among raters, and P(E) is the proportion of agreement expected between the classifier and the training target by chance. In this example, the tabulation of predicted and observed classes are as follows:

Table 9.1 k-NN Cross Table of Predictions

	Prediction			
Test Label	1	2	3	4
1	29	0	0	0
2	2	6	2	0
3	0	1	10	1
4	0	0	0	3

The k-NN algorithm predicts 1, 2, 3 and 4 to be 31, 7, 12 and 4, times adding the columns of predictions, divided by the total number of observations. Thus, the probability that k-NN returns for 1, 2, 3, and 4 are 0.62, 0.14, 0.24, and 0.8 respectively. The ground true values are those observed or calculated from current data; here the observed values in the rows for 1, 2, 3 and 4 are 29, 10, 12 and 3. The probabilities are: 0.58, 0.2, 0.24 and 0.6 respectively. Then, the probability that both prediction and the observed (classifier) say 1, 2 and 3 are 0.62×0.58=0.3596, 0.14×0.2=0.028, 0.24×0.24=0.0576 and 0.8×0.6=0.48. Using this example, the overall probability of random agreement is:

$$P(A) = (29 + 6 + 10 + 3)/54 = 0.96$$

More specifically, if there are n_neighbors = 4, the closest four points to the next observed point on plotted data is the distance between them. The k-NN method uses Minkowski distance to determine the difference between the distances that can be small, which is equivalent to calculating the Euclidean distance between the next observation and each of the points in the model data. In this example:

$$P(E) = 0.3596 + 0.028 + 0.0576 + 0.48 = 0.9252$$

therefore, the kappa statistic is:

$$kappa=P(A)-P(E)1-P(E)=0.96-0.9252-0.9252\approx0.89$$

The closer to 1 means there is agreement between the tree structure and the machine learning predictions. The k-statistic is the proportion of total observations in which, in this case, software predictions agree about the finding. For example, if the two predictions were made on 100 sequences and both agree that the difficulty should be made easier for 5 players, hard for 5 players and stay the same for 70 players there would be 80% agreement. This does not account for the other 20% which could be one, the other or neither software predicted correctly. It also could be that chance plays a role in the 80%; however, it is widely accepted that the k-statistic is considered reliable if it is at least that high or higher.

The k-NN prediction rate is: Accuracy: 0.6578947368421053
The accuracy rate of the k-NN model developed for the toy only reached ~65%, which is not as good as other models despite its high appeal for real data that can be noisy. Given the other drawbacks to the model fit and scale of the data, there are others better suited for the toy.

K-mean Model

K-means is an unsupervised algorithm that generates clusters based on the similarity of a data point to a centroid within the cluster. The number of K clusters is predetermined based on the prediction classifications needed. Evaluation of clusters is based on similarity within the group and overlap between groups. More precisely, the aim of the algorithm is "…to minimize the Within-Cluster Sum of Squares (WCSS) and consequently maximize the Between-Cluster Sum of Squares (BCSS)", (Alghofaili, 2021). Categorical data and continuous data make up the features in the dataset; difficulty – categorical (1-7), pressure – continuous, motion – continuous, duration – continuous, and performance – categorical binary.

K- Means Model – Code 9.1 R statistics software

```
preprocessor = Pipeline([("scaler", MinMaxScaler()), ("pca", PCA(n_components=2,
random_state=42)),])
# Train K-Means clustering model
n_clusters = 4  # You can adjust the number of clusters based on your needs

clusterer = Pipeline([("kmeans", KMeans(
        n_clusters=n_clusters,
        init="k-means++",
        n_init=50,
        max_iter=500,
        random_state=42,),),])

pipe = Pipeline([("preprocessor", preprocessor), ("clusterer", clusterer)])

# Call the pipe function to run Pipeline
pipe.fit(features)

# Add the cluster labels back to the DataFrame
preprocessed_data = pipe["preprocessor"].transform(features)
predicted_labels = pipe["clusterer"]["kmeans"].labels_
silhouette_score(preprocessed_data, predicted_labels)
```

The scale for each of these clustering performance metrics ranges from -1 to 1. A silhouette coefficient of 0 indicates that clusters are significantly overlapping one another, and a silhouette coefficient of 1 indicates clusters are well-separated. In the data set features, the silhouette score is 0.6531743083708955.

Classification for K-means

Performance prediction according to Rivas et al. (2022) is one of the most important aspects of classification between groups. Typically, a binary classification process is used however the same process can be implemented for multiclass identification where more than two groups are required. “Statistical and probability techniques (Thomas, 2004), as well as machine learning methods, can be used to carry out the [classification] process. Following an approach based on machine learning models, the most common classification techniques are tree-based models" (Briemen, 2017). Regarding the evaluation of the different models, there are several types of metrics that can be used to assess how good is the classification. Some of the most used models are precision and recall. "In binary classification, the precision is the proportion of positive classifications that were correct, in the same way, the recall is the proportion of real positives that were correctly identified. With these two metrics, we obtain what is known as F1-Score which is the harmonic mean of precision and recall. Thanks to these measurements it is possible to know how the classifying model behaves. In addition, the so-called confusion matrix is used to evaluate the deficiencies of a model.", Rivas et al. (2022)

Although classification and labeling are often used interchangeably, here the distinction is that the label identifies the data for each set of five questions within the three levels of difficulty. This was done manually for supervised models such as K-mean and ANN. Classification on the other hand is a concept of combining data from the five sequences but not having a fixed set of criteria for each variable. Although using the term clustering Saxena et al. (2017) describe the process this way, "clustering divides data patterns into subsets in such a way that similar patterns are clustered together." Once all the test observations are classified (clustered) we can see how the model is performing. Each set of five was validated for accuracy however the criteria for what is 'accurate' is debatable however consistently applied in distance from the centroid. As previously discussed, the frustration score calculated in the TS software is a floating value. The number of sequences needed to change the difficulty level varies depending on the variables and thresholds of the prediction. The K-means model should therefore perform better because each set of five is evaluated without the influence of the previous sequence.

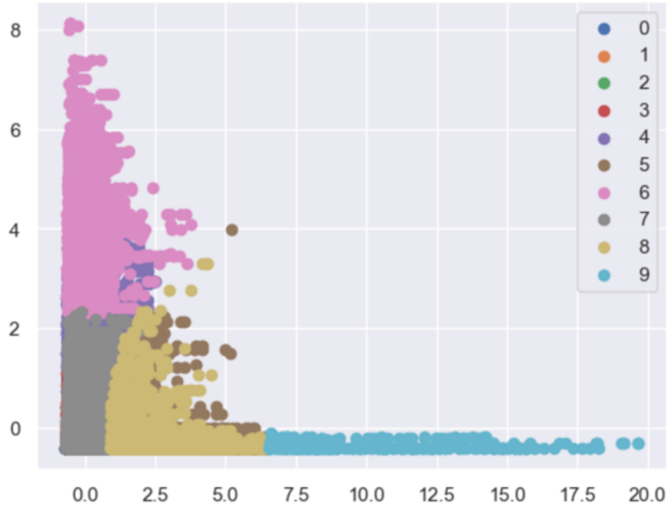


Figure 9.2 k-Means skewed distribution scatter plot of sequence predictions.

When prolonged ‘stay’ predictions were found in the TS data for example, player performance was calculated to assess the likelihood of the player who continually struggled (scored less than 3/5 over multiple sequences) or showed mastery (scored more than 3/5 over multiple sequences). Recalculation of the frustration score determined any adjustments made to the prediction. Using the adjusted data set the K-means model did not perform significantly better than the k-NN or as we will see, the ANN model. Other than more data we could adjust the centroid of each cluster to better define the group, however this is computationally expensive and not feasible for active game play.

ANN Model

TensorFlow and Building an ANN

Early investigation of AI systems considered five different frameworks to work with that included neural networks. Three top contenders were selected for their large community support, resources, and the fact that they were considered to be more ‘established’ platforms. These criteria narrowed the field to include FaceBook’s PyTorch, Microsoft’s Cortana (which as of this writing has changed to Azure) and Google's Tensorflow. In the end, I almost selected PyTorch. It is an open source, deep learning framework built for research, considered to be stable and includes technical support needed for production deployment. It enables fast, flexible experimentation through a tape-based autograd system designed for python-ish execution. Autograd is the package name for automatic differentiation, which is explained at this link in detail.

https://justindomke.wordpress.com/2009/03/24/a-simple-explanation-of-reverse-mode-automatic-differentiation/.

In short, it is a way to record to a tabletape, a single row reference, values found, in this case neural network values such as x1, x2,… xN. The values are traditionally calculated in reverse order i=N-1, N-2,… 1. The recorded tabletape allows for quick referencing and relatedness to the values in real time. With the release of PyTorch 1.0, the framework also offers graph-based execution, a hybrid front-end allowing switching between test modes, distributed training for very large data sets, as well as being an efficient mode for mobile deployment. PyTorch was promising at first, but support for IoT objects was lacking, other than mobile phones, and for political reasons we abandoned the framework for other software. Cortana/Azure also did not have much support for IoT development, in part because the focus of many commercial projects has been on speech recognition, image identification, and natural language processing. Many of these features relied on built-in functionality of cameras and micro controllers attached to smartphones. Microsoft's 'insider' lab provides development support for IoT and AI; however, it is intended for entrepreneurs looking to kick start a business rather than researchers wanting to experiment with the platform. Google's TensorFlow was therefore selected, because of its specific IoT functionality built to include non-phone specific devices. Google also has a great deal of online resources and technical support for the software that made implementation and development go quickly.

Unlike the clustering algorithms of k-means and k-NN, Artificial Neural Networks can be described as a classification algorithm that uses a layered schema where perceptrons (Rosenblats, 2021), the numeric signal of data, are passed between layers to an output layer that has a binary or multiclass prediction. A list of output layers more commonly used in machine learning include: Sigmoid, ReLU, Swift, and Softmax. The process of training a machine learning model is to find the optimal parameters that can accurately capture the relationship between features X and Y. To achieve this, a training dataset $D = \{x_i, y_i\}$ N i=1 with N samples is needed. Then a loss function L is adopted to quantify the difference between two outputs, i.e., the ground-truth one y_i and the predicted one $f\theta$ (x_i). The goal of training a model is to minimize this loss function without overfitting, where the model no longer predicts but applies a precalculated answer. To improve prediction power, adding an activation function using ReLu, Sigmoid, or other neural-network layers is a combination of running multiple training cycles, changing the number of Epochs and density (D) values in the activation function.

Because this study has multiple classifications of output, Softmax as the final layer preserves the number of predicted classes and in this case three are defined. ReLU (Rectified Linear Unit) is added to the hidden layers of the network. Although the term 'linear' gives an impression of a linear function, ReLU has a derivative function and allows for backpropagation while simultaneously making it computationally efficient. Softmax uses a normalization strategy where the sum of the input vectors are set to equal 1 without losing the relationship between numbers. Euler's constant is used with the exponent input values to form the k probability distribution. This value is used to calculate the decay or growth of a particular factor over time.

$y = e^x$, where y is the output, e is Euler constant = 2.718281828459045, and x is the input

The relationships between all the input vectors are assigned weights normalized and summed (N-1) through to the next neural network layer. In this model, each of the three layers having an activation function; two use ReLu, which processes multiple neurons into the previously discussed output layer, SoftMax. This output layer is a prediction distribution measured against the trained target output distribution. According to Harrell (2001), if labels are integer or float values, then you need to follow sparse categorical entropy as the loss function. The loss function is the difference between the predicted and the actual outcome distributions, and it is used to adjust the weights of the parameters in the model to minimize the loss. The loss function aims to find the optimal set of weights that most accurately predict the output. More specifically, the sparse categorical entropy is simply the 3-class categories (easier, stay, or harder) of the output prediction once the loss function is applied. The main issue here is that the ReLU function does not activate all the neurons at the same time. Because the activation function in the final layer for a multi-class problem will always be SoftMax, the only difference in configuration lies in which loss function to choose.

TensorFlow has two different loss functions for multi-class classification and the choice is determined by the format of the data. **SparseCategoricalCrossentropy** computes the cross-entropy between the target and predictions when there are two or more labels within the target class. This loss function expects the labels to be provided as integers in a single vector. If the target variable has labels in the one-hot representation, **CategoricalCrossentropy** must be used instead. There is an important distinction between these two loss functions that needs to be considered when compiling a model for training. It is important to choose the correct corresponding cross-entropy based on the target variable’s data layout.

How the Machine Learning Model is Trained and Tested

Toy interactions are determined by the start of a single question and end by the child’s last input of the answer. The number of interactions needed to train the machine learning model varies based on the model; however, a benchmark of 10,000 interactions was set, based on the literature for model designs of this type. More recently, smaller sample sizes have been supported using data strength and quality tests for criteria. Reported by Rajput et. al., “increasing the data quality (50% then 100%) exhibited significant improvement in effect size (> 0.5) and ML accuracy (> 80%). Importantly, small sample sizes with 100% quality showed good effect sizes (around 0.9)”, indicating that quality rather than scale is important for ML predictions. Rajput et. al. go on to say that for machine learning performance, sample sizes between 100 – 1000 were considered large. During the TS study the toy collected over 90,000 lines of data for a data set of 1055 questions, split between the training and test set in a 80% - 20% respective ratio to optimize the model’s probability power. From the 844 questions (168 sequences) captured, a suitable training set of data for a confident ML model was achievable using two toys. Equal number of play sessions is unlikely from each toy nor from each engagement group; however, from this high number of samples scientific thresholds for analysis were obtained. The emphasis on data quality over quantity is an important consideration and one that evolved since the inception of this research.

There are several different machine learning techniques that have been used in educational tools and teaching aids. Framing of the problem within the probability parameters of machine learning necessitates an outcome that only has a limited number of possibilities. In this study, the output is the ML selection for the next set of questions to ask the player, which does two things; the first is that it provides a future focus of the game which tailors the game play for each player for what happens next. The value of this approach is the players responsiveness to increases, decreases, or remaining at the same game level, however, that responsiveness is tailored to each player. As changes of the game occurred during play, for example, response times needed to be immediate for the game to feel continuous. Question times are still relatively fast in responding to the input of the toy; however, predictions are calculated based on sequence results, which is an added functional consideration prior to ML predictions. Many of us have experienced the lag in first person shooter games, for example, where an interaction that produced some type of collision is delayed. Now imagine incrementing a point on the scoreboard to be dependent on the response rate, as well as the results of the interaction. Machine learning processes the physical interactions as well as point calculations (duration, performance, etc.) in order to make predictions in real time.

Expected Predictive Rate

As of Feb 10, 2024, the ANN model returned the following results:
50 epochs using sparse categorical cross entropy optimizer: Adam
Test Loss: 0.8842930197715759, Test Accuracy: 0.6005434989929199

Adding second dense layer with activation layer of ReLu and final layer of Softmax
Test Loss: 0.821271538734436, Test Accuracy: 0.6507320404052734

Changes to optimizer: adagrad
Test Loss: 0.8830541968345642, Test Accuracy: 0.6053682565689087

Changes to optimizer: SGD Stochastic Gradient Descent
Test Loss: 0.837097704410553, Test Accuracy: 0.6457409262657166

Changes to optimizer: RMSprop Root Means Square Propagation
Test Loss: 0.8030046224594116, Test Accuracy: 0.6660381555557251

Adding third dense layer with activation layer of ReLu and final layer of Softmax, optimizer: RMSprop

```
Test Loss: 0.7623239159584045, Test Accuracy: 0.6766858696937561
```

A final change to the first density layer from 16 to 32 improved the accuracy of the model to .0682 with a loss function of 0.753. A prediction rate of 68% is not very high; however, given the data set training and relative performance over the other cluster models ANN was decidedly the best model for this type of data and well suited for future modifications.

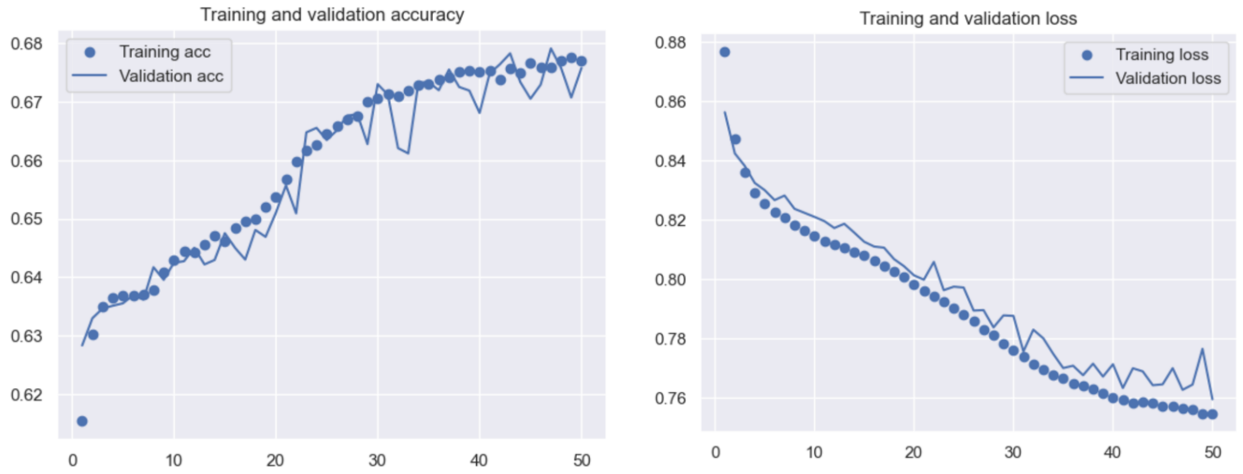


Figure 9.3 Training and Validation Loss and Accuracy Test for data fit.

As demonstrated in Fig 9.3, model accuracy is influenced by the closeness of fitting data, the number of layers in the model, the activation function, and the quality of the data itself. Literature references for prediction rates vary a great deal depending on the type of data and subjectivity of the decisions such as thresholds as a criterion for inclusion or not. Similar studies using IoT devices and multivariate data were closely scrutinized to set expectations for the model's performance. One such study is the *Affective Learning*, Picard et. al. (2004) "…data from the learning experiences, we developed a system that achieved an accuracy of 76% on affect category recognition from chair pressure patterns, and 88% on nine 'basic' postures that were identified as making up the affective behaviours." Affective learning — a manifesto (Picard et al, 2004). Because of the experimental nature of this study there was less concern for the year Picard's study was conducted taking into account the scope and complexity as something parallel. The goal of 75% accuracy seemed attainable given the advances in tools and computing power. The following sections describe how close to the goal I came and in defense of what was achieved.

Machine Learning Session Results and Comparison

In Machine Learning as an Experimental Science, Langley writes, “various performance measures are the natural dependent variables for machine learning experiments, just as they are for studies of human learning” (1988). Machine learning performance measures predicting the difficulty level accurately addresses RQ3 and RQ4 directly. As we evaluate the toy performance the dependent variable has shifted from the performance of the player, which weighs heavily on the outcome to the toy. The predictions made by the model evaluate the probability of difficulty is appropriately selected for the player to remain engaged, at a targeted level that challenges a player's performance. To show improvement we use the TS data as a benchmark for comparison and assume there is no difference between the two software. This assumption can be formulated as a null hypothesis that the machine learning software performs no better or worse than software that makes pre-predictions like the TS version. Chapter 8 describes the tree structured software and the evaluation of the ‘frustration score’ but how does it compare to the continuous evaluation of machine learning and its prediction? On the surface a direct performance indicator of the game versus the players performance would be sufficient to answer RQ3, however, a more valuable assessment is the relationship between the independent variables and that of machine learning performance is an indicator impact on player results. Statistical models such as MANCOVA not only allow for multivariate input but also the significance and strength of the variables on the output.

Comparison between Software
Influence on Player Performance – right vs. wrong
Variables of data – duration, difficulty, motion, pressure
Influence on Toy Performance – sequences of play: easy, stay, and hard
Variables of data – duration, difficulty, motion, pressure, and player performance

Table 9.3 shows the number of correct answers in relation to total questions and total sequences that had a prediction assignment. Generally, students played more using the ML software with an average of 14.1 sequences, an increase of 1.6 sequences or ~8 more questions.

Table 9.2 – Each ML participant’s session, correct answers over total questions within the total number of sequences

	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34
Correct Answers	20	17	6	5	17	21	110	18	16	45	86	21	32	17	20
Total Questions	52	44	17	13	58	58	196	37	26	69	144	40	54	37	50
Total Sequences	11	9	4	4	14	13	43	8	6	14	30	8	13	8	12

Both groups scored the same number of questions correctly with a mean of 30 for the TS group and 30.5 for the ML group. In the ML data there is no frustration score as there is in the TS group because the model is determining the difficulty level. The level of difficulty is high, and the players are performing at an average level. In the TS model we saw levels taper slowly and the player responded accordingly, improving, and struggling more or less based on the difficulty. The machine learning model tends to keep the level increasing even when their answers suggest they should stay or decrease. However, students seemed to respond to the challenge and maintained their performance level even after multiple increases in difficulty. The model only changes its prediction after multiple sequences of lower scores, as we see in participant 26, and 30 (participant 28 is similar but with a small sequence size).

The machine learning (ML) data differs from the TS data in that fewer predictions of 'stay' occurred with the toy responding more quickly to good or poor results. The total number for 'stay' predictions is 47 – approximately 36% of the total predictions for ML – compared to 77 – approximately 46% for TS. The percentage of harder questions was also larger for ML at 36% with TS at only 25%. Lastly, there was a less than one percent difference between the 'easier' predictions at 27% for both software. The number of easier, stay, harder predictions for ML was 35/47/46 versus TS predicting 46/77/43 respectively. The representations of sequences also indicate a difference, not only in the number of 'harder' predictions, but also as students performed relatively well under harder questions, or the drop after levels were increased was more gradual. Similar patterns of gradual increases in performance at easier levels is also evident as it is in the TS performance data. In these three examples, we see that scores of 0 correct responses occurred only slightly more in the machine learning set – TS: 12% and ML: 13%.

The purpose of correlational research is to investigate 'the extent to which differences in one characteristic or variable are related to differences in one or more other characteristics or variables' (Leedy and Ormrod 2010). A correlation occurs if one variable (X) increases, and another variable (Y) increases or decreases. A study that produces a correlation coefficient of 0.00 signifies that there is no association between the variables investigated. Analyzing the machine learning data requires a multivariate approach to consider not only the relationship between features but also the correlation between the features and the three probable dependent outputs. The research design considers two groups: the tree structured group and the machine learning group.

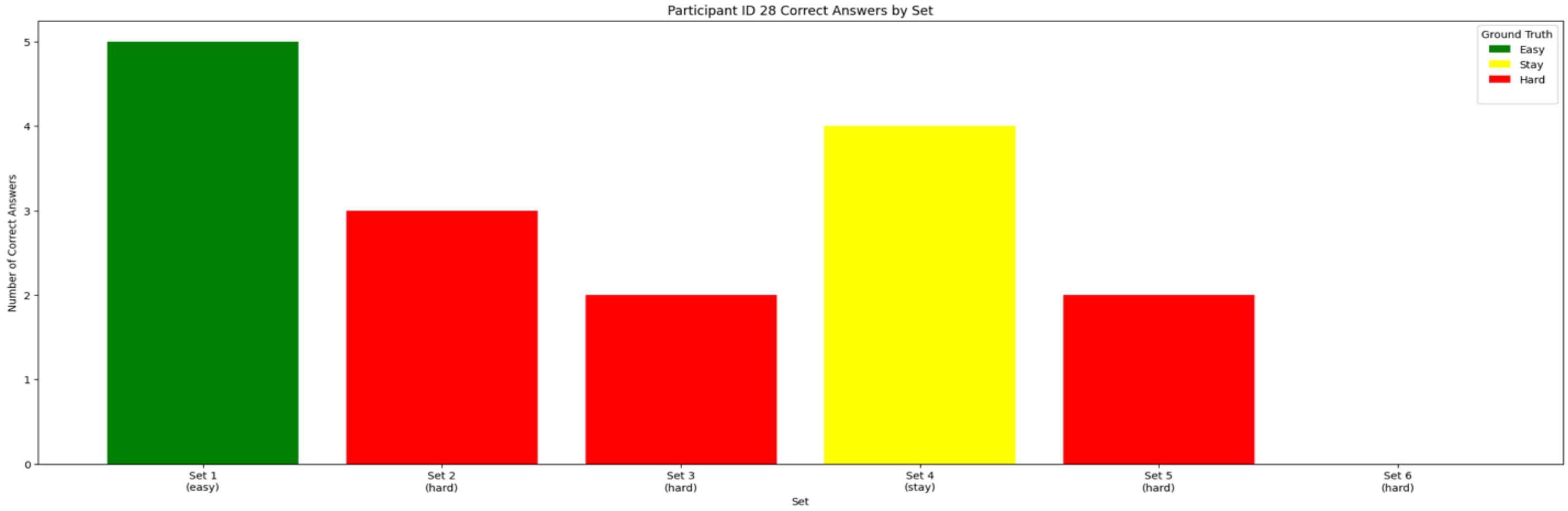


Figure 9.4 Predictions of each sequence (easy, stay, hard sets) and performance participants 26.

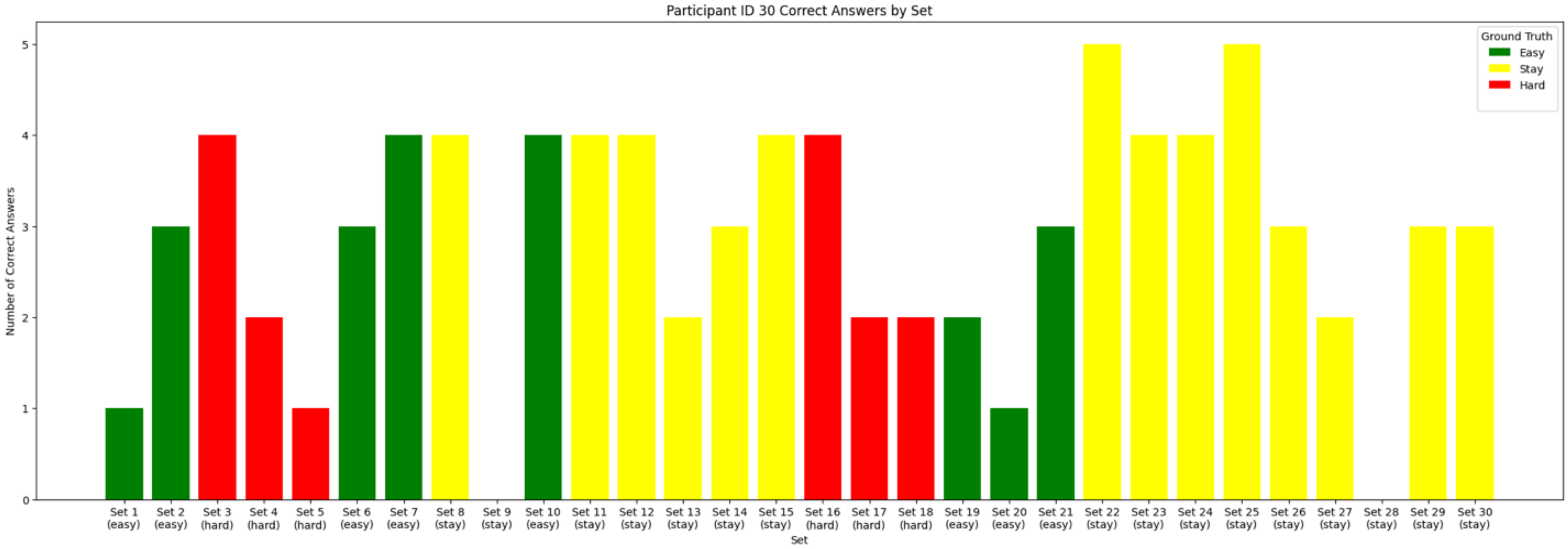


Figure 9.5 Predictions of each sequence (easy, stay, hard sets) and performance participants 28.

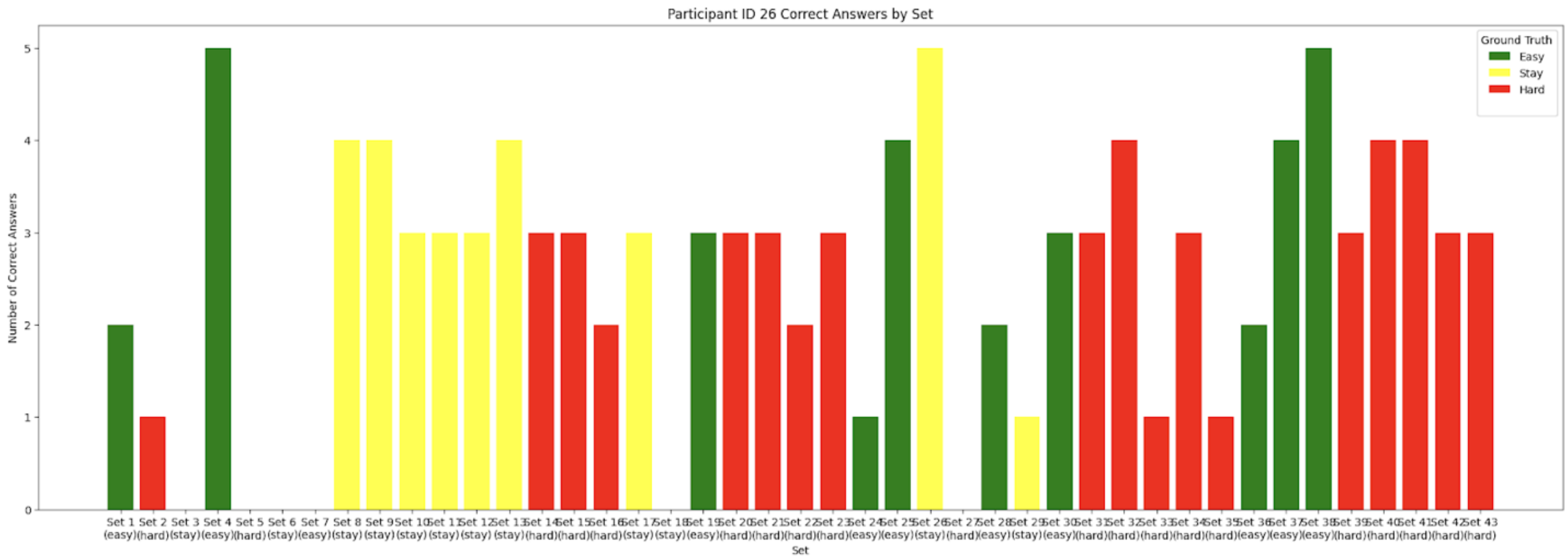


Figure 9.6 Predictions of each sequence (easy, stay, hard sets) and performance participant 30

Thus far the correlation between variables has only considered four of the five data points in chapter 8. In the TS software player performance has been programmed to have greater weight than other variables, therefore consideration of its relationship to other variables can be seen in an analysis of toy performance. To do this, the toy performance is compared between sessions of questions as well as the difference between groups. The first group of players is divided by session clusters to determine a correlation between player performance and each variable as the number of sessions increases. This analysis focuses on the player's ability to answer the questions correctly and consider the duration and difficulty as part of the calculation within each group. The player's physicality is captured in the motion and pressure levels to see their relationship to player performance as well. A comparison between variable influence on TS and ML software is discussed in Chapter 10 however variables for both software versions are the same to facilitate comparison.

A second grouping is based on the toy's prediction performance. The clusters in the group distinguish between the level of difficulty predicted by the toy for the next sequence of questions (easier, stay, or harder). Each group is measured based on a number of predictions in each group and the features that determine the prediction for each sequence. The sequences are then used to calculate the differences between the other input variables (see Fig. 8.1). The play sessions are defined by a series of 5 questions within a game (color or math) with a transition to the second game also containing five questions. Participants will switch from one game to the next as they continue to play, increasing in difficulty. One session - the minimum participation threshold, five sessions – will be considered a moderate level of engagement, and ten or more sessions are the highly engaged group. Some participants will not attain 'session' status if they interact with the toy less than ten times or without a transition between games. The participants who play with the toy long enough to experience at least one transition from the color game to math will be placed into the first group, 1+. As they continue to play with the toy, they will remain in group 1+ until five transitions are reached; at this point, they will move into group 5+. This form of organization is said to be 'in-group' because members of the 1+ group will be members of the 5+ group. Not all participants will be placed in all group levels because they may not all engage with the toy to higher levels of sessions. It is also the reason for the ranges of sequences to include the data collected from sequences that are less than five but larger than one.

Conclusion

In this chapter, comparisons between models, software, and individual variables provided evidence to support decisions selecting models, testing the shape of the data, and the method for comparative analysis. Comparisons within each software also provide some insight into the performance of the toy relative to the performance of the players. In broad terms, we see a difference in how the software responds to the players more dynamically in the machine learning model and is slower to react than the tree structured version. Patterns in the data only begin to emerge clustering around characteristics such as light and heavy button pushing, and the shapes of motion. What is still missing from these observations is the statistical analysis to determine the combined effect on variables such as performance, between software, and for predictions. Lastly, the model is validated in rejecting or accepting the null hypothesis followed by their interpretation and what it means for future studies with the toy.

Chapter 10 - Analysis and Discussion

Designing For a Data Collection System

The design process used to create the toy discussed earlier in the literature review (see Chapter 2), provided a framework to map out a plan. Coinciding with this research which is preliminary and experimental in nature due to the technologies implemented, there are no theories readily available to ground the design decisions other than what can be drawn from related fields. Assumptions must be drawn from and at times adapted to the current context and the qualities of the toy, its design, and the examination of outcomes. The specific decisions concerning the design of the toy and its parts, such as the buttons, handles, and display of lights, each represent tradeoffs, which together shape the result, unlike games designed and developed for interactive devices such as smartphones, tablets, and laptops, a toy that is networked into the Internet of things (IoT) has its own unique set of challenges that these other devices do not.

The most obvious of these challenges is the creation of content as feedback to the user. On a typical electronic device, text, images, symbols, etc., can all be generated on the screen, which has shared characteristics with these devices. An IoT device needs to provide the infrastructure to accommodate the content as well as provide a context for the content in the design. For example, if a button needs to display multiple symbols to indicate the mode or function of the current state of the button, it will need to be equipped with lights or a display of some kind to afford the communication. A simple solution may include a screen like other devices; however, that would negate the inherent principle of embedding the learnable elements into the interaction, adding one more layer separating the user from the content through simulated physical interaction. It would also have the effect of focusing the user's attention on the screen rather than the toy, as well as building in feedback through the display, which again shifts the user's attention toward scorekeeping or time checks that detract from learning, a type of over gamification (Thibault & Hamari (2021).

The toy's central light – labeled 'Game Mode' in the game instruction manual (Appendix M) – has a few communication functions dictating its physical design and layout. The first indication of a solid white light indicates the toy is operating and turns on when the power switch is moved to the on position. Once the Wi-Fi network is located, the toy flashes a solid light to indicate it is connected. During the color sequences, the light shuts off, but during the math sequences, it is used to indicate the math operations to understand the sequence. For example, button 5 – game mode '+' – button 2 – game mode '=' – button 7 is the sequence. The game mode button must display a solid light to indicate on (the flashing light is still solid and is included as one function), a '+' sign to indicate addition, and an '=' sign to indicate the answer to the sequence. The design of this single light indicator uses 9 mini LEDs to perform the functions but should not be interpreted as a button. During the paper prototyping sessions, the game mode light was raised, and although smaller and less prominent than the buttons on the toy, participants would often press it during the math game. The grid layout allowed for a variety of symbols to be including add, subtract, multiply, and ‘H’ for incorrect.

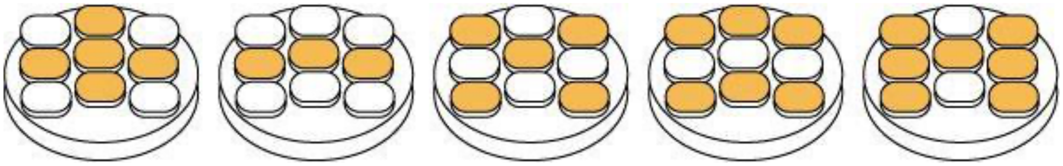


Figure 10.1. From left to right: plus, minus, multiply, equals, and error.

The choice to include this central light was made early in the design however, the different symbols and lighting between button signals is in direct response from the paper prototype study. Most participants struggled with the meaning of the symbol and provided feedback on when and what the lights should be triggered.

Button Design

From the diagram of the buttons in Figure 10.2, the design is multifunctional, 3D printed using rigid and elastic polymer and electronics. Each aspect of the button has been considered, from the shape of the button for stability and scale for a child within the target group to the communication needs and requirements for a seemingly simple component of the toy.

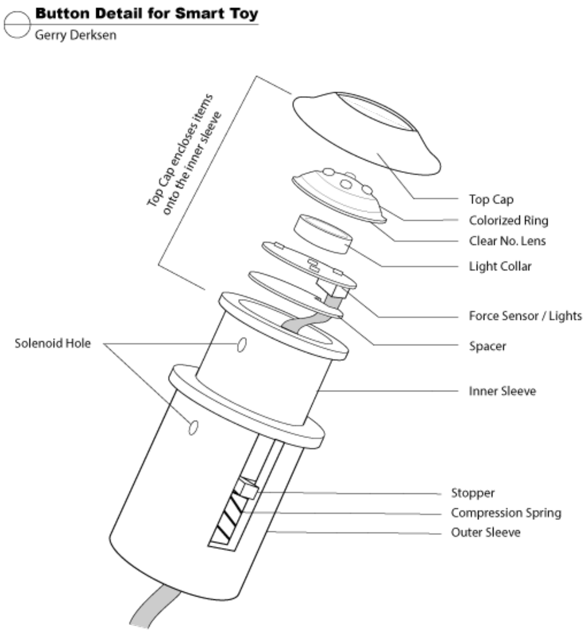


Figure 10.2 Button Diagram Exploded View

Early versions of the button were designed to include a dual light system that indicates color or number; however, these versions were not easy to implement into the toy and, at the same time, provide the functionality that was required. For example, the force sensors embedded in each button were positioned at the top to collect the pressure data applied as a measure of force over the duration of the press. The advantage of this is to distinguish not only the force applied but also how quickly it is applied as a characteristic of force that may provide insight into the emotional state of the user. Placing it at the bottom of the button, the data becomes shortened instance to a single reading, collecting less information and providing less rich data. Placing the sensor at the top implies that ranges of pressure and duration are parsed separately but identifies a specific characteristic when taken together. When categorical data becomes more granular, the result is continuous data that is more nuanced and potentially able to distinguish subtle differences in emotional expression.

Lights surrounding the button rim indicate the color within the sequence. The full-color list includes turquoise, blue, red, yellow, orange, purple, green, white, and pink. The position of the lights did not change much from the paper prototype to the ML version of the toy. The only significant issue considered in the lighting design was that neither color nor number should appear to have greater importance over the other. However, the brightness levels of the numbers shown in white light appear higher in value than the color button lights. In contrast, the colors are unique to each button – a form of novelty that engages the user equally.

Similarly, the numbers were placed in the middle of the button, where the colored lights were placed around the outer edge. The scale of the colors was larger than the numbered areas to equalize the visual importance. These positions were based on the readability of the numbers clustered in the center of each button.

Pre-test Analysis

As described in Chapter 6, the number of participants for this study was thirty-two between the ages of 5 and 10 years of school age and were students at a school specializing in education for autistic children. They were divided into two groups of 16 participants, each providing an equal number of the two software versions to be tested. Table 10.1 describes the participant's pre-existing skill level and range for both groups, which were established using a pretest to determine their level of understanding of color labeling and numbers and simple math questions.

Table 10.1 Mean values for pre-test as well as standard deviation and percentages of players

	Color Q1 Ident. 4	Color Q2 List 10	Color Q3 1 Fav.	Math Q1 Ident. 5	Math Q2 Add 2	Math Q3 Sub. 1
Mean TS	3.3125	2.625	0.9375	4.3125	1.3125	0.5625
SD - TS	1.352467	2.446085	0.25	0.7932003	0.9464847	0.5123475
Mean ML	3.625	2.9375	0.8125	4.4375	0.9375	0.375
SD - ML	0.6191392	1.388944	0.4031129	0.813941	0.8539126	0.5
TS + ML Mean	3.46875	2.78125	0.875	4.375	1.125	0.46875
lowest to highest distribution	2/32 = 0		12.5% - 4/32 = 0	12.5% - 4/32 = 3	34% - 11/32 = 0	53% - 17/32 = 0
	8/32 = 2		87.5% - 28/32 = 1	28% - 9/32 = 4	22% - 7/32 = 1	47% - 15/32 = 1
	7/32 = 3			59% - 19/32 = 5	44% - 14/32 = 2	
	21/32 = 4					

The first three items asked questions about color, such as identifying the four colored squares provided, listing ten colors you know, and asking what your favorite color is. The mean value of both the tree-structured group (TSG) and the machine learning group (MLG) is less than 0.3 for all the color questions. The TSG scored slightly less on the first two questions – identifying the color squares provided and listing up to 10 colors they know – and the MLG was also better at identifying their favorite color. Only one student identified ten colors, and the highest number was 5, which was listed by 4 participants, with an even distribution below 5. Excluding the outlier who could identify 10, the average answer was 2.54 of 5 colors, which was less than the score for the question on identifying colors, which was 3.48 of 4.

Number identification was even better than color identification, with an average score of over 4 out of 5 numbers, and the standard deviation is close to the mean, less than 1 in both groups. This is likely due to the representation of the numbers using pips similar to the buttons on the toy. Observed in the sessions, many players counted the pips on the dice to get the right answer. The MLG did slightly better with simple math questions, which included addition and subtraction: on average, 21 of the 32 students scored 50% or higher on the addition question, and 47% answered the subtraction question correctly. Although it was not critical that all participants fully understood math, only 25% of the students could not answer any of the math questions; however, all the students could identify at least 3 of 5 numbers, and 59% could identify all five numbers. Question 3 for both color and math are binomial distributions with a standard deviation of the math question ~50%, but the favorite color question expectedly skews left p = 0.875 because most children have a favorite color and could name it. With this as a benchmark expected performance of the student's abilities was likely to be widespread. The average difficulty level was 2.4 and 2.7 out of 5 for TS and ML sessions respectively. These results were mitigated by each player first and only time they played the games.

The well-known difficulties that many autistic children have with mimicry are mitigated by using a toy to follow rather than another child or therapist. This is partly evidenced in the duration of play, where the average sequence length was a minute and 40 seconds. Only 3 participants did not meet the interaction threshold: two occurred in the machine-learning group and one in the tree-structured group. All three of the participants had difficulty engaging with the toy due to an experience just before participating or due to distractions in the testing room. For example, one child only played one sequence of colors before discovering the video camera used to record the session. Another participant came from a music class and was not convinced the toy would make noise. Many children with autism dislike loud noises. She played a number of sets with her ears plugged and scored poorly because pressing the buttons was difficult, even though she indicated the correct sequence. Subsequently, these results were discarded from the analysis. These observations and the evaluation of game play in chapter 8 and 9 provide the backdrop for the toy performance analysis. To address the research questions more fully the following path of analysis and testing the validity of the statistic models is not straight forward. Finding an appropriate model that shows if there is a significant difference between the software was a two part process. The first uses MANOVA, a regression model for multivariate data, and Kruskal-Wallis a model for multivariate data that is not normally distributed. Starting with the shape of the data we will see why these two approaches were taken.

Toy Analysis and Findings

In Chapter 9 some evidence of player performance increasing in the machine learning software was seen in Figures 9.5, 9.6, and 9.7. A statistical analysis of the two software begins with the Hotelling T^2 test, a hypothesis test for multiple groups. Rather than a t-test that compares a single independent variable, the Hotelling T^2 test uses multiple variables for both groups of tree-structured players and machine learning players. It is used to evaluate the alternative hypothesis that a more responsive software (ML) to the individual user will perform better than typical software that has been designed for the average of users. Using the player performance variable (right vs. wrong) and speed versus the question difficulty level as independent variables suggests two important aspects of engagement but are only described by the shape of the data and not a direct correlation test. For example, a player who takes more time to answer questions might be described as less engaged with the subject due to inability, difficulty, distraction, boredom, confusion, or other reasons. However, it could be said that a child who takes their time to answer are more engaged, particularly at later stages of play or higher difficulty levels. The shape of the data from the latter player should exhibit more gradual changes between questions, either in reduced time or increases in time relative to correct answers. The player who suddenly gives up or finds the game’s continual changes quickly exhibits a more immediate loss of engagement.

Hotelling's two-sample T^2*-test*
data: dfToy [1:189,] and dfToy [190:378,]
T.2 = 23.573, df1 = 5, df2 = 372, p-value < 2.2e-16

Instead of observing a single variable influence on the two software, we observe the 5 different variables. Hotelling's T-squared provides a formal way to decide whether the observed differences are strong evidence of a true difference between TS and ML from the variables or whether the differences are consistent with typical variation within a group From the results of df1 and df2 we see a large difference between the software groups and a p < 0.5 which indicates we can reject the null hypothesis. This test compares of means of the two software and indicates they are different (ie ML is larger than TS) meaning they performed differently. It also suggests that the MANOVA statistic is an appropriate choice to further test them and validate the statistic with additional metrics.

The Shape of Toy Performance Data and Recalling Correlation

If we recall the Pearson test of correlation for the TS data we found r = 0.3 a weak and generally inverse proportion to each other. MANOVA has multiple assumptions related to correlation, including the absence of multicollinearity and the linearity of dependent variables. There should be no multicollinearity among dependent variables which are the software outputs of difficulty level. To check for multicollinearity, we check the variance inflation factor (VIF) with the following level guides to determine the interdependence of the variables. If the value of VIF is less than 1, there is no correlation. If it is between 1 and 5, there is a moderate correlation. If it is above 5, there is a strong correlation.

```
TS:     duration     motion    pressure performance  difficulty
         1.017097    1.008183    1.040609    1.100039    1.130005

ML:     duration     motion    pressure performance  difficulty
        1.024297    1.015606    1.057977    1.055347    1.023849
```

The VIF results above show the multicollinearity test, with all the variables indicating a moderate correlation to the ground-true values of both kinds of software. The result is close to the threshold for both. Taken separately, one version of the software could outperform the other to meet the assumption; however, this makes for a difficult comparison between the two. Although subtle difference a closer look at the shape of the data offers a hint as to why a significant difference in the hypothesis test but little difference in the correlation test which was also weak for the TS data.

Distributions of Each Variable

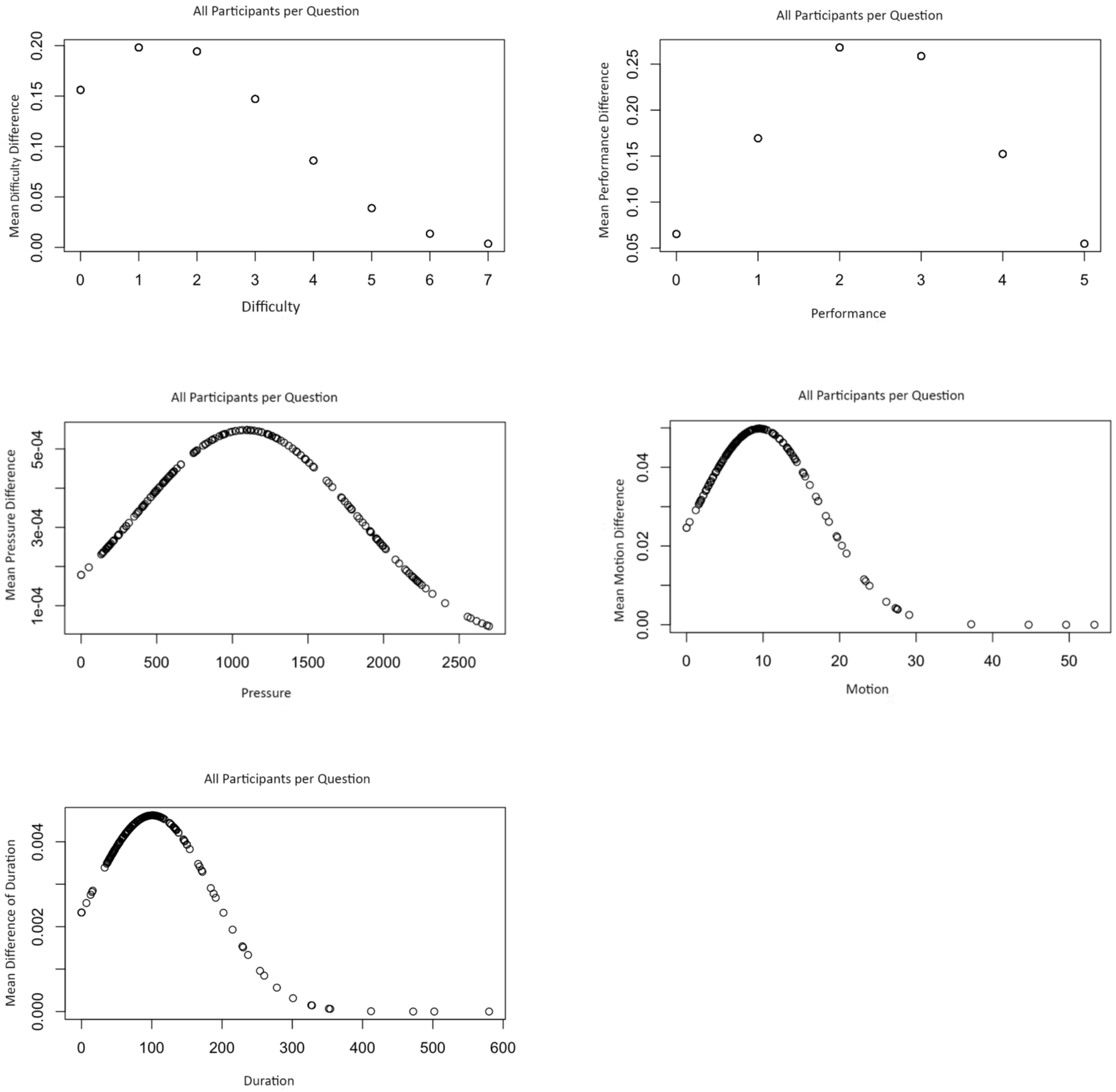


Figure 10.3 Distributions of all participants for five machine learning model features.

All of this assumes that the variables are normal distributions; however, looking at the graphs of the participants in Figure 10.3, there was skewed data, and although parabolic in nature, it is not completely parametric. Using the Anderson-Darling normal distribution test the p-values are below 0.05, which means that no assumptions of the variables were equally distributed between predictions. When dealing with t-test and ANOVA assumptions, you can transform the dependent variable, keeping a uniform relationship between variables, or, in other words, make sure there is homoscedasticity. A common approach is to use log10 transformation to achieve normalization; however, values still fell outside the p-value range, and the data transformation was not going to provide a normal distribution. According to Brownlee, in his statistical approach to analyzing ML models, the following are possible explanations for non-parametric data:

1. Data is not real-valued but instead is ordinal, intervals, or some other form.
2. Data is real-valued but does not fit a well-understood shape or
3. Data is almost parametric but contains outliers, multiple peaks, a shift, or other features.

The data collected was outlined in the previous chapter, and although the sequence level and game difficulty are ordinal, the other data points are real-valued. The data fit is somewhat parabolic; however, non-ordinal data does lean positive without a left tail (see Figure 10.3), suggesting some other features, such as outliers. Two variables were controlled for outlying values for duration, where periods between players were reset to 0, and pressure values below 40 were not used to calculate the prediction of sequence level. The most convincing evidence was the Shapiro-Wilk and Anderson-Darling tests, both of which indicated that the data was not equally distributed.

Reconsidering the Assumptions of MANOVA

In order to use MANOVA statistic model the following assumptions must be met:

- Observations are randomly and independently sampled from the population.
- Each dependent variable has an interval measurement.
- Dependent variables are multivariate normally distributed within each group of the independent variables (which are categorical).
- The population covariance matrices of each group are equal (this is an extension of homogeneity of variances required for univariate ANOVA)

We have met the first two assumptions but question the normality of the data. From the distribution plots in Figure 10.3, it appears that the variables are near normal; however, with further testing it is clear, they are not. At a second glance the data may be skewed, or Poisson where the intercept is greater than 0. Transforming the data was attempted using log10 and log5; however, these methods only pushed the curve to the other side of the plot. Using the Shapiro-Wilk's statistic we find the following determinant.

Testing positive determinant in R

```
covPerformance <- cov(df)
det(covPerformance)
1.418705e+12 = 66.48217084991465
```

From Machine Learning Statistics (Dangeti, 2017), a positive determinant is a value that indicates a matrix's orientation. That is the direction of the independent variable vector, which, after it is transformed, will take a directional space. A positive determinant means the matrix is oriented in a specific direction. A matrix is positive definite if it is symmetric, and all its pivots are positive. The determinant of a matrix is the product of eigenvalues. If all eigenvalues are positive, then the determinant is also positive. Because the positive determinant is 66.48, we can assume the values in all the features are positive because a number of them are ordinally constrained to real numbers. Determinants can only use complete matrices which is one of the reasons for using peak values of pressure and motion and sequence comparison which is a complete set of predictions.

The means of the tree-structured and machine-learning groups on the pre-tests of skill competency was calculated to prove that the participants were homogenous in terms of their level of ability before administering the treatment. MANOVA has two specific assumptions: homogeneity of covariance matrices and homogeneity of variances. Pituch & Stevens (2015) describe other assumptions that need to be met. To test the homogeneity of covariance, the multivariate normality: data or residuals should have a multivariate normal distribution for each combination of independent and dependent variables (checked by the Shapiro-Wilk test for univariate normality and Mardia’s skewness and kurtosis for multivariate normality)

Shapiro-Wilk normality test

```
data:  Z
W = 0.24663, p-value < 2.2e-16
```

According to Pituch & Stevens (2015), the homogeneity of the variance-covariance matrices should be equal. We have seen that the variance-covariance among the individual groups is similar. However, the Shapiro-Wilk’s test verifies a significant difference (p-value < 2.2e-16) between the mean and distribution. The W value indicates how well the distribution quantiles fit the standard normal quantiles measured between 0 and 1. The closer to 1 the more perfect a fit to standard normality. The result of W = 0.19601 also confirms the data is not normally distributed. The Shapiro-Wilk test indicates that there is no normal distribution of all independent and dependent variables and therefore, the null hypothesis is rejected.

Although the third assumptions for the MANOVA statistical method is normality some researchers have argued for a loosening of this assumptions (Finch, 2005) because the reality of data is messy, noisy, and not always perfectly formed. Cohen (1968) suggests that because parametric analyses such as t-tests, analysis of variance (MANOVA), and analysis of covariance (MANCOVA) explore relationships among variables, quantitative studies would therefore produce correlational evidence. However, when describing research using this analysis technique, participants can appear to be a monolith of a single category if independence between variables is not assured. In this study, many categories form between players, between player and toy performance, and among predetermined thresholds between ranges, which are clustered according to the characteristics of the group or data (Curtis, 2016). Furthermore, before we begin the software analysis, the correlation and covariance between the variables of the TSG will set a benchmark to compare to machine learning. According to the Klopper Research Group (2022), 'using multivariable regression, one or more independent variable(s) is/are used to predict a single dependent variable. In multivariate regression, one or more independent variable(s) is/are used to predict more than one dependent variable.' Therefore, multivariable regression for the tree structure software is more appropriate for determining the relationship between the independent and dependent variables. Although two dependent variables are being examined (the players' performance value and the toy performance value), the players' performance variable shifts from dependent to independent in the analysis. The first analysis focuses on player performance, namely their right versus wrong answers as the dependent variable. The second analysis focuses on the toy and includes the player performance data as an independent variable to assess the toy performance as the dependent variable in predicting the next question or difficulty level. Using MANOVA is most appropriate because, in both cases, the dependent variable (continuous outcome), one categorical variable that defines the comparison groups, and the value of covariates of independent variables of unknown effect indicate the toy's influence on players and if that influence can be predetermined. As we saw in the pre-test analysis the two groups were equally skilled and any high-level change between the two kinds of software supports a fair comparison between the tree structure and machine learning models and not due to uneven distribution of participants.

The Fourth Assumption Covariance of Software Groups

There are two parts to the fourth assumptions still to be met for the MANOVA statistic to be a valid approach. They include linearity – dependent variables should be linearly related to each group of the independent variable. If there are more than two dependent variables, the pair of dependent variables should be linearly related. The second part of the assumption is that dependent variables should be continuous, whereas, in the research design used for testing the predictions, the dependent variable is ordinal. This is not critical however the covariate relationship between independent and dependent variables provides some indication of effect on each other.

In addition to correlation tests which measures the strength between variables as we saw in Pearson's correlation test and VIF correlation of independent variables between software groups. The Box's M test is used to indicate covariance or relationship of variables between the two software. It will determine homogeneity of covariance our fourth assumption.

```
Box's M-test for Homogeneity of Covariance Matrices
data:  dfMan
Chi-Sq (approx.) = 130.08, df = 15, p-value < 2.2e-16
```

It is well documented that Box's M test has little power to substantiate a relationship (O'Brien & Kaiser, 1985; Keselman et al., 1998; Pituch & Stevens, 2015). For this reason, the statistic uses a lower alpha level, such as 0.001 in this case, to assess the p-value for significance. Even with a lower estimated p-value, we see that there is a significant difference between the two kinds of software in terms of their relationship between independent variables. The change between software suggests that the ML model is more influenced by these variables than the tree-structured version, which had standard weights applied to them, lessening their impact.

Multivariant Analysis of Variance (MANOVA

Despite indications of non-normality of the data and other assumptions not meeting the criteria an analysis using MANOVA will provide a benchmark however should be read with caution. Precautions such as bootstrapping have not been applied and it should be repeated the state of the ML model is experimental. The following results compare the independent variables to the toy performance in making predictions of difficulty level.

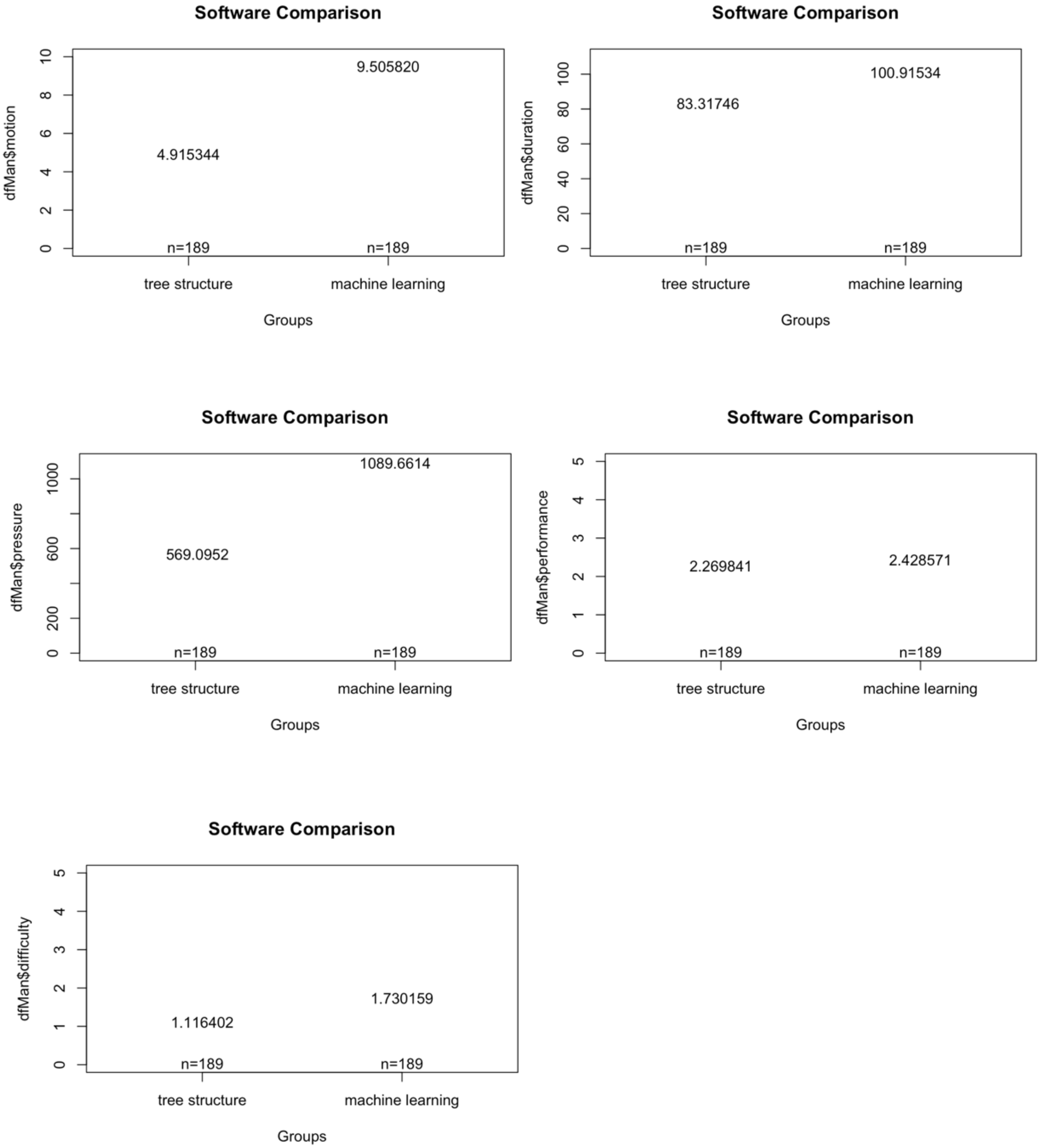


Figure 10.4 Statistical significance of each variable on the two software versions using MANOVA

MANOVA analysis – 14 participants from each group

Statistical Summary

```
               Df     Pillai        approx. F        num Df        den Df          Pr(>F)
Targetmodel    1      0.20948         19.715            5          372          < 2.2e-16 ***
Residuals      376
---
Significance codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The data, therefore, required a non-parametric method that could also handle the multivariate aspect of the data. The Kruskal -Wallis test is well suited for non-parametric data; however, additional requirements must be met to be confident in the results.

Earlier in this chapter it was suggested some statisticians have advocated for using MANOVA with non-parametric data, even though some assumptions may be violated (Finch, 2005; Doblin et al., 2020). Both Finch and Doblin et al., as well as Konietschke (2015), demonstrate the use of bootstrapping MANOVA to accommodate non-parametric data as well as ordinal dependent variables. Within this research's scope, the MANOVA model results are shown without the inclusion of bootstrapping. In order to examine the data using a more widely adopted method, the Kruskal-Wallis statistic, an equivalent model for multivariate, non-parametric data was found to be a better fit.

A Shift Toward the Kruskal -Wallis Model

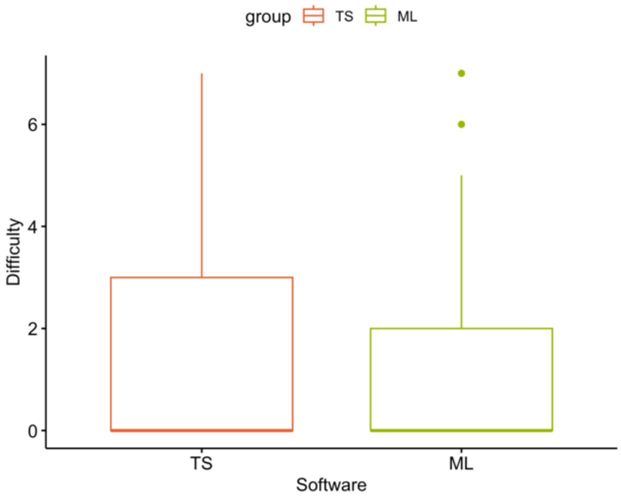
> *"Essentially, all models are wrong, but some models are useful." - George Box*

The Kruskal-Wallis method of analysis answers the question, "Are the independent variables significantly different among the three predictions?" If they are significant, it would suggest that this variable has an influence on the prediction. Still, if not, a closer look is needed at how consistently this applied across all three predictions. Conversely, if the null hypothesis is not rejected, the variable influence on the prediction is consistent and more likely to be an influential variable on all predictions, albeit weak. The method also addresses RQ2: are these the right features to determine the predictions? This is a more nuanced question than at first glance. The influence of difficulty on the prediction of 'easy' is a more cogent argument if the current difficulty is high and the previous performance is low (depending on the correlation and to what degree of influence is found in the details).

The following results consider the software comparison first, followed by a thorough examination of the prediction variable of ground-true. Stated earlier in chapter 8, the machine learning model is trained on modified tree structured ground-true results. The data is separated by software treatment using 14 of the 16 participants to achieve parity in results. Assumption tests are also covered in addition to the tests already discussed thus far.

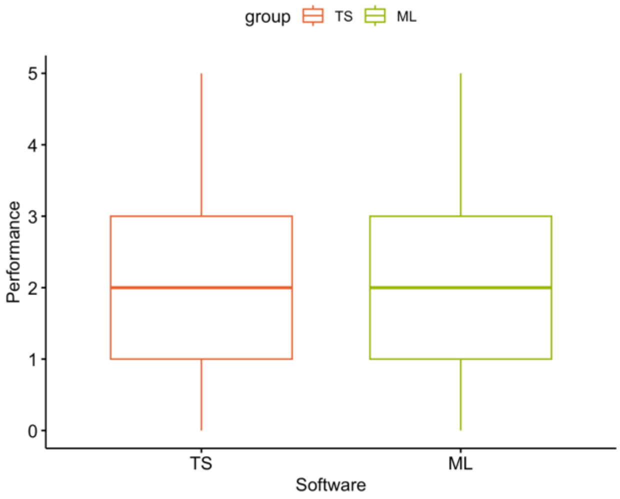
Grouped by TS and ML: a Comparison.

To understand the data more clearly the following graphs, provide results from an analysis of the five feature variables on the prediction between TS and ML. The box plots show the range of values on the top and bottom of the box with the mean line indicated between the range or set to one of the outer edges of the box.



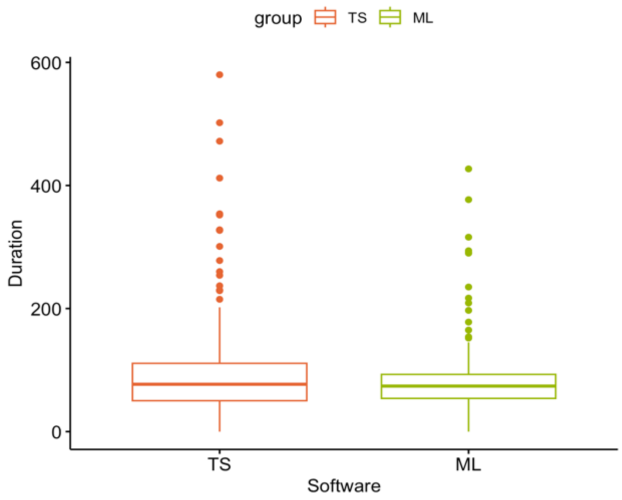
```
Targetmodel   count    Mean     sd.    median   IQR
 <ord>      <int>   <dbl>   <dbl>   <dbl>   <dbl>
1 ML        190      1.72     2.20     0          3
2 TS        188      1.12     1.66     0          2
```

Figure 10.5.a Difficulty level comparison of software



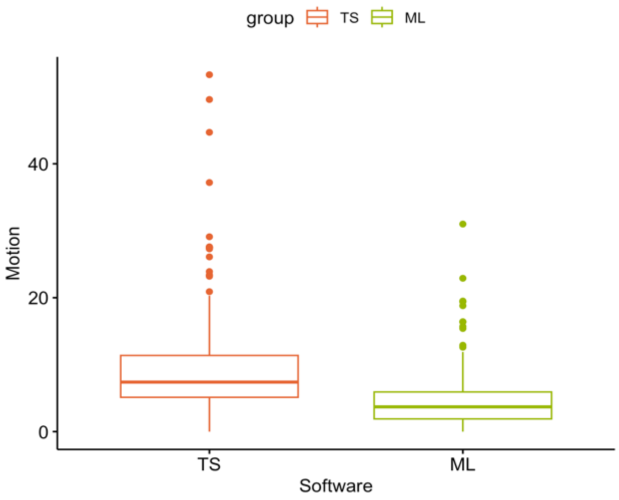
```
Targetmodel   count    Mean     sd.    median   IQR
<ord>        <int>   <dbl>   <dbl>   <dbl>   <dbl>
1 ML        190      2.42     1.43     2          2
2 TS        188      2.28     1.39     2          2
```

Figure 10.5.b Performance level comparison of software



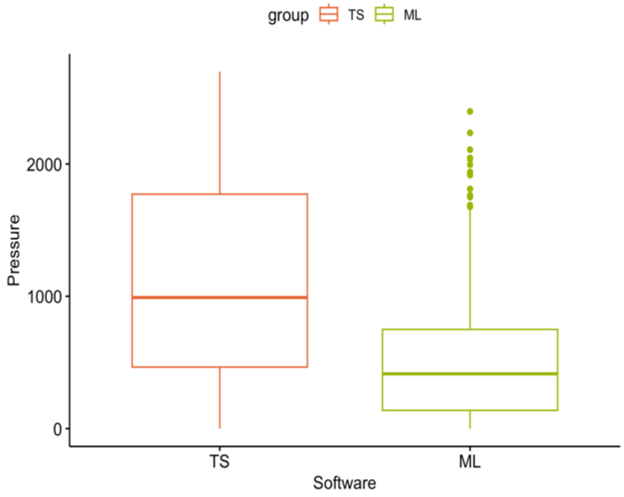
Targetmodel	count	Mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 ML	190	100.	86.4	77	60.8
2 TS	188	83.8	59.2	74	39

Figure 10.5.c Duration of time comparison of software



Targetmodel	count	Mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 ML	190	9.48	7.99	7.4	6.25
2 TS	188	4.91	4.53	3.7	4.03

Figure 10.5.d Motion amount comparison of software



Targetmodel	count	Mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 ML	190	1095	729	991	1307
2 TS	188	561	558	414	612.

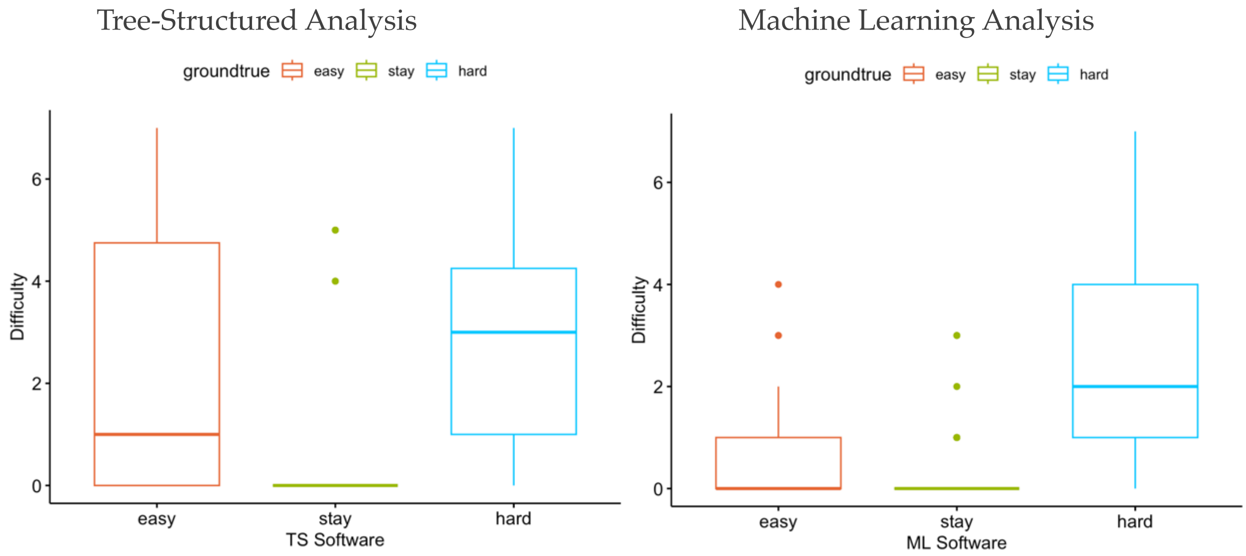
Figure 10.5.e Pressure level comparison of software

Figure 10.5 Comparison between software and each features impact on prediction

Table 10.2 Kruskal-Wallis rank sum test between software

Between Software	variable	chi-squared	df	p-value
	pressure	58.827	1	1.721e-14
	difficulty	4.3012	1	0.03809
	motion	65.872	1	4.812e-16
	duration	1.8781	1	**0.1706**
	performance	0.72463	**1**	**0.3946**

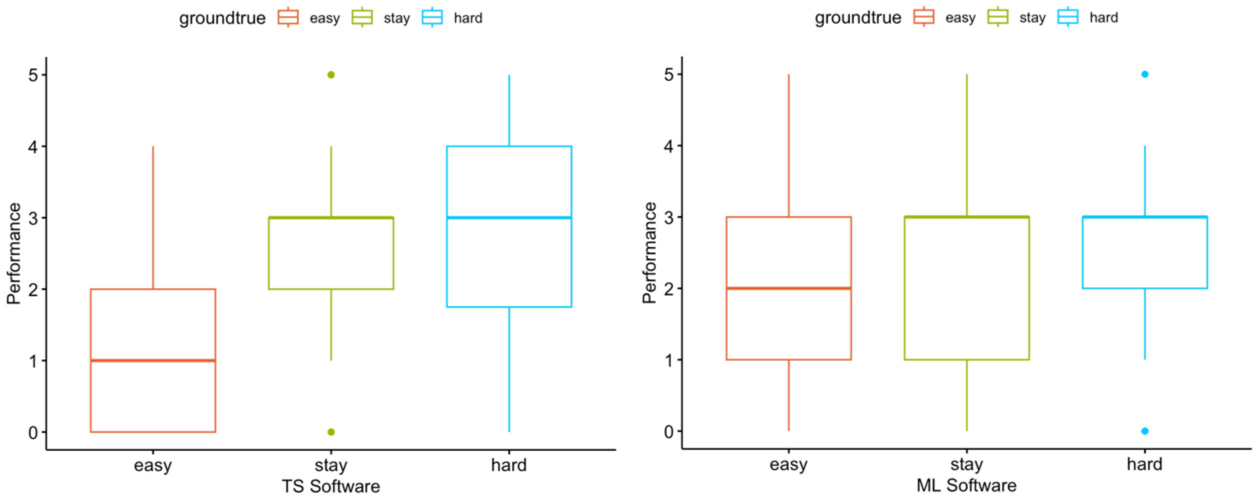
We see a significant difference in the three variables: pressure, difficulty, and motion indicating a difference between the two kinds of software. The ground true values are taken in their entirety and not separated between subgroups; however, the mean for pressure, difficulty, and motion are all larger in the TS group. This is also true of the interquartile range of the same variables, which is the middle 50% of the distribution – indicating machine learning is 47%, 66%, and 64.5% of the tree structure software predictions. Pressure and motion have similar results in the MANOVA analysis, as well as difficulty, although there is more of a difference, or what Science Direct has termed 'trending toward' greater significance. There is no difference between the duration, which is expected despite the difficulty level for the ML group to be slightly higher. Most notable is the failure to reject the performance values that contradict the MANOVA finding. There we found a significant difference, which shows that the ML model outperformed the TS group in the MANOVA but not using Kruskal-Wallis. A more focused analysis of the prediction variable follows to clarify the interpretation of this contradiction.



```
Targetmodel  count  mean   sd.   median  IQR
 <ord>       <int>  <dbl>  <dbl> <dbl>  <dbl>
1 easy        26     2.27  2.60    1     4.75
2 stay        79     0.342 1.21    0      0
3 hard        84     2.87  2.08    3     3.25
```

```
Targetmodel  count  mean   sd.   median  IQR
 <ord>       <int>  <dbl>  <dbl> <dbl>  <dbl>
1 easy        46    0.543  1.00    0      1
2 stay        77    0.273  0.700   0      0
3 hard        65    2.54   1.90    2      3
```

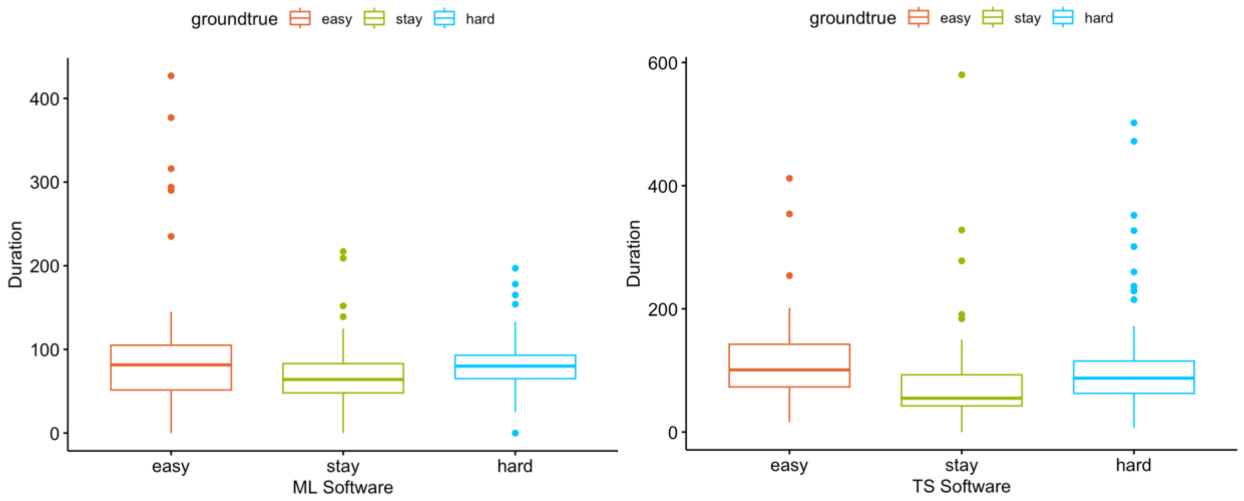
Figure 10.6.a Difficulty Influence on Ground Truth



Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	26	1.42	1.36	1	2
2 stay	79	2.58	1.37	3	1
3 hard	84	2.60	1.37	3	2.25

Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	46	2.07	1.54	2	2
2 stay	77	2.30	1.47	3	2
3 hard	65	2.42	1.17	3	1

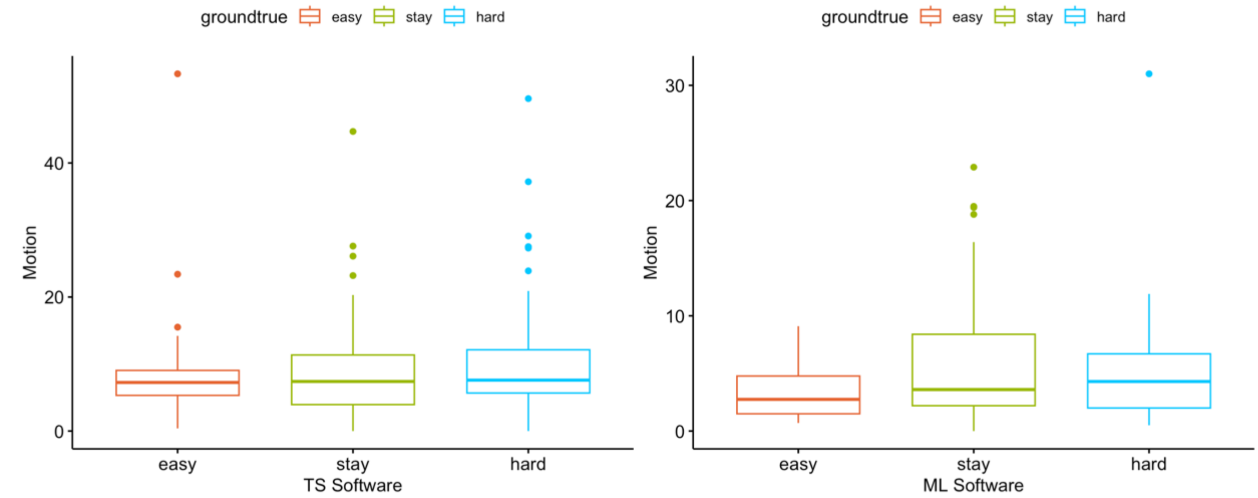
Figure 10.6.b Performance Influence on Ground Truth



Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	26	129	90.7	101	69.2
2 stay	79	79.0	78.8	55	50.5
3 hard	84	111	88.1	87.5	52.5

Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	46	108	98.7	81.5	53.5
2 stay	77	70.3	36.4	64	35
3 hard	65	82.7	34.3	80	28

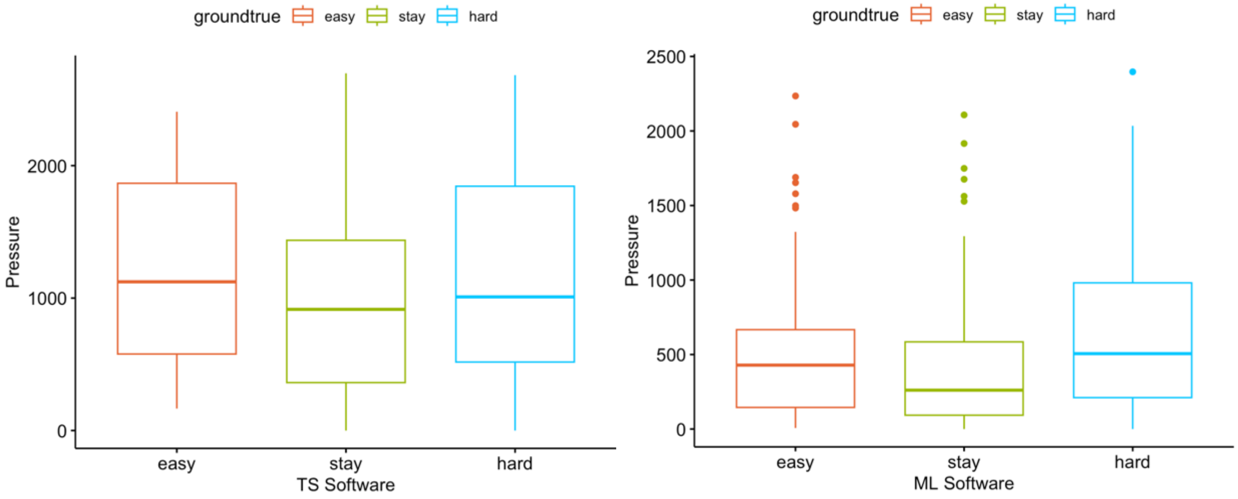
Figure 10.6.c Duration Influence on Ground Truth



Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	26	9.53	10.2	7.25	3.72
2 stay	79	8.86	6.99	7.4	7.4
3 hard	84	10.1	8.21	7.6	6.45

Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	46	3.28	2.08	2.75	3.27
2 stay	77	5.61	5.36	3.6	6.2
3 hard	65	5.25	4.48	4.3	4.7

Figure 10.6.d Motion Influence on Ground Truth



Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	26	1198	706	1123	1289
2 stay	79	994	727	915	1074.
3 hard	84	1146	732	1009	1327.

Targetmodel	count	mean	sd.	median	IQR
<ord>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1 easy	46	587	574	429	522.
2 stay	77	447	489	261	492
3 hard	65	678	603	506	770

Figure 10.6.e Pressure Influence on Ground Truth

Figure 10.6 Comparative analysis of easy, stay, and hard (Target model) predictions of each software, count = no. of observations, mean = mean value, sd = standard deviation, median = median value, IQR = interquartile range.

Insights from Both Statistical Models

Comparing assumptions of Kruskal-Wallis (KW) to MANOVA it is easy to see how the data is better suited for this statistic. The following assumption description is taken from the Machine Learning Statistic text (Dangeti, 2017) and The Kruskal-Wallis Test (McClenaghan, 2023).

- Data are assumed to be non-Normal or take a skewed distribution.
- The variable of interest should have two or more independent groups.
- The data are assumed to take a similar distribution across the groups.
- The data should be randomly selected independent samples, in that the groups should have no relationship to each other.
- Each group sample should have at least 5 observations for a sufficient sample size.

From these assumptions the data from TS and ML models are calculated and validated as follows.

Table 10.3 Kruskal-Wallis rank sum test prediction of tree structure chi-squared = mean^2, df = degrees of freedom, and p-value = likelihood of observation within the null hypothesis

Tree-Structured Software predictions	variable	chi-squared	df	p-value
	pressure	2.8691	2	**0.2382**
	difficulty	81.422	2	< 2.2e-16
	motion	1.2452	2	**0.5365**
	duration	24.223	2	5.496e-06
	performance	14.19	2	0.0008294

Table 10.4 Kruskal-Wallis rank sum test prediction of machine learning chi-squared = mean^2, df = degrees of freedom, and p-value = likelihood of observation within the null hypothesis

Machine Learning Software predictions	variable	chi-squared	df	p-value
	pressure	7.0508	2	0.02944
	difficulty	87.251	2	< 2.2e-16
	motion	6.8149	2	0.03313
	duration	9.8921	2	0.007112
	performance	2.1261	2	**0.3454**

Trending Toward Significance

In Tables 10.3 and 10.4, we see that there are differences in the input variables between the two software predictions The null hypothesis states there is no difference between the independent variables and ground-true predictions. There are two ways to interpret the hypothesis. The first is that if there is a difference, then we could say that the influence of the variable is significant and, therefore, highly influential on the predictions. The variance of difference could be due to the particular difficulty level, but for now it is a difference between predictions. Kruskal-Wallis uses rank order to determine the differences between groups, indicating that a significant difference between easy, stay and hard predictions is sensitive to the sample size. The proportion of the sample is measured by the mean rank sum of each prediction and in this case is 189, 188 predictions for TS and ML software respectively. Incidentally, this meets the first assumption of the Kruskal-Wallis test of sample size must be large (+30). Two other assumptions which are also met in the analysis method are an ordinally scaled characteristic (easy = -1, stay = 0 and hard = 1) and the non-parametric nature of the data.

The second interpretation is that no matter if the question is easier, the same or harder, the prediction is correct, subsequently if there are no significant differences between the predictions, we could say that the toy is responding to the level of the player. In this scenario, the machine learning model appears to outperform the tree structured model because the child's performance variable is predicted similarly across all three levels. Examining the differences between the predictions to provide insight into a clearer interpretation is useful.

The Kruskal-Wallis test confirmed that the groups' distributions were dissimilar from one another and the aggregate dataset. The following Dunn's test will indicate which prediction groups were dissimilar from one another, and which were not. The top number for each comparison is Dunn's pairwise z test statistic. It represents the expected average difference between the two predictions. The lower number is the raw p-value associated with the test. Because difficulty and duration have consistently low p-values in both software, the focus of the Dunn's test will be on those variables that have changed between the two. The test allows for different algorithmic changes to be applied to the p-value; however, because there are only three comparisons being made, no adjustments were added.

Comparison of machine learning pressure and motion by ground-true

```
                     (No adjustment)
 Col Mean |                                  Col Mean |
Row Mean  |   easy      hard                Row Mean |   easy      hard
   hard   | -0.833300                          hard  | -2.555844
          |   0.2023                                 |  0.0053*
   stay    |  1.502016      2.614933                     stay    |  -1.970537    0.743529
           |    0.0665        0.0045*                            |    0.0244*    0.2286
          Pressure                                              Motion
```

Comparison of tree structure performance by ground-true

```
                    (No adjustment)
 Col Mean |
Row Mean  |      easy        hard
-------------------------------
   hard   |  -3.577060
          |     0.0002*
   stay   |  -3.481579      0.099571
          |     0.0002*       0.4603
          Performance
```

According to the author of the dunn.test package for python Alexis Dimno (2017) “The null hypothesis for each pairwise comparison is that the probability of observing a randomly selected value from the first group that is larger than a randomly selected value from the second group equals one half.” The test uses the cumulative distribution function (CDF) of each prediction group, assuming they are equal but have different mean values; the larger group Dimno (2017) is referring to is the ordinal scaled value of the predictions.

In the above Dunn’s test result, we see that the comparison between harder and stay relative to easier have p-values above 0.05 and therefore we fail to reject the null hypothesis, meaning they have the same predictive result and only their mean value changes. We would expect the difference between harder and easier predictions to be similar because they often happen between players who achieve a high level followed by a player who starts from the easiest level, or when a player has reached a threshold of skill that is harder and performs very poorly over many following questions. The relationship between easier and stay can also be seen in the data as players reach that threshold that becomes difficult to perform well enough for the next level but not poorly enough to drop a level. Harder relative to stay comparison is below the threshold, indicating that across all participants in the machine learning group, the distribution of these two predictions is not equal or less than each other. Given a new observation using the same variables to predict the placement within the distribution of these two, we would expect them to be equal. In the context of the game, a player who achieves a difficulty level should be able to maintain this level of skill. There is an important distinction to be made between the performance of the toy and its ability to predict the next best question however, as we have seen, this does not ensure that the student will perform any better.

Conclusion

The difference is in the method we trust; the MANOVA analysis is accepted to be a stronger test than the Kruskal-Wallis test however, the data does not meet the requirements, and therefore, MANOVA is not a good fit. Finch (2005) suggests that non-parametric data can be used with MANOVA with additional assumptions. In this case, these additional assumptions such as size parody and amount of skewness, are met however, the potential for type I error is looming, and without further examination, significance statements cannot be made. Therefore, we must rely on the results of Kruskal-Wallis which determined there is no significant difference between TS which did not predict the next question as well as ML on the performance of the participants.

Chapter 11 - Conclusion

This study concerns the use of intelligent educational toys that modify the game to meet the specific academic needs of individual children with autistic spectrum disorder. Integrated sensors collect data for a neural network that predicts students' performance and cognitive states, adjusts the difficulty, and addresses other cognitive difficulties by changing games with varying levels. These changes allow the toy to engage children more and easily transition between games. Evidence in the performance analysis illustrates greater variation from sequence to sequence in the tree-structured version of the software compared with the machine learning version, which fluctuated less. The context of play provides a space for creating narratives to strengthen the memory and recall of information while fluctuating the experience to promote active learning. Using an information literacy framework, defined by the principles of the constructivist learning theory, knowledge is constructed rather than innate or passively absorbed, learning is an active process, all knowledge is socially constructed but also personal, and learning exists in the mind. It is this last point that is, in part, the motivation for the study. In a way, the toy is a system that captures and responds to patterns formed in the mind to anticipate both their strengths and weaknesses.

RQ 1 – Are there patterns in learning discernible from interactions with the toy?

The characteristics of these patterns vary from student to student. We see them in the features collected by the toy, although how they can be optimized for the prediction model can be further refined. Patterns do exist; however, in the physical interactions broadly defined here, the interrelationship between the features is not as influential as expected. Yet a strong correlation between features does not necessarily mean improvements in toy predictions and, more importantly, child performance. Each feature’s discrete patterns are worthy of their own study. This model was set up only to use the features as related influencers on the prediction. The weak correlations between the independent variables met an assumption of the Kruskal–Wallis statistic and provided a level of confidence in the choice of model. More data is needed to develop an ML model for pattern recognition across every permutation between features. However, as this model demonstrates, a moderate rate of predictability is possible with a moderately sized data set.

RQ 2 – Are these the right features to make successful predictions?

What the studies suggest is there are significant differences in the features selected and the prediction of the machine-learning toy. This version of the toy also outperforms more traditional forms of software development in terms of consistent predictions across the three possible choices. They do not confirm that these predictions necessarily improve a child’s performance. More study needs to be done on the clarity of the data and how that influences the statistical outcome of the toy's performance on a player's performance. As the quality of data increases and model modifications are made, a reciprocal effect could develop, which may improve child performance as a result. Some statistical models, like MANOVA, indicated a significant difference between the two software whereas others did not. The same quality changes to the data could make these more powerful statistics closely align with the data and conform to the assumptions of such models.

RQ 3 – How does TS's pre-prediction approach compare with ML in prediction of question difficulty?

The question of toy performance is not an easy one to answer within the context of difficulty levels. We see there is a difference between the two software and how they respond to the player but what is a 'correct' prediction. The most direct answer is if a player is given a difficulty level who then goes on to improve their score, we could say this is a correct prediction in which case ML slightly outperforms TS. The probability of the model to select a difficulty level that is more likely to increase a player's performance, singles out performance as the only measure. If we consider the notion of 'challenge' as a measure of correctness, the answer becomes more nuanced. Measures of duration and physical indicators like pressure and motion are markers of engagement or 'flow' which uses challenge as a factor in achieving the flow state. We want to challenge students even if they get the answer wrong however, we ultimately want to see them achieve and perform well which may come from playing more than one time.

RQ 4 – How well can ML predict game-level difficulty?

Educational toys that can identify the optimal time for learning and anticipate when support is needed will benefit all students experiencing learning difficulties and help them improve. Leveraging machine learning to enhance education, mainly to accelerate learning for children with autism, is a complex task, and this study initiates these efforts. It could also be interpreted as any entry point into many related studies that seem daunting. However, what is the alternative? Should we continue to make modest attempts at improving education at a rate that is a fraction of technologies such as artificial intelligence? Technology is expected to disrupt many facets of life, and what better time to take advantage of this disruption to help kids who have not had as many advantages afforded to them?

RQ 5 – Can ML improve performance scores over tree structured programming?

At this time no definitive answer can be given. The Kruskal-Wallis statistic model does not show a significant difference in player performance but with a number of qualified statements MANOVA is unclear. Given more testing which brings more data and modifications to the machine learning model a significant difference may be possible in future studies. An improvement to the model along with more data could bring about a more significant influence on player performance. This study is tentative step in that direction and encouraging for further study.

Chapter 12 - Future Studies

This study was a significant undertaking with some detailed analysis of the top-level research questions. However, the potential for the toy can take many directions. Some include modifying the existing toy to collect additional data to improve gameplay and enhance learning or improving the data for a similar study. Still, other future studies could involve new questions about learning with autism or study other features known to contribute to learning, which were beyond the scope of this project. Emotion and engagement are two areas that are hard to capture and process in real-time for gameplay. The system infrastructure is robust enough to include additional sensors or data collection devices that make it possible to append them to the current toy.

Possible Changes to the Study

An early design concern was the possibility that players of the math game could follow the lights sequence and not consider the significance of numbers. This concern was waylaid in watching children count out the buttons' pips to remember the answer and practice their math skills. Players could be encouraged further by presenting the first two numbers to the operator and letting the players provide the answer. Indeed, this change would further the divide between difficulty levels and encourage greater cognitive engagement.

Increasing the number of predictions could result in more granular conditions determining the outcome. Instead of the three difficulty level choices – harder, easier, or the same – five possible choices would like very hard, harder, the same, easier, very easy - improve the game experience for better players and could respond more quickly to struggling players. The more available outcomes add to the challenge of keeping the game easy to play but could improve engagement, particularly with repeat players. Another gameplay change would allow for corrections. Occasionally, a player would press buttons accidentally or out of order, causing the toy to respond not in line with the player's understanding but correctly registering an incorrect answer. A 'submit' button or other action could solve this and improve the game experience. Children of this age seek approval and praise from their teachers, which is often observed during the sessions. Getting the 'right' answer is a motivating factor that could be improved while also enhancing the training of the machine learning model.

Tracking Motor Skills that Disrupt Learning

Although the x, y, and z position data is collected, the toy's orientation is underutilized here. It could be leveraged to understand the more considerable impact of stereotypical motor movements (SMM) on learning. The SMMs are not a consistent characteristic of all ASD children like many of the ASD indicators, but head bobbing, hand and finger actions, or mouthing are typical when they are present (Sadouk et al., 2018). In their article from Computational Intelligence and Neuroscience, Sadouk et al. explain their approach to capturing this data, "The SMM dataset consists of time-series data that are composed of multiple channels D, i.e., x, y, and z coordinate measurements recorded from multiple sensors/devices. The first step is to convert these D-channel raw data into multiple fixed length signals in both time and frequency-domain, denoted as frames" (2018).

Future Emotional Tie to Learning

"We have been building technological affordances that serve as emotional and inspirational mentors and that foster the creative and idiosyncratic connections to learning that help community members progress through these planes" (Picard et al., 2004).

In an interview panel with Noam Chomsky (linguistics) and Gary Marcus (psychology), they took a critical look at the future direction of machine learning. In part, they call for more attention to neuro-symbolic AI, which focuses on understanding the world. Most recently, we have seen ChatGPT 4 be over-hyped; however impressive its ability to write text is, incorrect or false statements are often made. More importantly, Chomsky and Marcus point out that more training and money to run servers to process writing will only marginally improve the chatbot. What it lacks is the understanding of the physical world in a meaningful way (Marcus, 2023). Senior Researcher of AI at Google Research, Vinodkumar Prabhakaran, explains that there is a need to incorporate types of data that represent the characteristics of an idea without long semantic descriptions. For example, Prabhakaran describes using decibel levels of police radio conversations and transcriptions for data used in machine learning to understand how and what was being said more fully. Many examples of data sets used in current AI tools are texts found on the internet or easily accessible databases because they are digitized information. This study, however small, attempts to add physical data points to influence the algorithm. One could add many more features to the data to broaden the understanding of the model. To restate Rajput et al. (2017), prediction could be improved by the quality of the data by 5-20%, and such refinements could be achieved through collection and feature engineering. While these features significantly influence the predictions, other biometric data could also contribute to the connecting patterns found here. We could add even more complex feature-tracking techniques to capture engagement or emotions observed during the study and should be included as factors in education and a positive experience.

In future examples, I plan to investigate what levels of engagement influence students in the subject. Other perspectives, such as cultural differences and biometric data, may also cross the influence threshold. “If you just accumulate statistical evidence and don’t understand the dynamics of things, you have a problem” (Marcus quote by Ray, 2022). Reductionism of the models to operate more efficiently and for the broader audience all too often favors output performance or, more simply, right versus wrong as the most crucial measure. Education is a complex process, but studies of this kind are the first step toward unraveling the relationships between its features and unwinding individual characteristics to rethread a model and make better predictions.

Three Levels of Engagement

Early discussions concerning engagement approached the topic as a single state of interest that waxes and wanes depending on the stimulation levels of the game. After observing children playing the game, another phenomenon appears to be happening, which is more complex and nuanced than a stimulation-response relationship. Children were offered the option to play the game at different times during the day apart from other activities in school. Their initial interest may have been motivated by several factors; however, their time with the game evolved as they worked through the rules and interactions required to play. Following the sequence of lights is easy enough for all the players, and no one has reported being unable to understand the game's procedures. After some time, children who answered correctly more often remained engaged, as evidenced by their playing longer than those who did not get many correct answers. The more successful player's engagement levels dropped; however, after they understood the mechanics and succeeded in the game challenges, these factors made it less enjoyable – which is expected according to flow theory (Csikszentmihalyi, 2014). Except for the level of difficulty change, the interaction is the same. In a typical video game, new and varied sequences of button presses are required to accomplish a task, which is not the case in this study. Task variation and difficulty level could make the game more engaging and keep the player's attention longer. A second phase of engagement came after a period of success due to needing a clearer understanding of when or what is required to win. Completing level eight with a performance level of 4 or 5 out of 5 correct answers will win the game, and only 3 of the 32 players accomplished this goal. The last engagement phase was observed in the first two children who played the machine-learning version of the toy and also played the tree structure version. Although excited to play, their interest in the game dropped much sooner than other players who were new to the game. Even though they volunteered to play the game again, they quit sooner than players who could have performed better. They did not realize the game would respond differently, or they genuinely became disinterested sooner than the first time they played. These were just the observed phases of engagement. Still, variations in the game, game interactions, and general familiarity contribute to a loss in engagement, and the difficulty level is only one factor investigated in this study. Future studies should control for other influences on engagement or build them in as features for the machine learning model.

In addition to constructivist theory, the protégé effect, having students teach others or teach the toy instead of getting all the input information from the game, also has potential. Players seemed to pick up the game quickly and were less intimidated meeting with me, a stranger offering to play, rather than their usual caregivers. One of the tenets of the Turing test proposed that if a human interacts with a computer, and the human believes the computer is a person, then the computer has achieved human intelligence (Turing 1950). Placed in the context of a teacher, the children may find it helpful to learn the skill and the subject, reaffirming their understanding of the game if the toy is their pupil. If the toy learns as the child learns, it is easy to imagine a lifelong partnership.

Although many participants received one-on-one attention from their teachers, the toy could also be studied as a socialization tool if it was a multiplayer game. Having students play against each other and letting the toy interpret everyone’s skill level to control the sequence at the individual level allows everyone to compete. For example, one child could enter the sequence quickly on their toy while another receives the game at a pace that is targeted toward a player at a slower pace. Duration would be used to calculate frustration scores, but a separate performance score could introduce competition. Autistic students may prefer this in remote settings; however, what Webb et al. (2001) call deep learning (conceptual and practical understanding) reverses the Turing concept, where a human is perceived as a machine, placing distance between players, easing them into social interaction. Engaging with the toy was not intimidating to children, and following its directions came naturally. Another potential for the toy is studying its ability to teach using mimicry, a strategy some autistic children do not use. The complexity of this audience adds to the complexity of developing a workable educational technology system. What is most motivating is the number of possibilities the toy offers to research. As with many systems, improved technologies, more user input, and more data will provide insights into autistic learners and will likely occupy much more of my research agenda in the future.

Bibliography

Aazam, M., Zeadally, S., & Harras, K. A. (2018). Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities. Future Generation Computer Systems, 87, 278-289.

Ackermann, E. (2001). Piaget's constructivism, Papert's constructionism: What's the difference. Future of learning group publication, 5(3), 438.

Ackermann, E. K. (2004). Constructing knowledge and transforming the world. A learning zone of one's own: Sharing representations and flow in collaborative learning environments, 1, 15-37.

Adnan, M., Habib, A., Ashraf, J., Mussadiq, S., Raza, A. A., Abid, M., ... & Khan, S. U. (2021). Predicting at-risk students at different percentages of course length for early intervention using machine learning models. Ieee Access, 9, 7519-7539.

Adebisi, R. O., Liman, N. A., & Longpoe, P. K. (2015). Using Assistive Technology in Teaching Children with Learning Disabilities in the 21st Century. Journal of Education and Practice, 6(24), 14-20.

Ahadi, A., Lister, R., Haapala, H., & Vihavainen, A. (2015, August). Exploring machine learning methods to automatically identify students in need of assistance. In Proceedings of the eleventh annual international conference on international computing education research (pp. 121-130).

Alberto, P., Troutman, A. C., & Axe, J. B. (2006). Applied behavior analysis for teachers (pp. 1-474). Upper Saddle River, NJ: Pearson Merrill Prentice Hall.

Alghofaili, Y. (2020). Interpretable K-Means: clusters feature importances. Towards Data Science.

American Psychiatric Association et al. Diagnostic and statistical manual of mental disorders (DSM-5 R). American Psychiatric Pub, 2013.

Anderson, L. (2001). W. & Krathwohl, D. A Taxonomy for Learning, Teaching and Assessing: a Revision of Bloom's Taxonomy of Educational Objectives". New York: Longman.

Anderson, L.W. (Ed.), Krathwohl, D.R. (Ed.), Airasian, P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R., Raths, J., & Wittrock, M.C. (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom’s Taxonomy of Educational Objectives (Complete edition). New York: Longman

Arroyo, I., Cooper, D. G., Burleson, W., Woolf, B. P., Muldner, K., & Christopherson, R. (2009). Emotion sensors go to school. In Artificial intelligence in education (pp. 17-24). Ios Press.

Ausubel, D. P., Stager, M., & Gaite, A. J. H. (1968). Retroactive facilitation in meaningful verbal learning. Journal of Educational Psychology, 59(4), 250.

Bara, F., Gentaz, E., Colé, P., & Sprenger-Charolles, L. (2004). The visuo-haptic and haptic exploration of letters increases the kindergarten-children’s understanding of the alphabetic principle. Cognitive development, 19(3), 433-449.

Bada, S. O., & Olusegun, S. (2015). Constructivism learning theory: A paradigm for teaching and learning. Journal of Research & Method in Education, 5(6), 66-70.

Baumeister, M., Ditzhaus, M., & Pauly, M. (2024). Quantile-based MANOVA: A new tool for inferring multivariate data in factorial designs. Journal of Multivariate Analysis, 199, 105246.

Bell, B. A. (2014). Encyclopedia of research design.

Becker, K. (2007). Battle of the Titans: Mario vs. MathBlaster. In EdMedia+ Innovate Learning (pp. 2707-2716). Association for the Advancement of Computing in Education (AACE).

Bjørgen, K. (2016). Physical activity in light of affordances in outdoor environments: Qualitative observation studies of 3–5 years olds in kindergarten. Springerplus, 5(1), 1-11.

Blackler, A. (2008). Intuitive interaction with complex artefacts: empirically based research. VDM Verlag Publishing Group.

Bloom, B. S. (1987). A response to Slavin’s mastery learning reconsidered. Review of Educational Research, 57(4), 507–508. https://doi.org/10.3102/00346543057004507

Bloom, B. (1956). Bloom’s taxonomy.

Blöte, A. W., Van der Burg, E., & Klein, A. S. (2001). Students' flexibility in solving two-digit addition and subtraction problems: Instruction effects. Journal of Educational Psychology, 93(3), 627.

Bolding, K., & Rudy, J. W. (2006). Place learning in the Morris water task: Making the memory stick. Learning & memory, 13(3), 278-286.

Borghi, A. M., Flumini, A., Natraj, N., & Wheaton, L. A. (2012). One hand, two objects: Emergence of affordance in contexts. Brain and Cognition, 80(1), 64-73.

Bormann, J. (2014). Affordances of flipped learning and its effects on student engagement and achievement.

Bosseler, A., & Massaro, D. W. (2003). Development and evaluation of a computer-animated tutor for vocabulary and language learning in children with autism. Journal of autism and developmental disorders, 33(6), 653-672.

Boyd, B. A., Conroy, M. A., Mancil, G. R., Nakao, T., & Alter, P. J. (2007). Effects of circumscribed interests on the social behaviors of children with autism spectrum disorders. Journal of autism and developmental disorders, 37, 1550-1561.

Breiman, L.: Classification and Regression Trees. Routledge, New York (2017)

Brown, E., & Cairns, P. (2004). A grounded investigation of game immersion. In Proceedings of the Conference on Human Factors in Computing Systems (pp. 1297–1300). New York: ACM.

Brophy, J., & Evertson, C. M. (1978). Context variables in teaching. Educational Psychologist, 12(3), 310-316.

Brownlee, J. (2018). Statistical methods for machine learning: Discover how to transform data into knowledge with Python. Machine Learning Mastery.

Bull, R., & Scerif, G. (2001). Executive functioning as a predictor of children's mathematics ability. Inhibition, switching, and working memory. Developmental Neuropsychology, 19, 273-293.

Bulunuz, M. (2013). Teaching science through play in kindergarten: Does integrated play and science instruction build understanding? European Early Childhood Education Research Journal, 21(2), 226-249.

Bulaghi, Z. A., Navin, A. H. Z., Hosseinzadeh, M., & Rezaee, A. (2020). SENET: A novel architecture for IoT-based body sensor networks. *Informatics in Medicine Unlocked*, *20*, 100365.

Butera, G., & Haywood, H. C. (1995). Cognitive education of young children with autism: an application of Bright Start. Learning and cognition in autism, 269-292.

Burnett, M. M., & Myers, B. A. (2014). Future of end-user software engineering: beyond the silos. In Future of Software Engineering Proceedings (pp. 201-211).

Calik, N. C., & Kargin, T. (2010). Effectiveness of the Touch Math Technique in Teaching Addition Skills to Students with Intellectual Disabilities. International journal of special education, 25(1), 195-204.

Calvillo-Gámez, E. H., Cairns, P., & Cox, A. L. (2015). Assessing the core elements of the gaming experience. Game user experience evaluation, 37-62.

Cai, S, Zhu, G., Wu, Y., Liu, E., Hu, X. (2018). A case study of gesture-based games in enhancing the fine motor skills and recognition of children with autism. Interactive Learning Environments, pg. 1-14. 10.1080/10494820.2018.1437048.

Cannon, H. M., & Feinstein, A. H. (2005). Bloom beyond Bloom: Using the revised taxonomy to develop experiential learning strategies. In Developments in Business Simulation and Experiential Learning: Proceedings of the Annual ABSEL conference (Vol. 32).

Cen, H., Chen, Y., Kulkarni, T., & Schneider, J. (2018). Personalized learning with machine learning: A review. Educational Psychology Review, 30(4), 533-558.

Charlton, B., Williams, R. L., & McLaughlin, T. F. (2005). Educational Games: A Technique to accelerate the acquisition of reading skills of children with learning disabilities. International Journal of Special Education, 20(2), 66-72.

Chamak, B., Bonniau, B., Jaunay, E., & Cohen, D. (2008). What can we learn about autism from autistic persons?. Psychotherapy and psychosomatics, 77(5), 271-279.

Chapman, P., Selvarajah, S., & Webster, J. (1999, January). Engagement in multimedia training systems. In Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences. 1999. HICSS-32. Abstracts and CD-ROM of Full Papers (pp. 9-pp). IEEE.

Chauhan, R. S., Taneja, K., Khanduja, R., Kamra, V., & Rattan, R. (2022). Evolutionary computation with intelligent systems. Evol. Comput. with Intell. Syst.

Church, B. A., Rice, C. L., Dovgopoly, A., Lopata, C. J., Thomeer, M. L., Nelson, A., & Mercado, E. (2015). Learning, plasticity, and atypical generalization in children with autism. Psychonomic bulletin & review, 22(5), 1342-1348.

Costa, E. B., Fonseca, B., Santana, M. A., de Araújo, F. F., & Rego, J. (2017). Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses. Computers in human behavior, 73, 247-256.

Cooper, J. O., Heron, T. E., & Heward, W. L. (2007). Applied behavior analysis.

Chauhan, R. S., Taneja, K., Khanduja, R., Kamra, V., & Rattan, R. (2022). Evolutionary computation with intelligent systems. Evol. Comput. with Intell. Syst.

Chen, S. F., & Rosenfeld, R. (1999, March). Efficient sampling and feature selection in whole sentence maximum entropy language models. In 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258) (Vol. 1, pp. 549-552). IEEE.

Christy, K. R., Minich, M., Tao, R., Riddle, K., & Kim, S. (2022). To tailor or not to tailor: An investigation of narrative tailoring for health communication. Journal of Health Communication, 27(3), 152-163.

Ciolacu, M., Tehrani, A. F., Beer, R., & Popp, H. (2017, October). Education 4.0—Fostering student's performance with machine learning methods. In 2017 IEEE 23rd International Symposium for Design and Technology in Electronic Packaging (SIITME) (pp. 438-443). IEEE.

Cohen J (1968) Multiple regression as a general data-analytic system. Psychological Bulletin. 70, 6: Part 1, 426- 443. 10.1037/h0026714

Cleveland, H. H., Jacobson, K. C., Lipinsky, J. J., & Rowe, D. C. (2000). Genetic and shared environmental contributions to the relationship between the home environment and child and adolescent achievement. Intelligence, 28, 69-86.

Costa, E. B., Fonseca, B., Santana, M. A., de Araújo, F. F., & Rego, J. (2017). Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses. Computers in human behavior, 73, 247-256.

Cooper, J. O., Heron, T. E., & Heward, W. L. (2007). Applied behavior analysis.

Csikszentmihalyi, M. (2002). Flow: The psychology of happiness: The classic work on how to achieve happiness. London, UK: Rider.

Csikszentmihalyi, M., Csikszentmihalyi, M., Abuhamdeh, S., & Nakamura, J. (2014). Flow. In Flow and the foundations of positive psychology: The collected works of Mihaly Csikszentmihalyi, 227-238.

Curtis, E. A., Comiskey, C., & Dempsey, O. (2016). Importance and use of correlational research. Nurse researcher, 23(6).

De Freitas, S. (2018). Are games effective learning tools? A review of educational games. Journal of Educational Technology & Society, 21(2), 74-84.

Dimitrov, D. M., & Rumrill Jr, P. D. (2003). Pretest-posttest designs and measurement of change. Work, 20(2), 159-165.

Dimno, Alexis (2017), Dunn's Test of Multiple Comparisons Using Rank Sums, distributed under GPL-2 license, package "dunn.test"

Dobler, D., Friedrich, S., & Pauly, M. (2020). Nonparametric MANOVA in meaningful effects. Annals of the Institute of Statistical Mathematics, 72(4), 997-1022.

Drigas, A. S. and Ioannidou, R. E., 2012. Artificial intelligence in special Education: A decade review",
International Journal of Engineering Education, 28, 6, 1366- 1372.

Edmonds, Bruce M., Syntactic Measures of Complexity (PhD dissertation, University of Manchester, 1999), available at https://objectsindevelopment.net/phd-thesis/.

Ekman, P. (1999). Facial Expressions. New York: John Wiley & Sons Ltd.

Espy, K. A., McDiarmid, M. M., Cwik, M. F., Stalets, M. M., Hamby, A., & Senn, T. E. (2004). The contribution of executive functions to emergent mathematic skills in preschool children. Developmental Neuropsychology, 26, 465-486.

Fikar, P., Güldenpfennig, F., & Ganhör, R. (2018, June). The use (fulness) of therapeutic toys: Practice-derived design lenses for toy design. In Proceedings of the 2018 Designing Interactive Systems Conference (pp. 289-300).

Filsecker, M., & Kerres, M. (2014). Engagement as a volitional construct: A framework for evidence-based research on educational games. Simulation & Gaming, 45(4-5), 450-470.
Finch, H. (2005). Comparison of the performance of nonparametric and parametric MANOVA test statistics when assumptions are violated. Methodology, 1(1), 27-38.

Fogg, B. J. (2009, April). A behavior model for persuasive design. In Proceedings of the 4th International Conference on Persuasive Technology (pp. 1-7).

Fogg, B. J., & Nass, C. (1997). How users reciprocate to computers: an experiment that demonstrates behavior change. In CHI'97 extended abstracts on Human factors in computing systems (pp. 331-332).

Frascara, J. (Ed.). (2003). Design and the social sciences: making connections (Vol. 2). CRC Press.

Garretson, H. B., Fein, D., & Waterhouse, L. (1990). Sustained attention in children with autism. Journal of autism and developmental disorders, 20(1), 101-114.

Garris, R., Ahlers, R., & Driskell, J. E. (2002). Games, motivation, and learning: A research and practice model. Simulation & Gaming, 33, 441-467. doi:10.1177/1046878102238607

Gennari, J. H. (1989) P. Langley, and D. Fisher. Models of incremental concept formation. Artificial Intelligence, (40):11–61

Gernsbacher, M. A. (2006). Toward a behavior of reciprocity. The journal of developmental processes, 1(1), 139.

Gibson JJ (1979) The theory of affordances. In: The ecological approach to visual perception. Houghton Mifflin, Hopewell, pp 127–143

Gibson, J. J. (1962). Observations on active touch. Psychological review, 69(6), 477.

Gick, M. L., & Holyoak, K. J. (1987). The cognitive basis of knowledge transfer. In Transfer of learning (pp. 9-46). Academic Press.

Ghiselli, E. E. (1960). The prediction of predictability. Educational and Psychological Measurement, 20(1), 3-8.

Gosen, J., & Washbush, J. (2004). A review of scholarship on assessing experiential learning effectiveness. Simulation & Gaming, 35(2), 270-293.

Gorton, Ian. Essential Software Architecture, Springer Berlin / Heidelberg, 2011. ProQuest Ebook Central, https://ebookcentral.proquest.com/lib/clemson/detail.action?docID=993698.

Granic, I., Lobel, A., & Engels, R. C. (2014). The benefits of playing video games. American psychologist, 69(1), 66.

Gray, L., and Lewis, L. (2021). Use of Educational Technology for Instruction in Public Schools: 2019–20 (NCES 2021017). U.S. Department of Education. Washington, DC: National Center for Education Statistics. Retrieved [date] from: https://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2021017

Hall M. A. (1999) Correlation-based feature selection for machine learning. PhD thesis, The University of Waikato.

Harrell, F. E. (2001). Regression modeling strategies: with applications to linear models, logistic regression, and survival analysis (Vol. 608). New York: Springer.

Heick, T. (2021, March 22). Learning theories: Adaptive control of thought. Teach Thought. https://www.teachthought.com/learning/theory-cognitive-architecture/

Hedges, H., & Cullen, J. (2012). Participatory learning theories: A framework for early childhood pedagogy. Early Child Development and Care, 182(7), 921-940.

Hemenover, S. H., & Bowman, N. D. (2018). Video games, emotion, and emotion regulation: Expanding the scope. Annals of the International Communication Association, 42(2), 125-143.

Hernandez, J., Paredes, P., Roseway, A., & Czerwinski, M. (2014, April). Under pressure: sensing stress of computer users. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 51-60).

Hilppö, J. (2016). Children's sense of agency: A co-participatory investigation.

Honauer, M., Moorthy, P., & Hornecker, E. (2019, October). Interactive soft toys for infants and toddlers-design recommendations for age-appropriate play. In Proceedings of the annual symposium on computer-human interaction in play (pp. 265-276).

Hodent, C. (2014). Toward a playful and usable education. Learning by playing: Video gaming in education, 69-86.

Hofmann, S. G., Sawyer, A. T., & Fang, A. (2010). The empirical status of the "new wave" of cognitive behavioral therapy. Psychiatric Clinics, 33(3), 701-710.

Holzinger, A. (2019). Introduction to machine learning & knowledge extraction (make). Machine learning and knowledge extraction, 1(1), 1-20.

Hoskin, T. (2012). Parametric and nonparametric: Demystifying the terms. In Mayo Clinic (Vol. 5, No. 1, pp. 1-5).

Hrouda-Rasmussen, S. (2021, March 18). Toward Data Science. https://towardsdatascience.com/what-is-bayesian-inference-4eda9f9e20a6

Hruby, G. G. (2001). Sociological, postmodern, and new realism perspectives in social constructionism: Implications for literacy research. Reading Research Quarterly, 36(1), 48-62.

Huang, Y., & Shen, F. (2016). Effects of cultural tailoring on persuasion in cancer communication: A meta-analysis: Cultural tailoring in cancer communication. Journal of Communication, 66(4), 694–715. doi:10.1111/jcom.12243

Hunicke, R., LeBlanc, M., & Zubek, R. (2004, July). MDA: A formal approach to game design and game research. In Proceedings of the AAAI Workshop on Challenges in Game AI (Vol. 4, No. 1, p. 1722).

Illeris, K. (2018). A comprehensive understanding of human learning. In Contemporary theories of learning (pp. 1-14). Routledge.
Indriayu, M. (2019). Effectiveness of Experiential Learning-Based Teaching Material in Mathematics. International Journal of Evaluation and Research in Education, 8(1), 57-63.

Ingersoll, B & Schreibman, L. (2006) Teaching Reciprocal Imitation Skills to Young Children with Autism Using a Naturalistic Behavioral Approach: Effects on Language, Pretend Play, and Joint Attention, Journal of Autism and Developmental Disorders, Vol. 36, No. 4

Jahadakbar, M., Araujo de Aguiar, C. H., Nikkhah Dehnavi, A., & Ghandi, M. (2023, July). Sounds of Play: Designing Augmented Toys for Children with Autism. In Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments (pp. 338-346).

Jarvis, P., Newman, S., & Swiniarski, L. (2014). On 'becoming social': The importance of collaborative free play in childhood. International Journal of Play, 3(1), 53-68.

Jarvis, P. (2012). Towards a comprehensive theory of human learning. Routledge.

Jiang, S., Pang, G., Wu, M., & Kuang, L. (2012). An improved K-nearest-neighbor algorithm for text categorization. Expert Systems with Applications, 39(1), 1503-1509.

Johnston, J. S. (2009). What does the skill of observation look like in young children?. International Journal of Science Education, 31(18), 2511-2525.

Jolliffe, T., & Baron-Cohen, S. (1999). Linguistic processing in high-functioning adults with autism or Asperger syndrome. Can local coherence be achieved? A test of central coherence theory. Cognition, 71, 149-185.

Jones, J., Lerman, D. C., & Lechago, S. (2014). Assessing stimulus control and promoting generalization via video modeling when teaching social responses to children with autism. Journal of applied behavior analysis, 47(1), 37-50.

Kamii, C., Lewis, B. A., & Kirkland, L. D. (2001). Fluency in subtraction compared with addition. The Journal of Mathematical Behavior, 20(1), 33-42.

Kaye, J. J. (2007, April). Evaluating experience-focused HCI. In CHI'07 extended abstracts on Human factors in computing systems (pp. 1661-1664).

Keay-Bright, W., & Howarth, I. (2012). Is simplicity the key to engagement for children on the autism spectrum? Personal and ubiquitous computing, 16, 129-141.

Kennedy, J. M., Gabias, P., & Heller, M. A. (1992). Space, haptics and the blind. Geoforum, 23(2), 175-189.

Keselman, H. J., Huberty, C. J., Lix, L. M., Olejnik, S., Cribbie, R. A., Donahue, B., ... & Levin, J.

R. (1998). Statistical practices of educational researchers: An analysis of their ANOVA, MANOVA, and ANCOVA analyses. Review of educational research, 68(3), 350-386.

Korkmaz, C., & Correia, A. P. (2019). A review of research on machine learning in educational technology. Educational Media International, 56(3), 250-267.

Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. Theory into practice, 41(4), 212-218.

Klopfer, E., Osterweil, S., & Salen, K. (2009). Moving learning games forward. Cambridge, MA: The Education Arcade.

Konietschke, F., Bathke, A. C., Harrar, S. W., & Pauly, M. (2015). Parametric and nonparametric bootstrap methods for general MANOVA. Journal of Multivariate Analysis, 140, 291-301.

Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. Theory into practice, 41(4), 212-218.

Kučak, D., Juričić, V., & Đambić, G. (2018). MACHINE LEARNING IN EDUCATION-A SURVEY OF CURRENT RESEARCH TRENDS. Annals of DAAAM & Proceedings, 29.

Kukiela, Daniel (2023), Neural Networks from Scratch, print on demand https://nnfs.io/

Kullback S. and R. A. Leibler. On information and sufficiency. Ann. Math. Statist., 22(1):79{86, 03 1951.

Kang, E. B. (2023). Target model tracings (GTT): On the epistemic limits of machine learning. Big data & society, 10(1), 20539517221146122.

Kurzweil R. 2012 How to create a mind. New York, NY: Viking.

Laine, T. H., & Lindberg, R. S. (2020). Designing engaging games for education: A systematic literature review on game motivators and design principles. IEEE Transactions on Learning Technologies, 13(4), 804-821.

LeCun Yann, (2022), A Path Towards Autonomous Machine Intelligence Version 0.9.2

Lenberg, P., Feldt, R., & Wallgren, L. G. (2015). Behavioral software engineering: A definition and systematic literature review. Journal of Systems and Software, 107, 15-37.

Lindgren, R., Tscholl, M., Wang, S., & Johnson, E. (2016). Enhancing learning and engagement through embodied interaction within a mixed reality simulation. *Computers & education*, *95*, 174-187.

Linstead, E., German, R., Dixon, D., Granpeesheh, D., Novack, M., & Powell, A. (2015, December). An application of neural networks to predicting mastery of learning outcomes in the treatment of autism spectrum disorder. In 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA) (pp. 414-418). IEEE.

Macintosh, G., Gentry, J. W., & Stoltman, J. J. (1993, March). A systematic approach to the development and evaluation of experiential exercises. In Developments in Business Simulation and Experiential Learning: Proceedings of the Annual ABSEL conference (Vol. 20).

Mansur, A. B. F., Yusof, N., & Basori, A. H. (2019). Personalized learning model based on deep learning algorithm for student behaviour analytic. Procedia Computer Science, 163, 125-133.

Marcus, G. (2023). Hoping for the best as AI evolves. Communications of the ACM, 66(4), 6-7.

Mathews TJ, Hamilton BE. Mean age of mothers is on the rise: United States, 2000–2014. NCHS data brief, no 232. Hyattsville, MD: National Center for Health Statistics. 2016. https://www.cdc.gov/nchs/products/databriefs/db232.htm

Mauhibah, R., & Karso, K. (2020, March). Student Difficulties in Addition and Subtraction of Two Digit Numbers. In International Conference on Elementary Education (Vol. 2, No. 1, pp. 618-623).

McAllister, G., & White, G. R. (2015). Video game development and user experience. In Game user experience evaluation (pp. 11-35). Springer, Cham.

McBride, D. L., Zollman, D., & Rebello, N. S. (2010). Method for analyzing students' utilization of prior physics learning in new contexts. Physical Review Special Topics-Physics Education Research, 6(2), 020101.

McClenaghan, E. (2023), The Kruskal-Wallis Test, Technology Network, Informatics online publication. https://www.technologynetworks.com/informatics/articles/the-kruskal-wallis-test-370025

McGeoch, J. A. (1942). The psychology of human learning.

McMahan A (2003) Immersion, engagement and presence: a method for analyzing 3-D video games. In: Wolf MJP, Perron B (eds) The video game theory reader. Routledge, New York

Miller, P. C., Shim, J. E., & Holden, G. W. (1998). Immediate contextual influences on maternal behavior: Environmental affordances and demands. Journal of Environmental Psychology, 18(4), 387-398.

Milligan, G. W., & Hirtle, S. C. (2012). Clustering and classification methods. Research Methods in Psychology, 2, 189-210.

Minogue, J., & Jones, M. G. (2006). Haptics in education: Exploring an untapped sensory modality. Review of educational research, 76(3), 317-348.

Newman, T.M. (1994). The effectiveness of a multisensory approach for teaching addition to children with Down syndrome. Montreal: Unpublished master's thesis, McGill University.

Norman, D. A., & Nielsen, J. (2010). Gestural interfaces: a step backward in usability interactions, 17(5), 46-49.

Norman, D. (2004). Affordances and design. Unpublished article, available online at: http://www. jnd. org/dn. mss/affordances-and-design. html.

O'Brien, R. G., & Kaiser, M. K. (1985). MANOVA method for analyzing repeated measures designs: an extensive primer. Psychological bulletin, 97(2), 316.

O'Brien, H. L., & Toms, E. G. (2008). What is user engagement? A conceptual framework for defining user engagement with technology. Journal of the American Society for Information Science and Technology, 59(6), 938-955.

Oxford, R. (1990). Language learning strategies What every teacher should know. Heinle & Heinle Publishers.

Pande, M., & Bharathi, S. V. (2020). Theoretical foundations of design thinking–A constructivism learning approach to design thinking. Thinking Skills and Creativity, 36, 100637.

Papert, S. (1980). Mindstorms. Children, Computers and Powerful Ideas. New York: Basic books.

Pardos, Z. A., Heffernan, N. T., & Koedinger, K. R. (2013). Using machine learning to predict student performance in intelligent tutoring systems. User Modeling and User-Adapted Interaction, 23(1-2), 105-140.

Passolunghi, M. C., Vercelloni, B., & Schadee, H. (2007). The precursors of mathematics learning: Working memory, phonological ability and numerical competence. Cognitive Development, 22, 165-184.

Pedro, M. O., Baker, R., Bowers, A., & Heffernan, N. (2013, July). Predicting college enrollment from student interaction with an intelligent tutoring system in middle school. In Educational Data Mining 2013.

Peer, J. (2016) A Methodology for the Design of Experiments in Computational Intelligence with Multiple Regression Models 4: e 2721 Published Online National Library of Medicine, Dec. 1 2016 doi http://10.7717/peerj.2721

Penman, Paul, 2022, Binary Logistic Regression, Data Science Institute, as of 10 February 2023 https://www.datascienceinstitute.net/blog/binary-logistic-regression-an-introduction#:~:text=Binary%20logistic%20regression%20models%20the,or%20presence%20and%20so%20on.

Piaget, J., & Inhelder, B. (2008). The psychology of the child. Basic books.

Picard, R. W., Papert, S., Bender, W., Blumberg, B., Breazeal, C., Cavallo, D., ... & Strohecker, C. (2004). Affective learning—a manifesto. BT technology journal, 22(4), 253-269.

Pituch, K. A., & Stevens, J. P. (2015). Assumptions in MANOVA. In Applied multivariate statistics for the social sciences (pp. 219-264). Routledge.

Prescott, T. J., Diamond, M. E., and Wing, A. M. (2011). Active touch sensing. Philos. Trans. R. Soc. Lond. B. Biol. Sci. 366, 2989–2995. doi: 10.1098/rstb.2011.0167

Perkins, D. N., & Salomon, G. (1992). Transfer of learning. International encyclopedia of education, 2, 6452-6457.

Prensky, M. (2003). Digital game-based learning. Computers in Entertainment (CIE), 1(1), 21-21.

Pupo, M. (1994). Teaching intellectually disabled students addition through a multisensory approach. Montreal: Unpublished Master's thesis, McGill University.

Ray, Tierman (2022), Meta's LeCun is finally coming around to the things I said years ago. ZDNet Podcast with Gary Marcus. Oct. 2022

Rajput, D., Wang, WJ. & Chen, CC. Evaluation of a decided sample size in machine learning applications. BMC Bioinformatics 24(48), 2023. https://doi.org/10.1186/s12859-023-05156-9

Raskin, J. D. (2002). Constructivism in psychology: Personal construct psychology, radical constructivism, and social constructionism. American Communication Journal, 5(3), 1-25.

Rebello, N. S., Cui, L., Bennett, A. G., Zollman, D. A., & Ozimek, D. J. (2017). Transfer of learning in problem solving in the context of mathematics and physics. In Learning to solve complex scientific problems (pp. 223-246). Routledge.

Reichardt C.S. Quasi-Experimentation: A Guide to Design and Analysis. The Guilford Press; 2019.

Reaven, J., Blakeley-Smith, A., Culhane-Shelburne, K., & Hepburn, S. (2012). Group cognitive behavior therapy for children with high-functioning autism spectrum disorders and anxiety: A randomized trial. Journal of Child Psychology and Psychiatry, 53(4), 410-419.

Reigeluth, C. M. (2016). Instructional theory and technology for the new paradigm of education. Revista de Educación a Distancia (RED), (50).

Reiner, M. (1999). Conceptual construction of fields through tactile interface. Interactive Learning Environments, 7(1), 31-55.

Rivas, A., Fraile, J. M., Chamoso, P., González-Briones, A., Rodríguez, S., & Corchado, J. M. (2019). Students performance analysis based on machine learning techniques. In Learning Technology for Education Challenges: 8th International Workshop, LTEC 2019, Zamora, Spain, July 15–18, 2019, Proceedings 8 (pp. 428-438). Springer International Publishing.

Rogoff, B. (1990). Apprenticeship in thinking: Cognitive development in social context. Oxford university press. 5860/choice.28-0612.

Rogoff, B., Mistry, J., Göncü, A., Mosier, C., Chavajay, P., & Heath, S. B. (1993). Guided participation in cultural activity by toddlers and caregivers. Monographs of the Society for Research in Child development, i-179.

Rosenblatt, F., & Papert, S. (2021). Perceptron (Vol. 9). April. (Original chapter from Psychology Review, 1958)

Rowe, J. P., Shores, L. R., Mott, B. W., & Lester, J. C. (2011). Integrating learning, problem solving, and engagement in narrative-centered learning environments. International Journal of Artificial Intelligence in Education, 21(1-2), 115-133.

Russell, S. J., & Norvig, P. (2016). Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,

Sadouk, L., Gadi, T., & Essoufi, E. H. (2018). A novel deep learning approach for recognizing stereotypical motor movements within and across subjects on the autism spectrum disorder. Computational intelligence and neuroscience, 2018.

Sanusi, I. T., Oyelere, S. S., Vartiainen, H., Suhonen, J., & Tukiainen, M. (2022). A systematic review of teaching and learning machine learning in K-12 education. Education and Information Technologies, 1-31.

Sauro, J., & Lewis, J. R. (2016). Quantifying the user experience: Practical statistics for user research. Morgan Kaufmann.

Sathian, K. (1998). Perceptual learning. Current Science, 75, 451–456.

Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O. P., Tiwari, A., ... & Lin, C. T. (2017). A review of clustering techniques and developments. *Neurocomputing*, *267*, 664-681.

Schank, R. C., & Berman, T. (2006). Living stories: Designing story-based educational experiences. Narrative Inquiry, 16(1), 220-228.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural networks, 61, 85-117.

Schoultz, J., R. S.lj., and J. Wyndhamn, Heavenly talk: Discourse, artifacts, and children’s understanding of elementary astronomy, Hum. Dev. 44 (2001)

103–118, http://dx.doi.org/10.1159/000057050.

Schofield, J. W., Evans-Rhodes, D., & Huber, B. R. (1990). Artificial intelligence in the classroom: The impact of a computer-based tutor on teachers and students. Social Science Computer Review, 8(1), 24-41.

Sekeroglu, B., Dimililer, K., & Tuncal, K. (2019, March). Student performance prediction and classification using machine learning algorithms. In Proceedings of the 2019 8th International Conference on Educational and Information Technology (pp. 7-11).

Seminara, L., Gastaldo, P., Watt, S. J., Valyear, K. F., Zuher, F., & Mastrogiovanni, F. (2019). Active haptic perception in robots: a review. Frontiers in neurorobotics, 13, 53.

Shemshack, A., & Spector, J. M. (2020). A systematic literature review of personalized learning terms. Smart Learning Environments, 7(1), 33.

Shin, J., Lee, H., & Lee, K. (2016). Transfer learning for object recognition in the real world. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3829-3837).

Siemens, George. (2005). Connectivism: A learning theory for the digital age. International Journal of Instructional Technology and Distance Learning. Online] retrieved from: http://www. idtl. org/Journal/Jam _05/article01. html.

Sierpinska, A., & Kilpatrick, J. (Eds.). (2012). Mathematics education as a research domain: A search for identity: An ICMI study (Vol. 4). Springer Science & Business Media.

Skelly, T.C., Fries, K., Linnett, B., Nass, C., & Reeves, B. (1994). Seductive interfaces: Satisfying a mass audience. In C. Plaisant (Ed.), Proceedings of the Conference on Human Factors in Computing Systems (pp. 359–360). New York: ACM.

Sosniak, L. A. (1994). Bloom's taxonomy. L. W. Anderson (Ed.). Chicago, IL, USA: Univ. Chicago Press.

Syriopoulou-Delli, C. K., & Gkiolnta, E. (2022). Review of assistive technology in the training of children with autism spectrum disorders. International Journal of Developmental Disabilities, 68(2), 73-85.

Tall, D. (2013). How humans learn to think mathematically: Exploring the three worlds of mathematics. Cambridge University Press.

Takatalo, J., Häkkinen, J., & Nyman, G. (2015). Understanding presence, involvement, and flow in digital games. In Game user experience evaluation (pp. 87-111). Springer, Cham.

Thibault, M., & Hamari, J. (2021). Seven points to reappropriate gamification. In Transforming Society and Organizations through Gamification: From the Sustainable Development Goals to Inclusive Workplaces (pp. 11-28). Cham: Springer International Publishing.

Thomas, E.H., Galambos, N.: What satisfies students? Mining student-opinion data with regression and decision tree analysis. Res. High. Educ. **45**(3), 251–269 (2004)

Thompson, J. R. (2001). Estimating equations for kappa statistics. Statistics in medicine, 20(19), 2895-2906.

Turney, Shaun. (2022) Pearson Correlation Coefficient (r): | Guide & Examples
Published https://www.scribbr.com/statistics/pearson-correlation-coefficient/

Turing AM (1950) Computing machinery and intelligence. Mind 59(236): 433–460. doi:10.1093/mind/LIX.236.433

Von Collani, E., & Dräger, K. (2001). Binomial distribution handbook for scientists and engineers. Springer Science & Business Media.

Wang, J., Mendori, T., & Hoel, T. (2019). Strategies for multimedia learning object recommendation in a language learning support system: Verbal learners vs. visual learners. International Journal of Human–Computer Interaction, 35(4-5), 345-355.

Wasserstein, R. (2010). George Box: A model statistician. Significance, 7(3), 134-135.

Webb, B. (2001). Can robots make good models of biological behavior? Behavioral and brain sciences, 24(6), 1033-1050.

Winkler, R., & Roos, J. (2019). Bringing AI into the classroom: Designing smart personal assistants as learning tutors. Association for Information Systems, ICIS 2019 Proceedings 10

Wohlin, C., Höst, M., & Henningsson, K. (2003). Empirical research methods in software engineering. In Empirical methods and studies in software engineering (pp. 7-23). Springer, Berlin, Heidelberg.

VanLehn, K. (2006). The behavior of tutoring systems. International journal of artificial intelligence in education, 16(3), 227-265.

Vartiainen, H., Nissinen, S., Pöllänen, S., & Vanninen, P. (2018). Teachers' insights into connected learning networks: Emerging activities and forms of participation. AERA Open, 4(3), 2332858418799694.

Vartiainen, H., Tedre, M., & Valtonen, T. (2020). Learning machine learning with very young children: Who is teaching whom?. International journal of child-computer interaction, 25, 100182.

Verhelst, M., & Moons, B. (2017). Embedded deep neural network processing: Algorithmic and processor techniques bring deep learning to IoT and edge devices. IEEE Solid-State Circuits Magazine, 9(4), 55-65.

Vygotsky, L. S., & Cole, M. (1978). Mind in society: Development of higher psychological processes. Harvard university press.

Wei, X., Sun, S., Wu, D., & Zhou, L. (2021). Personalized online learning resource recommendation based on artificial intelligence and educational psychology. Frontiers in psychology, 12, 767837.

Wertsch, J. V., Daniels, H., Cole, M., & Wertsch, J. V. (2007). The Cambridge companion to Vygotsky. New York: Cambridge, 5-31.

Yang, Q., Scuito, A., Zimmerman, J., Forlizzi, J., & Steinfeld, A. (2018, June). Investigating how experienced UX designers effectively work with machine learning. In Proceedings of the 2018 designing interactive systems conference (pp. 585-596).

Appendix A: Arduino Code for Paper Prototype

Code 1 Arduino ‘Sketch’:

The following is intended for 3 force sensors that buffer readings and send pressure values via WiFi to the database.

```
#include <SPI.h>
#include <WiFiNINA.h>
#include <Ethernet.h>
#include "arduino_secrets.h"

#define asrpin A0
#define bsrpin A1
#define csrpin A2
#define arrayLength 100
#define sendLength 500

//Variable to store FSR value
int fsrreading_01;
int fsrreading_02;
int fsrreading_03;

///////please enter your sensitive data in the Secret tab/arduino_secrets.h
char ssid[] = SECRET_SSID;     // your network SSID (name)
char pass[] = SECRET_PASS;     // your network password (use for WPA, or use as key for WEP)
int status = WL_IDLE_STATUS;    // the Wifi radio's status

//IPAddress server(00.00.00.00);  // IP of the MySQL *server* here
//char user[] = SECRET_SERV;              // MySQL user login username
//char password[] = SECRET_SERVPASS;        // MySQL user login password

//// data stored local in array
unsigned int saveData = 0;
unsigned int sensorData = 0;
int dataArray[arrayLength];
int toSend[sendLength];
unsigned int arrayIndex = 0;
```

```
//int query[128] = {fsrreading_01, fsrreading_02, fsrreading_03};

/// wifi client
WiFiClient client;
//MySQL_Connection conn((Client *)&client);

void setup() {
 //Initialize serial and wait for port to open:
 Serial.begin(115200);
 while (!Serial) {
  ; // wait for serial port to connect. Needed for native USB port only
 }

 // check for the WiFi module:
 if (WiFi.status() == WL_NO_MODULE) {
  Serial.println("Communication with WiFi module failed!");
  // don't continue
  while (true);
 }

 String fv = WiFi.firmwareVersion();
 if (fv < WIFI_FIRMWARE_LATEST_VERSION) {
  Serial.println("Please upgrade the firmware");
 }

 // attempt to connect to Wifi network:
 while (status != WL_CONNECTED) {
  Serial.print("Attempting to connect to WPA SSID: ");
  Serial.println(ssid);
  // Connect to WPA/WPA2 network:
  status = WiFi.begin(ssid, pass);

  // wait 10 seconds for connection:
  delay(10000);
 }

 // you're connected now, so print out the data:
 Serial.print("You're connected to the network");
 //printCurrentNet();
```

```
  //printWifiData();

   if (client.connect("IP for DB server ie 00.000.000.0", 80)) {

     client.println("POST /MattIndex.php HTTP/1.1");
     client.println("Host: www.yourserver.com");
     client.println("Content-Type: application/x-www-form-urlencoded; charset=UTF-
8");
  if (client.connect("IP for DB server ie 00.000.000.0", 80)) {

    client.println("POST /filename.php HTTP/1.1"); //www.yourserver/pathname.php
    client.println("Host: www.yourserver.com");
    client.println("Content-Type: application/x-www-form-urlencoded; charset=UTF-8");

    //dataArray[0] //storing the sensor index
    //dataArray[1] //store the actual sensor value for previous index


    //data = ["1,100", "2,200"]

    String toSend = "";
    //for(int i = 0; i< arrayLength; i++){


    for(int ct = 0; ct <= arrayLength; ct= ct + 2)
    {
      toSend = toSend + "d[]=" +  dataArray[ct] + "%2C" + dataArray[ct+1];

      if((ct+2) <= arrayLength)
      {
        toSend = toSend + "&";
      }
      //d[]=1%2C2
    }

   }
    client.println("Content-Length: " + sendLength );
    client.println();
    client.print(toSend);
    client.stop();
```

```
 }
 else{
  Serial.println("Connection failed.");
}

void loop() {
 fsrreading_01 = analogRead(asrpin);
 fsrreading_02 = analogRead(bsrpin);
 fsrreading_03 = analogRead(csrpin);
 static uint32_t tStart = millis(); // ms; start time
 const uint32_t DESIRED_PERIOD = 1000; // ms
 uint32_t tNow = millis(); // ms; time now
 if (tNow - tStart >= DESIRED_PERIOD) {
  tStart += DESIRED_PERIOD; // update start time to ensure consistent and near-exact
period

  Serial.println("taking sample");
 }
  if (analogRead(fsrreading_01) == HIGH)       //1 is pressed
 {
  saveData = 1;
  fsrreading_01 = sensorData;

 }

 if (analogRead(fsrreading_02) == HIGH)       //2 is pressed
 {
  saveData = 2;
  fsrreading_02 = sensorData;
 }

 if (analogRead(fsrreading_03) == HIGH)       //3 is pressed
 {
  saveData = 3;
  fsrreading_03 = sensorData;

 }
      //function to save sensor number and data
 dataArray[arrayIndex] = saveData;          //save switch number
 arrayIndex++;                        //move index to next array position
 dataArray[arrayIndex] = sensorData;   //save sensor data to array
 arrayIndex++;                        //move index to next array position

while (client.connected() || client.available())
 {
  Serial.write(client.read());
 }

  // check the network connection once every 10 seconds:
 delay(1000);

}
```

Appendix B Math - Color Game in JavaScript

```
// math color game javascript version 4


const logButtonEvents = false;


// frustration //////////////////////////////////


const advance = 200; // if the frustration level is below this value then increase the
difficulty


const retreat = 400; // if the frustration level is above this value then decrease the
difficulty


const weights = { pressure:0.0001, motion:0.0001, correct:-100, incorrect:100,
time:0.1 };


const updateFrustration = options => {
   if (inSession) {
       if ('pressure' in options) { // a button is being pressed; adjust the
frustration level (options.pressure is the magnitude of the pressure on the button)
           frustration += (options.pressure - 500) * weights.pressure;
       }
       else if ('motion' in options) { // the toy is being shaken; adjust the
frustration level (options.motion is the magnitude of the shaking)
           frustration += (options.motion - 100) * weights.motion;
       }
       else if ('correct' in options) { // a question was answered; adjust the
frustration score (options.correct is true if the answer was correct, false otherwise)
           frustration += (options.correct ? weights.correct : weights.incorrect) *
(difficulty + 1);
```

```
       }

       else if ('ctime' in options) { // another second has passed; adjust the
frustration (options.ctime indicates how many seconds have pased for the current
challenge)


           frustration += options.ctime * weights.time;

       }

   }

};


// mqtt /////////////////////////////////////////

const mqtt = require('mqtt');



const HOST = 'mqtts://7ddd3913d9a24f80856c69016e3fd7ab.s1.eu.hivemq.cloud:8883';

const SUBTOPIC = 'Toy-ef07e4/events';

const PUBTOPIC = 'Toy-ef07e4/commands';

const USERNAME = 'username';

const PASSWORD = 'password';


let options = {

   keepalive: 60,

   reconnectPeriod:1000,

   connectTimeout: 30 * 1000,

   clientId: '7cdfa1ef07e4',

   username: USERNAME,

   password: PASSWORD

};
```

```
let mqttClient = false;

const connect = async () => {
   console.log(`Connecting to ${HOST}`);
   mqttClient = await mqtt.connect(HOST, options);

   mqttClient.on('reconnect', () => { console.log('Reconnecting...'); });

   mqttClient.on('connect', async () => {
       console.log('Connected');
       await playSequence('123456789', 0.1, true);
       await playSequence('123456789', 0.1, false);
       playGame();
   });

   mqttClient.subscribe(SUBTOPIC, { qos:1 });

   mqttClient.on('error', err => {
       console.log('Error: ', err);
       mqttClient.end();
   });

   mqttClient.on("message", (topic, payload, packet) => {
       payload = payload.toString();

       if (payload.substring(0, 9) == 'buttons/1') onButtonEvent(0,
payload.substring(10));
       if (payload.substring(0, 9) == 'buttons/2') onButtonEvent(1,
payload.substring(10));
```

```
       if (payload.substring(0, 9) == 'buttons/3') onButtonEvent(2,
payload.substring(10));

       if (payload.substring(0, 9) == 'buttons/4') onButtonEvent(3,
payload.substring(10));

       if (payload.substring(0, 9) == 'buttons/5') onButtonEvent(4,
payload.substring(10));

       if (payload.substring(0, 9) == 'buttons/6') onButtonEvent(5,
payload.substring(10));

       if (payload.substring(0, 9) == 'buttons/7') onButtonEvent(6,
payload.substring(10));

       if (payload.substring(0, 9) == 'buttons/8') onButtonEvent(7,
payload.substring(10));

       if (payload.substring(0, 9) == 'buttons/9') onButtonEvent(8,
payload.substring(10));


       if (payload.substring(0, 3) == 'imu') onImuEvent(payload.substring(4));

   });

};


// game play ////////////////////////////////////


// complete collection of all questions to be filled in by initializeChallenges

let challenges = [];


// inSession is true when in the midst of a series of challenges

let inSession = false;


// this is where the answer is typed in

let answer = '';
```

```
// frustration level
let frustration = 0;

// the current level of difficulty; used to select challenges
let difficulty = 0;
// number of seconds since the start of the session and challenge
let sessionTime = 0;
let challengeTime = 0;

// helpers
const sleep = ms => new Promise(resolve => setTimeout(resolve, ms));

const setColorLed = async (b, lit) => {
   await mqttClient.publish(PUBTOPIC, `buttons/${b}/led1 ${lit ? 'true' : 'false'}`, { qos:1, retain:false });
   console.log(`setColorLed ${b} ${lit ? 'on' : 'off'}`);
};

const setNumberLed = async (b, lit) => {
   await mqttClient.publish(PUBTOPIC, `buttons/${b}/led2 ${lit ? 'true' : 'false'}`, { qos:1, retain:false });
   console.log(`setNumberLed ${b} ${lit ? 'on' : 'off'}`);
};

const setCenterDisplay = async ch => {
   await mqttClient.publish(PUBTOPIC, `center/display ${JSON.stringify(ch == '#' ? '::' : ch == '_' ? ' ' : ch)}`, { qos:1, retain:false });
   console.log(`setCenterDisplay ${ch}`);
};
```

```
// playCommands takes a string that encodes a series of commands
// commands are separated by spaces
// there are 3 types of commands LED, center, and sleep
// the LED command starts with a digit ('1' - '9'), followed by a 'c' (= color led) or
'n' (= number led), followed by '+' (= On) or '-' (= Off)
//   example: '3c+' turns button 3's color LED On
//   example: '5n-' turns button 5's number LED Off
// the center command starts with a 'c', followed by the character it should display
('#' = '::', '_' = ' ')
//   example: 'cH' displays the 'H" character on the center button
//   example: 'c_' displays the nothing on the center button
//   example: 'c#' displays the '::' in the center button
// the sleep command starts with an 's', followed by the number of seconds to sleep
//   example: 's7' sleeps for 7 seconds
//   example: 's2.3' sleeps for 2.3 seconds
//
// calling playCommands('1c+ s1 1c- s1 c=') will turn button1's color LED on, pause
for 1 second, turn button1's color LED off, pause for 1 second, then display '=' on
the center button.
//
// playCommands returns a promise which will resolve once all the commands have been
executed.  If playCommands is called while a different set of commands is being
played, the call will fail.


let playingCommands = false;


const playCommands = (commands) => {
   return new Promise(async (resolve, reject) => {
       if (!playingCommands) {
           playingCommands = true;
           let index = 0;
           const cmds = commands.split(' ');
           for (let i = 0; i < cmds.length; ++i) {
               if (cmds[i].length == 0) continue;
               let chars = [...cmds[i]]; // break current command into a array of
characters
               if (chars[0] >= '1' && chars[0] <= '9') { // button commands start with
a digit
                   if (chars[1] == 'c') await setColorLed(chars[0].charCodeAt() -
'0'.charCodeAt(), chars[2] == '+');
                   else await setNumberLed(chars[0].charCodeAt() - '0'.charCodeAt(),
chars[2] == '+');
               }
               else if (chars[0] == 'c') await setCenterDisplay(chars[1]); // center
commands start with 'c'
               else if (chars[0] == 's') await sleep(parseFloat(cmds[i].substring(1))
* 1000); // sleep commands start with 's'
           }
           resolve('Commands completed');
           playingCommands = false;
       }
       else reject(new Error('commands already playing'));
   });
};

// playSequence is a simpler version of playCommands
// it takes three arguments, sequence, period, and color.
// the sequence contains digits and center characters (+,-,=,:,*,#,H)
// period is the number of seconds LEDs are turned on for
// if color is true then the color LEDs are used, otherwise the number LEDs are used

//   example: playSequence('2+3=5', 1, false) will do the following:

//              - turn button2's number LED On for 1 second then turn it Off

//              - display a '+' on the center button then turn off the center display

//              - turn button3's number LED On for 1 second then turn it Off

//              - display a '=' on the center button then turn off the center display

//              - turn button5's number LED On for 1 second then turn it Off

//

// playSequence essentially transcribes the sequence string to a commands string and
then calls playCommands


const playSequence = (sequence, period, color) => {

   let commands = '';

   let cn = color ? 'c' : 'n';

   for (let i = 0; i < sequence.length; ++i) {

       let ch = sequence.charAt(i);

       if (ch >= '1' && ch <= '9') commands += `${ch}${cn}+ s${period} ${ch}${cn}- `;

       else commands += `c${ch} s${period} c_ `;

   }

   return playCommands(commands);

};


const flashButtons = (buttons, period, color) => {

   let commands = '';

   let cn = color ? 'c' : 'n';

   for (let i = 0; i < buttons.length; ++i) {

       let ch = buttons.charAt(i);

       if (ch >= '1' && ch <= '9') commands += `${ch}${cn}+ `;
   }
   commands += `s${period} `;
   for (let i = 0; i < buttons.length; ++i) {
       let ch = buttons.charAt(i);
       if (ch >= '1' && ch <= '9') commands += `${ch}${cn}- `;
   }
   return playCommands(commands);
};

// answering ////////////////////////////////////

let answerInterval = false;

const onButtonPressed = r => {
   if (answerInterval) {
       answer += (r + 1).toString();
       console.log(`Button ${r + 1} pressed;  answer: ${answer}`);
   }
};

const listenForAnswer = (correctAnswer, timeout) => {
   let countdown = 1000 * timeout;
   return new Promise((resolve, reject) => {
       if (answerInterval) {
           clearInterval(answerInterval);
           answerInterval = false;
       }
       answer = ''; // answer will be filled by button presses
       let correctLength = correctAnswer.length;
       answerInterval = setInterval(() => {
           if (answer.length >= correctLength) {
               clearInterval(answerInterval);
               answerInterval = false;
               resolve(answer.substring(0, correctAnswer.length) == correctAnswer ?
'correct' : 'incorrect');
           }
           else {
               countdown -= 100;
               if (countdown <= 0) {
                   clearInterval(answerInterval);
                   answerInterval = false;
                   resolve('timeout');
               }
           }
       }, 100);
   });
};

// initialization ///////////////////////////////

const initializeChallenges = () => {
   challenges = [];
   for (let i = 0; i < 8; ++i) challenges[i] = [];

   for (let a = 1; a <= 9; ++a)
```

```
        for (let b = 1; b <= 9; ++b) {
            if (a != b) challenges[0].push({ question:`${a}${b}`, answer:`${a}${b}` });
            if (a + b <= 9) challenges[1].push({ question:`${a}+${b}=${a + b}`, answer:`${a}${b}${a + b}` });
            if (a - b <= 9 && a - b >= 1) challenges[3].push({ question:`${a}-${b}=${a - b}`, answer:`${a}${b}${a - b}` });
            if (a * b <= 9) challenges[5].push({ question:`${a}*${b}=${a * b}`, answer:`${a}${b}${a * b}` });
            for (let c = 1; c <= 9; ++c) {
                if (a != b && b != c) challenges[2].push({ question:`${a}${b}${c}`, answer:`${a}${b}${c}` });
                if (a + b + c <= 9) challenges[5].push({ question:`${a}+${b}+${c}=${a + b + c}`, answer:`${a}${b}${c}${a + b + c}` });
                if (a - b - c <= 9 && a - b - c >= 1) challenges[5].push({ question:`${a}-${b}-${c}=${a - b - c}`, answer:`${a}${b}${c}${a - b - c}` });
                if (a * b * c <= 9) challenges[7].push({ question:`${a}*${b}*${c}=${a * b * c}`, answer:`${a}${b}${c}${a * b * c}` });
                for (let d = 1; d <= 9; ++d) {
                    if (a != b && b != c && c != d) challenges[4].push({ question:`${a}${b}${c}${d}`, answer:`${a}${b}${c}${d}` });
                    for (let e = 1; e <= 9; ++e)
                        if (a != b && b != c && c != d && d != e) challenges[6].push({ question:`${a}${b}${c}${d}${e}`, answer:`${a}${b}${c}${d}${e}` });
                }
            }
        }
};


// event handlers ///////////////////////////////
```

```
const onImuEvent = imu => {
   imu = JSON.parse(imu);
   updateFrustration(imu);
   if (!imu) {
       console.log("dropped imu data")
       return false;
     }
};

let buttonPressed = [false, false, false, false, false, false, false, false, false];

// new handler for buttons/N events
const onButtonEvent = (index, button) => {
   if (logButtonEvents) console.log(`onButtonEvent(${index}, ${button})`);
   button = JSON.parse(button);
   if (button.pressure > 40) {
       if (!buttonPressed[index]) {
           onButtonPressed(index);
           buttonPressed[index] = true;
       }
       updateFrustration(button);
   }
   else if (button.pressure < 30) buttonPressed[index] = false;
};

const playGame = async () => {
   if (inSession) return;
   inSession = true;
   let correctCount = 0;

   let incorrectCount = 0;

   let done = false;


   difficulty = 0;

   frustration = 0;

   sessionTime = 0;


   while (!done) {

       let list = challenges[difficulty];

       const index = Math.floor(0.999999 * Math.random() * list.length);

       const challenge = list[index];


       console.log('----------------------------------------------------------------
--------');

       console.log(`difficulty: ${difficulty}, challenge: ${challenge.question}`);


       answer = '';


       await sleep(2000); // pause before presenting a challenge


       challengeTime = 0;

       await playSequence(challenge.question, 0.8, (difficulty %2) == 0);


       const result = await listenForAnswer(challenge.answer, 300);

       if (result == 'correct') {

           updateFrustration({ correct:true });
           await flashButtons(challenge.answer, 0.8, (difficulty %2) == 0);
           correctCount++;
       }
       else {
           updateFrustration({ correct:false });
           await playSequence('H', 2, true);
           incorrectCount++;
       }

       console.log(`frustration: ${frustration}`);
       if ((correctCount + incorrectCount) % 5 == 0) { // every 5th question move up
or down based on frustration level
           if (frustration < advance) difficulty++;
           else if (difficulty > 0 && frustration > retreat) difficulty--;
           if (difficulty > 7) done = true;
       }
   }
   console.log('Game over');

   inSession = false;
};
const start = async () => {
   connect();
   initializeChallenges();
   setInterval(() => updateFrustration({ stime:sessionTime++,
ctime:challengeTime++ }), 1000);
};
start();
```

Appendix C Math - Color Game in Python

```
# Math_Color Game Python Machine Learning
#import asyncio
import time
from threading import Timer
import threading
import random
import json
import paho.mqtt.client as mqtt
import joblib

log_button_events = False
advance = 200
retreat = 300
weights = {'pressure': 0.0001, 'motion': 0.0001, 'correct': -100, 'incorrect': 100, 'time': 0.1}
in_session = False
frustration = 0
difficulty = 0
press = 0
mot = 0
rwcount = 0
session_time = 0
challenge_time = 0
answer = ''
answering = False
playing_commands = False
connected = False
```

```
#challenges = []


#HOST = 'mqtts://d23d1608d163422f829353b161472b59.s2.eu.hivemq.cloud:8883'

HOST = '7ddd3913d9a24f80856c69016e3fd7ab.s1.eu.hivemq.cloud'

SUBTOPIC = 'Toy-ef07e4/events'

PUBTOPIC = 'Toy-ef07e4/commands'

USERNAME = 'username'

PASSWORD = 'password'


options = {

   'keepalive': 60,

   'reconnectPeriod': 1000,

   'connectTimeout': 30 * 1000,

   'clientId': '7cdfa1ef07e4',

   'username': USERNAME,

   'password': PASSWORD,

   'port':8883,

   'bind_address':""

}


mqtt_client = None



def update_frustration(options):

   global frustration, in_session, press, mot, rwcount

   if in_session:

       if 'pressure' in options:

           press += (options['pressure'] - 500)
       elif 'motion' in options:

           mot += (options['motion'] - 100)

       elif 'correct' in options:

           rwcount += weights['correct']

       #elif 'ctime' in options:

            #frustration += options['ctime'] * weights['time']




def connect():

   global mqtt_client

   print(f"Connecting to {HOST}")

   mqtt_client = mqtt.Client(options['clientId'])

   mqtt_client.tls_set(tls_version=mqtt.ssl.PROTOCOL_TLS)

   mqtt_client.on_reconnect = lambda client, userdata, flags, rc:
print('Reconnecting...')

   mqtt_client.on_connect = on_connect

   mqtt_client.on_message = on_message


   try:

       mqtt_client.username_pw_set(options['username'], options['password'])

       mqtt_client.connect(HOST, options['port'], options['keepalive'],
options['bind_address'])

       mqtt_client.loop_start()

   except Exception as e:

       print(f"Error: {e}")

       mqtt_client.disconnect()




def on_connect(client, userdata, flags, rc):
   global connected
   print('Connected')
   play_sequence('123456789', 0.1, True)
   play_sequence('123456789', 0.1, False)
   connected = True


def on_message(client, userdata, msg):
   global answer
   payload = msg.payload.decode('utf-8')
   if payload.startswith('buttons/'):
       button_index = int(payload[8]) - 1
       on_button_event(button_index, payload[10:])
   elif payload.startswith('imu'):
       imu_data = json.loads(payload[4:])
       on_imu_event(imu_data)


def play_commands(commands):
   global playing_commands
   if not playing_commands:
       playing_commands = True
       index = 0
       cmds = commands.split(' ')
       for cmd in cmds:
           if len(cmd) == 0:
               continue
           chars = list(cmd)
```

```
            if '1' <= chars[0] <= '9':
                if chars[1] == 'c':
                    set_color_led(int(chars[0]), chars[2] == '+')
                else:
                    set_number_led(int(chars[0]), chars[2] == '+')
            elif chars[0] == 'c':
                set_center_display(chars[1])
            elif chars[0] == 's':
                time.sleep(float(cmd[1:]))
        playing_commands = False


def play_sequence(sequence, period, color):
    commands = ''
    cn = 'c' if color else 'n'
    for ch in sequence:
        if '1' <= ch <= '9':
            commands += f'{ch}{cn}+ s{period} {ch}{cn}- '
        else:
            commands += f'c{ch} s{period} c_ '
    play_commands(commands)



def flash_buttons(buttons, period, color):
    commands = ''
    cn = 'c' if color else 'n'
    for ch in buttons:
        if '1' <= ch <= '9':
            commands += f'{ch}{cn}+ '
```

```
   commands += f's{period} '
   for ch in buttons:
       if '1' <= ch <= '9':
           commands += f'{ch}{cn}- '
   play_commands(commands)


def set_color_led(button, lit):
   payload = f'buttons/{button}/led1 ' + ('true' if lit else 'false')
   mqtt_client.publish(PUBTOPIC, payload, qos=1, retain=False)
   print(f'setColorLed {button} {"on" if lit else "off"}')


def set_number_led(button, lit):
   payload = f'buttons/{button}/led2 ' + ('true' if lit else 'false')
   mqtt_client.publish(PUBTOPIC, payload, qos=1, retain=False)
   print(f'setNumberLed {button} {"on" if lit else "off"}')


def set_center_display(ch):
   payload = 'center/display ' + ('::' if ch == '#' else ' ' if ch == '_' else ch)
   mqtt_client.publish(PUBTOPIC, payload, qos=1, retain=False)
   print(f'setCenterDisplay {ch}')


def sleep(seconds):
   time.sleep(seconds)
```

```
#initialize the game challenges
challenges = []

def initialize_challenges():
   global challenges
   challenges = [[] for _ in range(8)]
   for a in range(1, 10):
       for b in range(1, 10):
           if a != b:
               challenges[0].append({'question': f"{a}{b}", 'answer': f"{a}{b}"})
           if a + b <= 9:
               challenges[1].append({'question': f"{a}+{b}={a + b}", 'answer':
f"{a}{b}{a + b}"})
           if a - b <= 9 and a - b >= 1:
               challenges[3].append({'question': f"{a}-{b}={a - b}", 'answer':
f"{a}{b}{a - b}"})
           if a * b <= 9:
               challenges[5].append({'question': f"{a}*{b}={a * b}", 'answer':
f"{a}{b}{a * b}"})
           for c in range(1, 10):
               if a != b and b != c:
                   challenges[2].append({'question': f"{a}{b}{c}", 'answer':
f"{a}{b}{c}"})
               if a + b + c <= 9:
                   challenges[5].append({'question': f"{a}+{b}+{c}={a + b + c}",
'answer': f"{a}{b}{c}{a + b + c}"})
               if a - b - c <= 9 and a - b - c >= 1:
                   challenges[5].append({'question': f"{a}-{b}-{c}={a - b - c}",
'answer': f"{a}{b}{c}{a - b - c}"})
               if a * b * c <= 9:
                   challenges[7].append({'question': f"{a}*{b}*{c}={a * b * c}", 'answer': f"{a}{b}{c}{a * b * c}"})
               for d in range(1, 10):
                   if a != b and b != c and c != d:
                       challenges[4].append({'question': f"{a}{b}{c}{d}", 'answer': f"{a}{b}{c}{d}"})
                   for e in range(1, 10):
                       if a != b and b != c and c != d and d != e:
                           challenges[6].append({'question': f"{a}{b}{c}{d}{e}", 'answer': f"{a}{b}{c}{d}{e}"})
   print('Finished initializing challenges')


def on_imu_event(imu_data):
   global frustration
   update_frustration(imu_data)
   if not imu_data:
       print("dropped imu data")
       return False


button_pressed = [ False, False, False, False, False, False, False, False, False ]

def on_button_event(index, button):
   global button_pressed

   if log_button_events:
       print(f'onButtonEvent({index}, {button})')
   button = json.loads(button)
   if button['pressure'] > 40:
       if not button_pressed[index]:
           on_button_pressed(index)
           button_pressed[index] = True
       update_frustration(button)
   elif button['pressure'] < 30:
       button_pressed[index] = False


def on_button_pressed(index):
   global answer
   if answering:
       answer += str(index + 1)
       print(f'Button {index + 1} pressed;  answer: {answer}')


class RepeatTimer(Timer):
   def run(self):
       while not self.finished.wait(self.interval):
           self.function(*self.args, **self.kwargs)


def listen_for_answer(correct_answer, timeout):
   global answer, answering
   answer = ''
   answering = True
   correct_length = len(correct_answer)
   for t in range(0, 10 * timeout):
       time.sleep(0.1)
       if len(answer) >= correct_length:
           answering = False
           return 'correct' if answer[:correct_length] == correct_answer else
'incorrect'
   answering = False
   return 'timeout'

'''def check_answer(correct_answer, correct_length, timeout, countdown):
   global answer_interval, answer
   if len(answer) >= correct_length:
       answer_interval.cancel()
       answer_interval = None
       result = 'correct' if answer[:correct_length] == correct_answer else
'incorrect'

       print(f'Answer: {result}')
   else:
       countdown -= 100
       if countdown <= 0:
           answer_interval.cancel()
           answer_interval = None
           result = 'timeout'
           print(f'Answer: {result}')'''


def play_game():
   global answer,challenges,in_session,difficulty, frustration,
session_time,challenge_time,press,mot,rwcount
   if in_session:
       return
   in_session = True
   clf = joblib.load('newClass.pkl')
   correct_count = 0
   incorrect_count = 0
   done = False

   difficulty = 0
   frustration = 0
   session_time = 0

   while not done:
       question_list = challenges[difficulty]
       index = random.randrange(0, len(question_list)) #int(0.999999 * random.random()
* len(question_list))
       challenge = question_list[index]

       print('------------------------------------------------------------------------
--')
       print(f'difficulty: {difficulty}, challenge: {challenge["question"]}')
       answer = ''
       sleep(2)  # pause before presenting a challenge
       challenge_time = 0
       play_sequence(challenge['question'], 0.8, difficulty % 2 == 0)

       result = listen_for_answer(challenge['answer'], 300)
       if result == 'correct':
           update_frustration({'correct': True})
           flash_buttons(challenge['answer'], 0.8, difficulty % 2 == 0)
           correct_count += 1
       else:
           update_frustration({'correct': False})
           play_sequence('H', 2, True)
           incorrect_count += 1
       print(f'frustration: {frustration}')
       if (correct_count + incorrect_count) % 5 == 0:
           # prediction values for ML model
           prediction = clf.predict([[press, mot, rwcount, session_time,
difficulty]]).round(2)
           if prediction[0, 1] == 1:
               difficulty += 1
           elif difficulty > 0 and prediction[0, 2] == 1:
               difficulty -= 1
           if difficulty > 7:
               done = True
   print('Game over')
   in_session = False


def start():
   global connected
   initialize_challenges()
   connect()
   while not connected:
       time.sleep(1)
   print('Starting game play')
   play_game()
   threading.Timer(1,lambda: update_frustration({'stime': session_time, 'ctime':
challenge_time}))


start()
```

Appendix D Machine Learning Model

```
# Machine Learning Model
import sqlite3
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import datetime
import tensorflow as tf
import seaborn as sns; sns.set()
import joblib
import sklearn
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import silhouette_score
from sklearn import preprocessing
from datetime import datetime
import time


def preprocess_data(data):
   # Feature scaling
   scaler = StandardScaler()
   scaled_data = scaler.fit_transform(data)

```

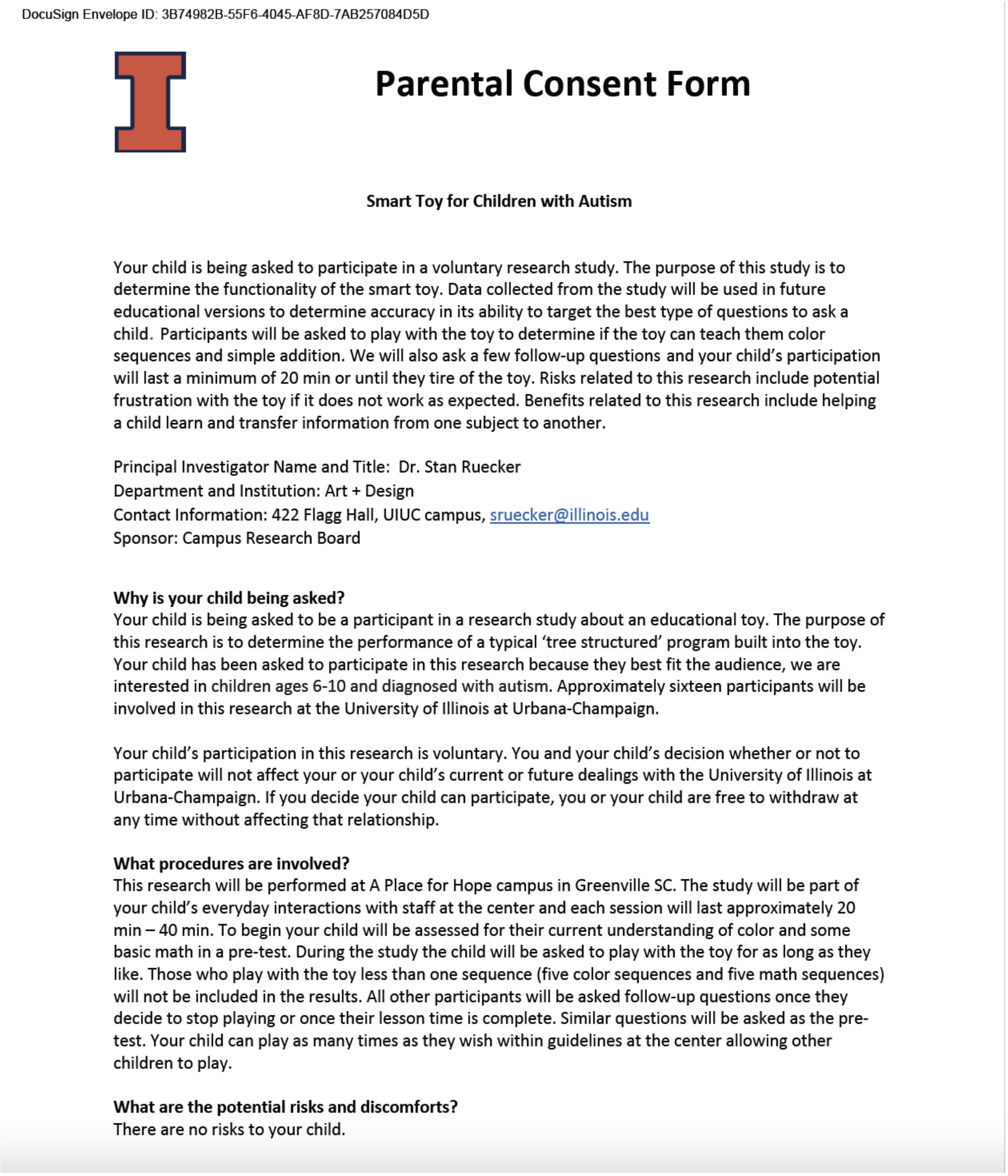
```
    # Reduce dimensionality using PCA (Principal Component Analysis)
    pca = PCA(n_components=5)
    reduced_data = pca.fit_transform(scaled_data)

    return reduced_data

# Connect to your SQLite database
conn = sqlite3.connect('messagesGT.db')  # Replace 'your_database.db' with your
database name
cursor = conn.cursor()

# From table messages and it has columns 'pressure', 'motion', 'difficulty',
'duration', 'performance', and 'label'
query = "SELECT time, motion, pressure, performance, difficulty, Targetmodel FROM
messages"
cursor.execute(query)

# Fetch all rows from the database
data = cursor.fetchall()

# Close the database connection
conn.close()

# Convert the data into a pandas DataFrame
df = pd.DataFrame(data, columns=[ 'time', 'motion', 'pressure', 'performance',
'difficulty', 'Targetmodel'])
df['time'] = df['time'].apply(lambda x: x.replace(' ', ''))
df['time'] = df['time'].str[:-4]
# motion = df.query('motion == motion')
```

```
#df["duration"] = ""
# Filter rows where 'led1' is True (start time)
#start_times = df[df['difficulty'] > 0]
#end_times = df[df['performance'] >= 0]



# Initialize a dictionary to store durations
durations = {}
newStartTime = datetime.strptime('00:00:00',"%H:%M:%S")
newEndTime = datetime.strptime('00:00:00',"%H:%M:%S")

#newStartTime = datetime.strptime(start_time,"%H:%M:%S")
#newEndTime = datetime.strptime(end_time,"%H:%M:%S")

# Iterate through start times and calculate durations
for index, row in df.iterrows():
   #if row['pressure'] > pressHigh:
       #pressHigh =
   shortenTime = row['time']

   if row['difficulty'] > 0:
       #shortenStartT = row['time'][:-4]
       newStartTime = datetime.strptime(shortenTime, "%H:%M:%S")


# Find the corresponding end time (next performance column value)
   # if (pd.notnull(row['performance'])):
   if row['performance'] >= 0:
       #shortenEndT = row['time'][:-4]
       newEndTime = datetime.strptime(shortenTime, "%H:%M:%S")
       #print(newEndTime, newStartTime)

   # calculate duration
   duration = newEndTime - newStartTime
   duration = duration.total_seconds()
   if duration > 0:
       durations[index] = duration

df['duration'] = df.index.to_series().map(durations)

features = df[['time', 'motion', 'pressure', 'performance', 'difficulty',
'Targetmodel', 'duration']]
features = features.drop(columns=['time'])


#for index, nstops in features.iterrows():
   #if nstops['difficulty'] > 0:
       #features = features[features["pressure"]!=0]

# add a new row with 0 0 0
firstFeatures = pd.DataFrame({'motion': [0], 'pressure': [0], 'performance': [0],
'difficulty':[0], 'Targetmodel':['easy'], 'duration':[0]})
features = pd.concat([firstFeatures, features])
features = features.reset_index(drop=True)
features['Targetmodel'].replace({'stay': 0, 'hard': 1, 'easy': 2}, inplace=True)
```

```
# Check for NaN values and replace with values from the row above
features.ffill(inplace=True)
#features.to_csv('newData.csv', index=False)

####features[['Targetmodel']] = features[["Targetmodel"]].astype(str).astype(int)
# Load a sample dataset (replace this with your database data loading)

X = features[['motion', 'pressure', 'performance', 'difficulty', 'duration']]
y = features[['Targetmodel']]

# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Standardize the data
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)


# Convert string features to numerical using LabelEncoder
label_encoder = preprocessing.LabelEncoder()
 # Encode labels in column 'Targetmodel'.
y_train_one_hot = y_train = label_encoder.fit_transform(y_train)
y_test_one_hot = y_test = label_encoder.fit_transform(y_test)


# Build the model
```

```
model = tf.keras.Sequential([
   tf.keras.layers.Dense(16, activation='relu', input_shape=(X_train.shape[1],)),
   tf.keras.layers.Dense(16, activation='relu'),
   tf.keras.layers.Dense(3, activation='softmax')

])
# Compile the model
model.compile(optimizer='RMSprop', loss='sparse_categorical_crossentropy',
metrics=['accuracy'])

# Train the model ------ change epochs if needed ------
model.fit(X_train, y_train_one_hot, epochs=20, batch_size=32, validation_split=0.2)

# Evaluate the model on the test set
loss, accuracy = model.evaluate(X_test, y_test_one_hot)
print(f'Test Loss: {loss}, Test Accuracy: {accuracy}')

'''
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(1, len(acc) + 1)

plt.plot(epochs, acc, 'bo', label='Training acc')
plt.plot(epochs, val_acc, 'b', label='Validation acc')
plt.title('Training and validation accuracy')
plt.legend()

plt.figure()

plt.plot(epochs, loss, 'bo', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend()

plt.show()

def predict_single_observation_if_condition(model, scaler, label_encoder, observation, condition_variable):
   if condition_variable:
       # Convert string feature to numerical using LabelEncoder
       observation[0] = label_encoder.transform([observation[0]])[0]

       # Standardize the observation
       scaled_observation = scaler.transform([observation])

       # Make a prediction using the trained model
       prediction_probabilities = model.predict(scaled_observation)

       # Get the predicted class (argmax of the probabilities)
       predicted_class = np.argmax(prediction_probabilities)

       return predicted_class
   else:
       return None


# Example usage:
# Replace `model`, `scaler`, and `label_encoder` with your trained model, scaler, and label_encoder
# Replace `observation_to_predict` with the actual observation you want to predict
# Set condition_variable to True or False based on your condition
observation_to_predict = ['stay', 3.1, 1.7, 3, 0.34]
condition_variable = True


predicted_class = predict_single_observation_if_condition(model, scaler, label_encoder, observation_to_predict, condition_variable)


if predicted_class is not None:
   print(f'The predicted class for the observation is: {predicted_class}')
else:
   print('Prediction skipped due to the condition.')
'''
joblib.dump(model, 'newClass.pkl', compress=('zlib', 3))
```

Appendix E Consent Form



DocuSign Envelope ID: 3B74982B-55F6-4045-AF8D-7AB257084D5D

Parental Consent Form

Smart Toy for Children with Autism

Your child is being asked to participate in a voluntary research study. The purpose of this study is to determine the functionality of the smart toy. Data collected from the study will be used in future educational versions to determine accuracy in its ability to target the best type of questions to ask a child. Participants will be asked to play with the toy to determine if the toy can teach them color sequences and simple addition. We will also ask a few follow-up questions and your child's participation will last a minimum of 20 min or until they tire of the toy. Risks related to this research include potential frustration with the toy if it does not work as expected. Benefits related to this research include helping a child learn and transfer information from one subject to another.

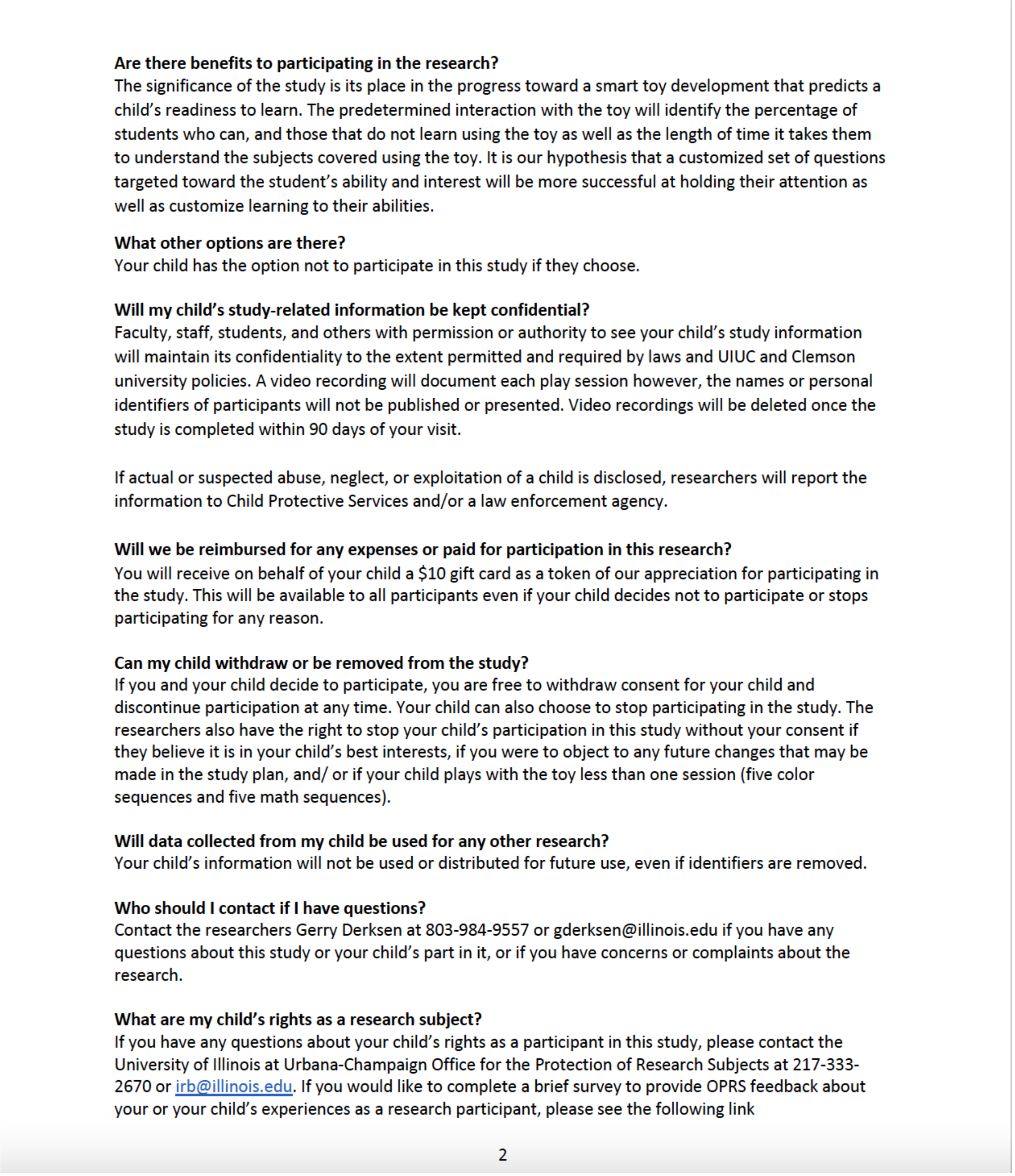
Principal Investigator Name and Title: Dr. Stan Ruecker
Department and Institution: Art + Design
Contact Information: 422 Flagg Hall, UIUC campus, sruecker@illinois.edu
Sponsor: Campus Research Board

Why is your child being asked?
Your child is being asked to be a participant in a research study about an educational toy. The purpose of this research is to determine the performance of a typical 'tree structured' program built into the toy. Your child has been asked to participate in this research because they best fit the audience, we are interested in children ages 6-10 and diagnosed with autism. Approximately sixteen participants will be involved in this research at the University of Illinois at Urbana-Champaign.

Your child's participation in this research is voluntary. You and your child's decision whether or not to participate will not affect your or your child's current or future dealings with the University of Illinois at Urbana-Champaign. If you decide your child can participate, you or your child are free to withdraw at any time without affecting that relationship.

What procedures are involved?
This research will be performed at A Place for Hope campus in Greenville SC. The study will be part of your child's everyday interactions with staff at the center and each session will last approximately 20 min – 40 min. To begin your child will be assessed for their current understanding of color and some basic math in a pre-test. During the study the child will be asked to play with the toy for as long as they like. Those who play with the toy less than one sequence (five color sequences and five math sequences) will not be included in the results. All other participants will be asked follow-up questions once they decide to stop playing or once their lesson time is complete. Similar questions will be asked as the pre-test. Your child can play as many times as they wish within guidelines at the center allowing other children to play.

What are the potential risks and discomforts?
There are no risks to your child.



Are there benefits to participating in the research?

The significance of the study is its place in the progress toward a smart toy development that predicts a child's readiness to learn. The predetermined interaction with the toy will identify the percentage of students who can, and those that do not learn using the toy as well as the length of time it takes them to understand the subjects covered using the toy. It is our hypothesis that a customized set of questions targeted toward the student's ability and interest will be more successful at holding their attention as well as customize learning to their abilities.

What other options are there?

Your child has the option not to participate in this study if they choose.

Will my child's study-related information be kept confidential?

Faculty, staff, students, and others with permission or authority to see your child's study information will maintain its confidentiality to the extent permitted and required by laws and UIUC and Clemson university policies. A video recording will document each play session however, the names or personal identifiers of participants will not be published or presented. Video recordings will be deleted once the study is completed within 90 days of your visit.

If actual or suspected abuse, neglect, or exploitation of a child is disclosed, researchers will report the information to Child Protective Services and/or a law enforcement agency.

Will we be reimbursed for any expenses or paid for participation in this research?

You will receive on behalf of your child a $10 gift card as a token of our appreciation for participating in the study. This will be available to all participants even if your child decides not to participate or stops participating for any reason.

Can my child withdraw or be removed from the study?

If you and your child decide to participate, you are free to withdraw consent for your child and discontinue participation at any time. Your child can also choose to stop participating in the study. The researchers also have the right to stop your child's participation in this study without your consent if they believe it is in your child's best interests, if you were to object to any future changes that may be made in the study plan, and/ or if your child plays with the toy less than one session (five color sequences and five math sequences).

Will data collected from my child be used for any other research?

Your child's information will not be used or distributed for future use, even if identifiers are removed.

Who should I contact if I have questions?

Contact the researchers Gerry Derksen at 803-984-9557 or gderksen@illinois.edu if you have any questions about this study or your child's part in it, or if you have concerns or complaints about the research.

What are my child's rights as a research subject?

If you have any questions about your child's rights as a participant in this study, please contact the University of Illinois at Urbana-Champaign Office for the Protection of Research Subjects at 217-333-2670 or irb@illinois.edu. If you would like to complete a brief survey to provide OPRS feedback about your or your child's experiences as a research participant, please see the following link

Appendix F IRB Approval Study 1

UNIVERSITY OF ILLINOIS
URBANA-CHAMPAIGN

Office of the Vice Chancellor for Research & Innovation

Office for the Protection of Research Subjects
805 W. Pennsylvania Ave., MC-095
Urbana, IL 61801-4822

[redacted]

August 10, 2022

Notice of Exempt Determination

Principal Investigator	Stanley Ruecker
CC	Gary Derksen
Protocol Title	*Smart Toy Usability Test*
Protocol Number	23191
Funding Source	Campus Research Board
Review Category	Exempt 3(i)(B)
Risk Determination	No more than minimal risk
Approval Date	August 10, 2022
Expiration Date	August 9, 2027

New Protocol, Consent has been stamped

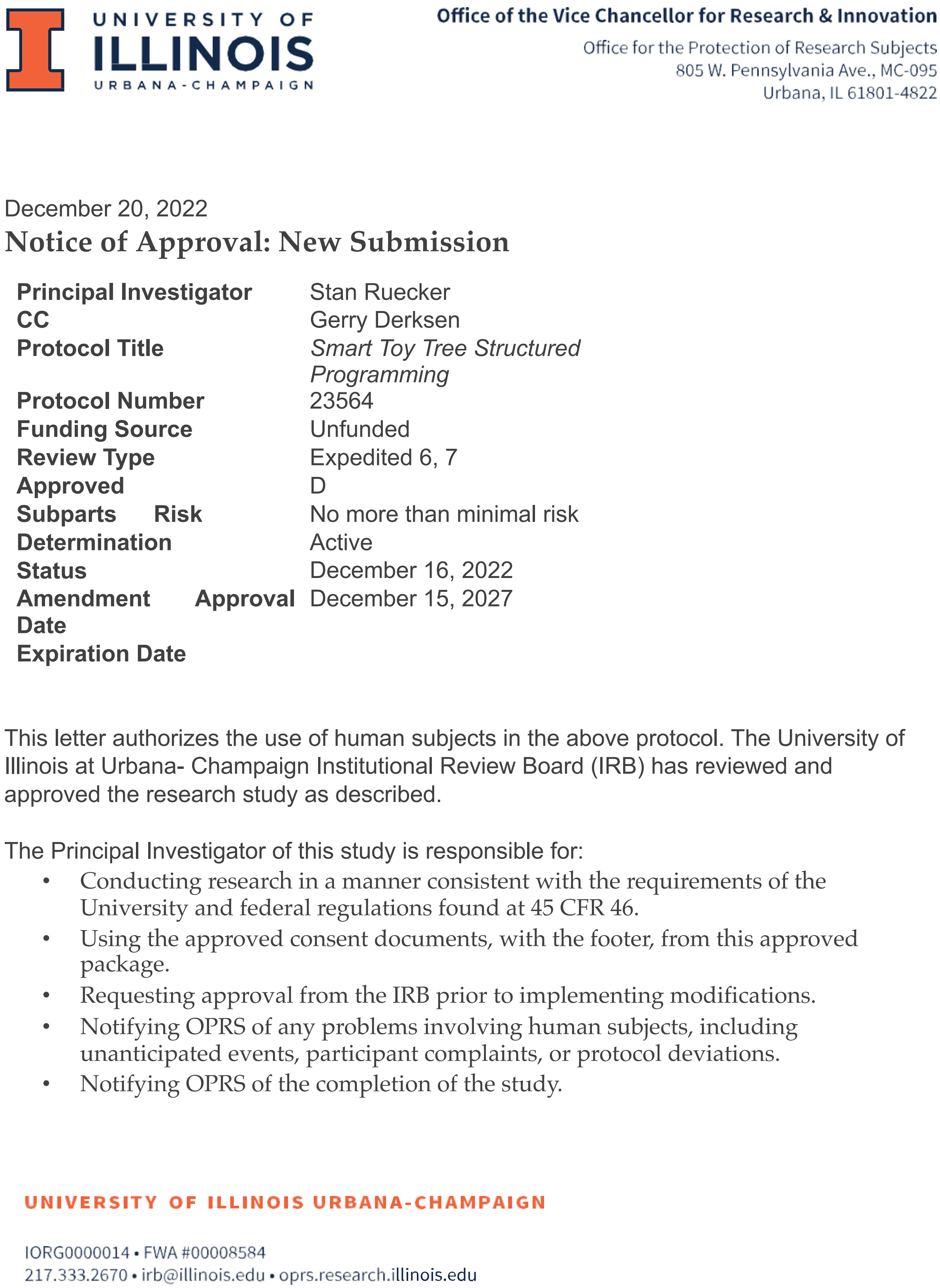
This letter authorizes the use of human subjOects in the above protocol. The University of Illinois at Urbana-Champaign Office for the Protection of Research Subjects (OPRS) has reviewed your application and determined the criteria for exemption have been met.

The Principal Investigator of this study is responsible for:

- Conducting research in a manner consistent with the requirements of the University and federal regulations found at 45 CFR 46.
- Requesting approval from the IRB prior to implementing major modifications.
- Notifying OPRS of any problems involving human subjects, including unanticipated events, participant complaints, or protocol deviations.
- Notifying OPRS of the completion of the study.

Changes to an **exempt** protocol are only required if substantive modifications are requested and/or the changes requested may affect the exempt status.

Appendix G IRB Approval and Amendment Study 2 and 3



UNIVERSITY OF
ILLINOIS
URBANA-CHAMPAIGN

Office of the Vice Chancellor for Research & Innovation

Office for the Protection of Research Subjects
805 W. Pennsylvania Ave., MC-095
Urbana, IL 61801-4822

December 20, 2022

Notice of Approval: New Submission

Principal Investigator	Stan Ruecker
CC	Gerry Derksen
Protocol Title	*Smart Toy Tree Structured Programming*
Protocol Number	23564
Funding Source	Unfunded
Review Type	Expedited 6, 7
Approved	D
Subparts Risk	No more than minimal risk
Determination	Active
Status	December 16, 2022
Amendment Approval Date	December 15, 2027
Expiration Date	

This letter authorizes the use of human subjects in the above protocol. The University of Illinois at Urbana- Champaign Institutional Review Board (IRB) has reviewed and approved the research study as described.

The Principal Investigator of this study is responsible for:

- Conducting research in a manner consistent with the requirements of the University and federal regulations found at 45 CFR 46.
- Using the approved consent documents, with the footer, from this approved package.
- Requesting approval from the IRB prior to implementing modifications.
- Notifying OPRS of any problems involving human subjects, including unanticipated events, participant complaints, or protocol deviations.
- Notifying OPRS of the completion of the study.

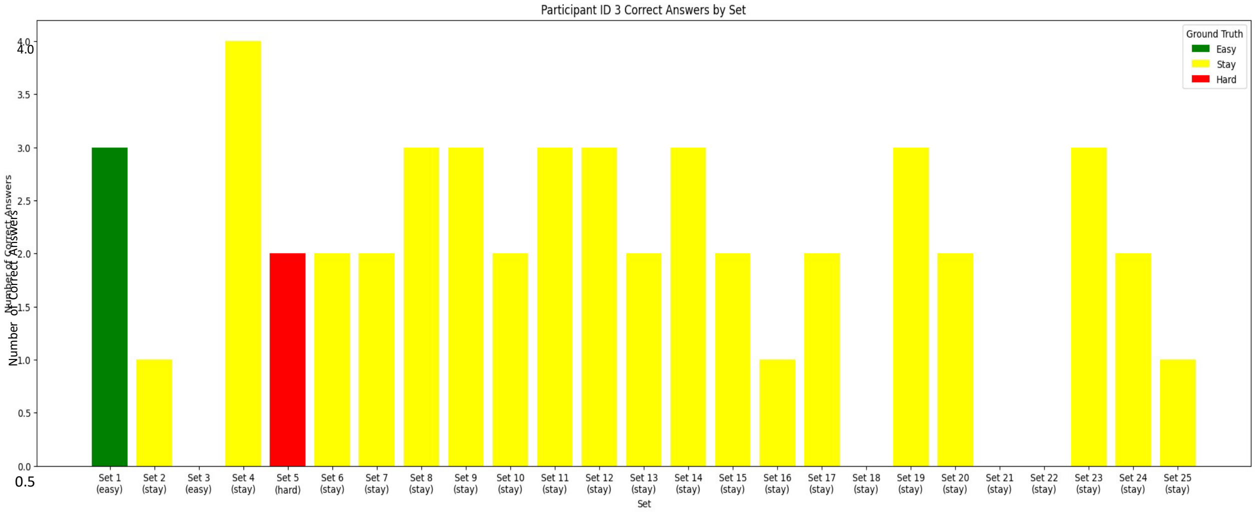
UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

IORG0000014 • FWA #00008584
217.333.2670 • irb@illinois.edu • oprs.research.illinois.edu

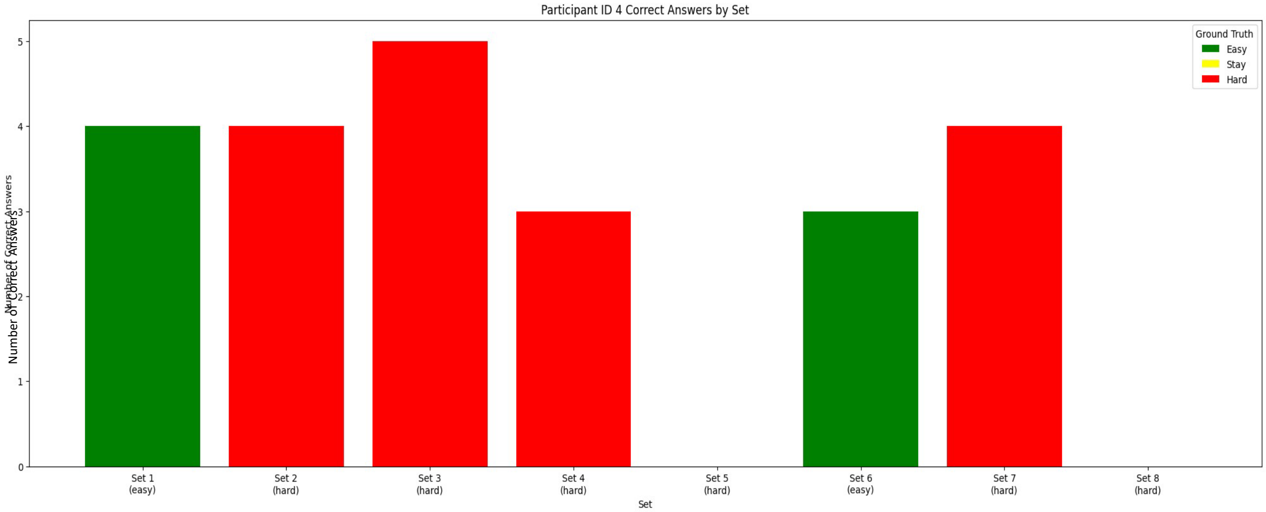
Appendix H Bar Plot Prediction – Performance TS

```
Participant ID 3:
Total number of correct answers: 49
Total number of sets: 25
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 3, 5, 5, 5, 5,
5, 5, 2]
Number of correct answers in each set: [3, 1, 0, 4, 2, 2, 2, 3, 3, 2, 3, 3, 2, 3, 2, 1, 2, 0, 3, 2, 0, 0,
3, 2, 1]
Target model values for each set: ['easy', 'stay', 'easy', 'stay', 'hard', 'stay', 'stay', 'stay', 'stay',
'stay', 'stay', 'stay', 'stay'
Participant ID 4:
Total number of correct answers: 23
Total number of sets: 8
Total number of questions in each set: [5, 5, 5, 4, 2, 4, 5, 1]
Number of correct answers in each set: [4, 4, 5, 3, 0, 3, 4, 0]
Target model values for each set: ['easy', 'hard', 'hard', 'hard', 'hard', 'easy', 'hard', 'hard']
Participant ID 5:
Total number of correct answers: 72
Total number of sets: 25
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 4, 5, 5, 5,
5, 5, 1]
Number of correct answers in each set: [1, 2, 3, 3, 2, 4, 4, 2, 4, 3, 2, 3, 3, 1, 5, 5, 2, 5, 4, 3, 3, 3,
2, 3, 0]
Target model values for each set: ['easy', 'stay', 'stay', 'stay', 'stay', 'stay', 'hard', 'hard', 'stay',
'hard', 'hard', 'stay', 'stay'
Participant ID 6:
Total number of correct answers: 26
Total number of sets: 10
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 1, 1, 1]
Number of correct answers in each set: [3, 2, 5, 3, 3, 5, 4, 1, 0, 0]
Target model values for each set: ['easy', 'stay', 'stay', 'hard', 'stay', 'stay', 'hard', 'hard', 'hard',
'easy']
Participant ID 7:
Total number of correct answers: 20
Total number of sets: 13
Total number of questions in each set: [5, 6, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 1]
Number of correct answers in each set: [3, 3, 1, 0, 1, 0, 3, 3, 1, 2, 3, 0, 0]
Target model values for each set: ['easy', 'stay', 'easy', 'easy', 'easy', 'easy', 'easy', 'stay', 'stay',
'stay', 'stay', 'stay', 'stay'
Participant ID 8:
Total number of correct answers: 31
Total number of sets: 18
Total number of questions in each set: [5, 6, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 3]
Number of correct answers in each set: [2, 2, 2, 2, 3, 0, 1, 3, 2, 2, 4, 2, 1, 2, 1, 2, 0, 0]
Target model values for each set: ['easy', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay',
'stay', 'stay', 'hard', 'easy'
Participant ID 9:
Total number of correct answers: 30
Total number of sets: 11
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 1]
Number of correct answers in each set: [2, 2, 4, 3, 3, 4, 2, 5, 2, 3, 0]
Target model values for each set: ['easy', 'stay', 'stay', 'hard', 'stay', 'stay', 'hard', 'easy', 'hard',
'easy', 'stay']
Participant ID 10:
Total number of correct answers: 9
Total number of sets: 5
Total number of questions in each set: [5, 2, 5, 5, 2]
Number of correct answers in each set: [3, 0, 2, 4, 0]
Target model values for each set: ['easy', 'stay', 'easy', 'stay', 'hard']
Participant ID 11:
Total number of correct answers: 15
Total number of sets: 7
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 10]
Number of correct answers in each set: [3, 3, 3, 1, 1, 1, 3]
Target model values for each set: ['easy', 'hard', 'stay', 'stay', 'easy', 'easy', 'easy']
Participant ID 12:
Total number of correct answers: 29
Total number of sets: 12
Total number of questions in each set: [4, 5, 5, 5, 5, 5, 4, 5, 5, 5, 5, 3]
Number of correct answers in each set: [2, 3, 3, 2, 0, 1, 4, 4, 4, 4, 2, 0]
Target model values for each set: ['easy', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay', 'hard', 'hard',
'hard', 'hard', 'stay']
Participant ID 13:
Total number of correct answers: 7
Total number of sets: 4
Total number of questions in each set: [5, 5, 5, 4]
Number of correct answers in each set: [3, 2, 2, 0]
Target model values for each set: ['easy', 'stay', 'easy', 'easy']
Participant ID 14:
Total number of correct answers: 18
Total number of sets: 9
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 5, 1]
Number of correct answers in each set: [4, 3, 4, 2, 2, 3, 0, 0, 0]
Target model values for each set: ['easy', 'hard', 'stay', 'hard', 'stay', 'easy', 'stay', 'easy', 'easy']
Participant ID 15:
Total number of correct answers: 34
Total number of sets: 12
Total number of questions in each set: [4, 5, 5, 5, 3, 5, 5, 5, 5, 5, 5, 5]
Number of correct answers in each set: [3, 5, 5, 3, 2, 3, 4, 3, 1, 1, 1, 3]
Target model values for each set: ['easy', 'stay', 'hard', 'hard', 'stay', 'stay', 'stay', 'hard', 'stay',
'easy', 'easy', 'easy']
Participant ID 16:
Total number of correct answers: 8
Total number of sets: 5
Total number of questions in each set: [5, 5, 5, 5, 2]
Number of correct answers in each set: [1, 3, 1, 3, 0]
Target model values for each set: ['easy', 'easy', 'stay', 'stay', 'stay']
Participant ID 17:
Total number of correct answers: 44
Total number of sets: 14
```

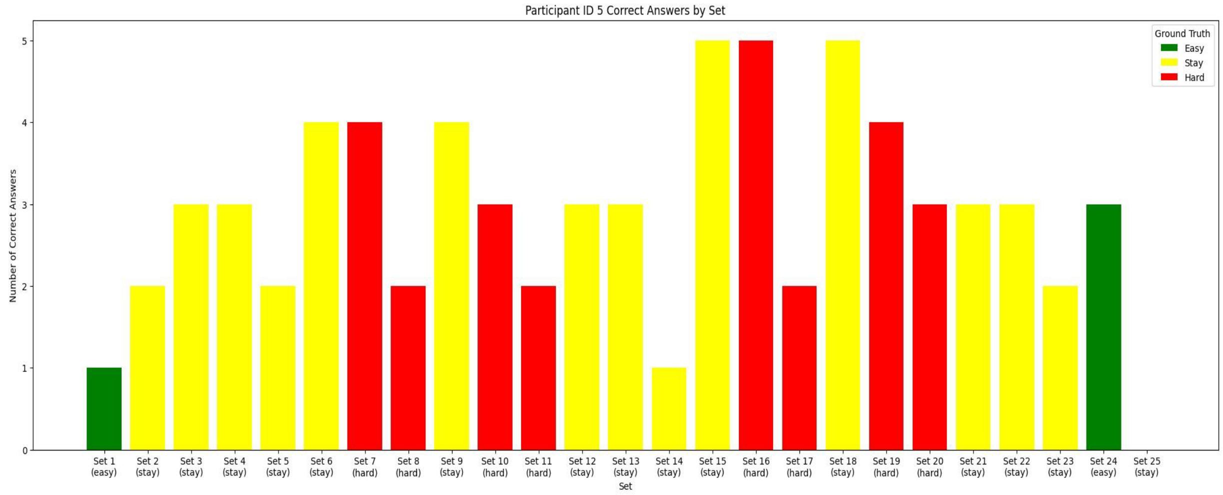
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 1, 5, 5, 5, 5, 5, 5, 5]
Number of correct answers in each set: [4, 4, 4, 4, 4, 5, 1, 3, 4, 3, 1, 4, 0, 3]
Target model values for each set: ['easy', 'hard', 'hard', 'hard', 'hard', 'hard', 'hard', 'hard', 'stay', 'hard', 'stay', 'easy', 'hard'
Participant ID 18:
Total number of correct answers: 81
Total number of sets: 38
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5
Number of correct answers in each set: [3, 0, 3, 4, 5, 3, 1, 1, 3, 3, 4, 5, 2, 2, 2, 0, 1, 2, 0, 1, 1, 0, 0, 1, 1, 2, 2, 3, 5, 5, 4, 2, 4 Target model values for each set: ['easy', 'stay', 'easy', 'stay', 'hard', 'hard', 'stay', 'easy', 'easy', 'stay', 'stay', 'hard', 'hard'



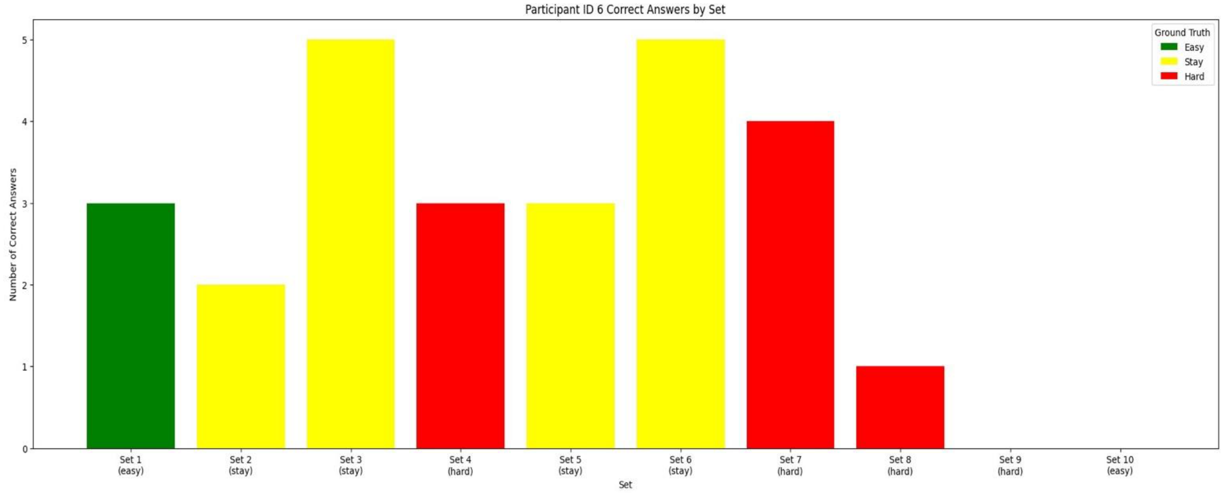
3 3



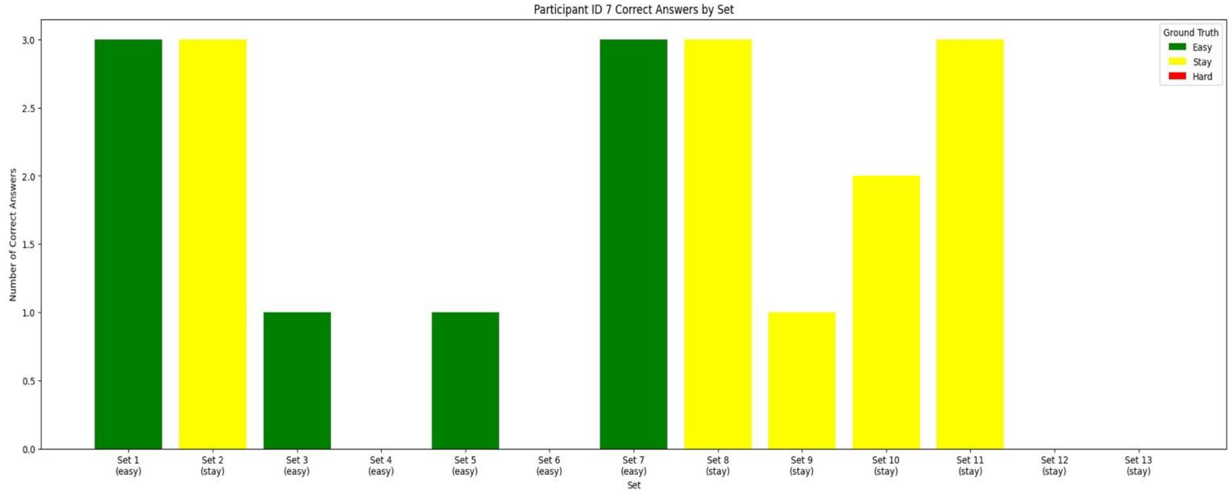
4 4


Participant ID 5 Correct Answers by Set
Number of Correct Answers
0
1
2
3
4
5
Set
Set 1 (easy)
Set 2 (stay)
Set 3 (stay)
Set 4 (stay)
Set 5 (stay)
Set 6 (stay)
Set 7 (hard)
Set 8 (hard)
Set 9 (stay)
Set 10 (hard)
Set 11 (hard)
Set 12 (stay)
Set 13 (stay)
Set 14 (stay)
Set 15 (stay)
Set 16 (hard)
Set 17 (hard)
Set 18 (stay)
Set 19 (hard)
Set 20 (hard)
Set 21 (stay)
Set 22 (stay)
Set 23 (stay)
Set 24 (easy)
Set 25 (stay)
Ground Truth
Easy
Stay
Hard

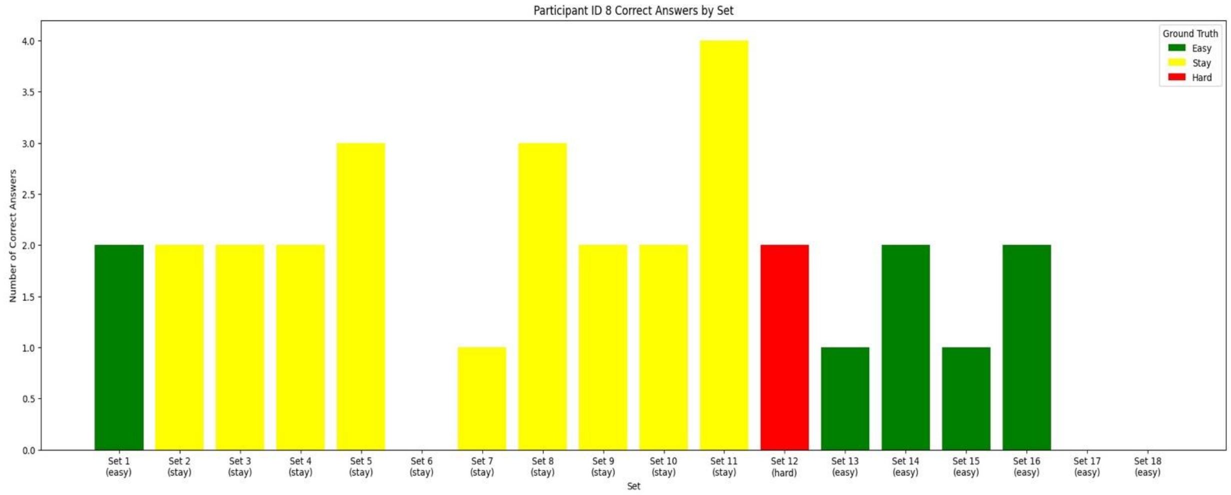
5 5


Participant ID 6 Correct Answers by Set
Number of Correct Answers
0
1
2
3
4
5
Set
Set 1 (easy)
Set 2 (stay)
Set 3 (stay)
Set 4 (hard)
Set 5 (stay)
Set 6 (stay)
Set 7 (hard)
Set 8 (hard)
Set 9 (hard)
Set 10 (easy)
Ground Truth
Easy
Stay
Hard

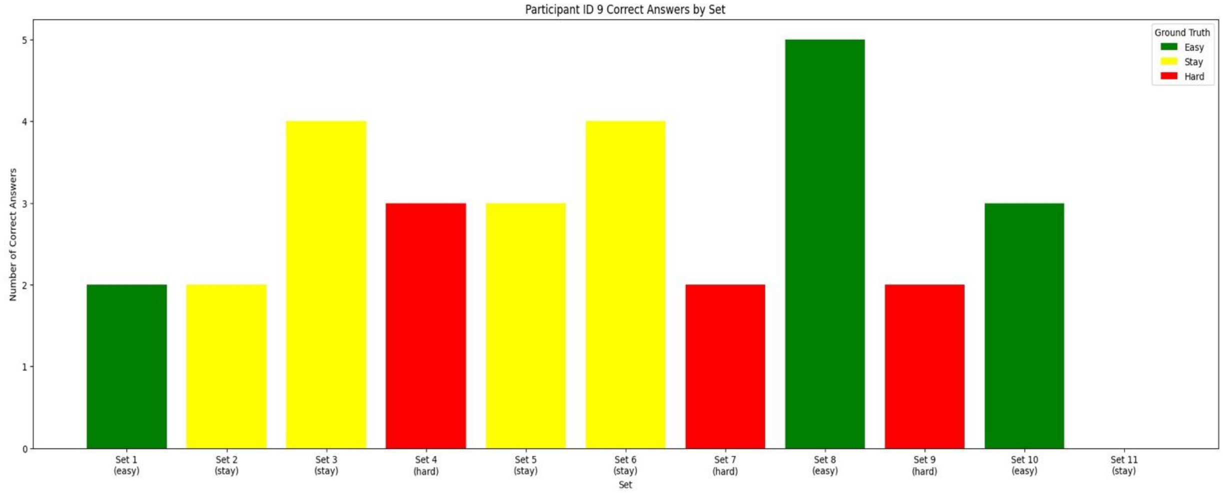
6 6


Participant ID 7 Correct Answers by Set
Number of Correct Answers
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Set
Set 1 (easy)
Set 2 (stay)
Set 3 (easy)
Set 4 (easy)
Set 5 (easy)
Set 6 (easy)
Set 7 (easy)
Set 8 (stay)
Set 9 (stay)
Set 10 (stay)
Set 11 (stay)
Set 12 (stay)
Set 13 (stay)
Ground Truth
Easy
Stay
Hard

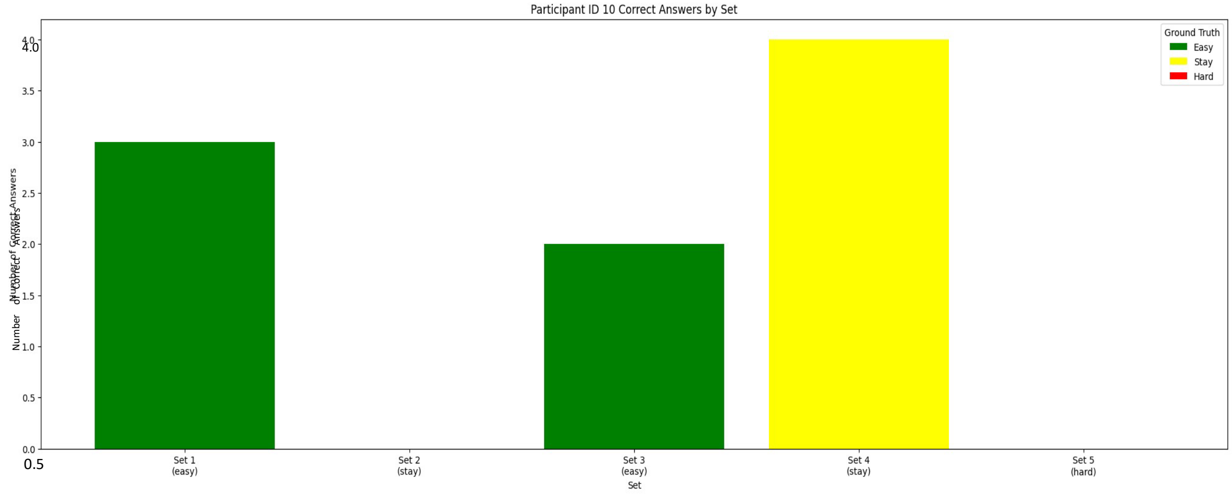
7 7


Participant ID 8 Correct Answers by Set
Number of Correct Answers
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Set
Set 1 (easy)
Set 2 (stay)
Set 3 (stay)
Set 4 (stay)
Set 5 (stay)
Set 6 (stay)
Set 7 (stay)
Set 8 (stay)
Set 9 (stay)
Set 10 (stay)
Set 11 (stay)
Set 12 (hard)
Set 13 (easy)
Set 14 (easy)
Set 15 (easy)
Set 16 (easy)
Set 17 (easy)
Set 18 (easy)
Ground Truth
Easy
Stay
Hard

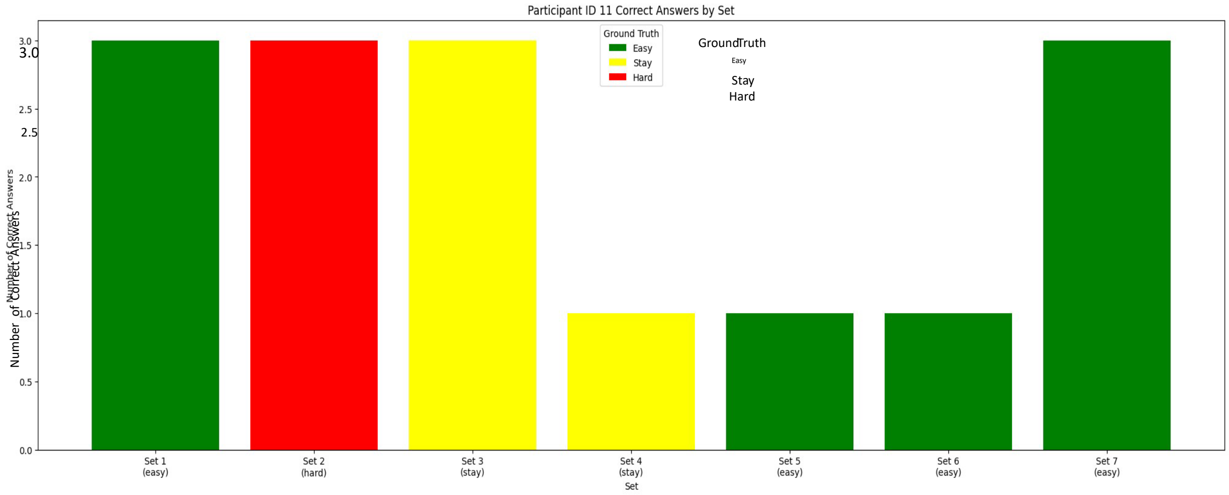
8 8


Participant ID 9 Correct Answers by Set
Number of Correct Answers
0
1
2
3
4
5
Set
Set 1 (easy)
Set 2 (stay)
Set 3 (stay)
Set 4 (hard)
Set 5 (stay)
Set 6 (stay)
Set 7 (hard)
Set 8 (easy)
Set 9 (hard)
Set 10 (easy)
Set 11 (stay)
Ground Truth
Easy
Stay
Hard

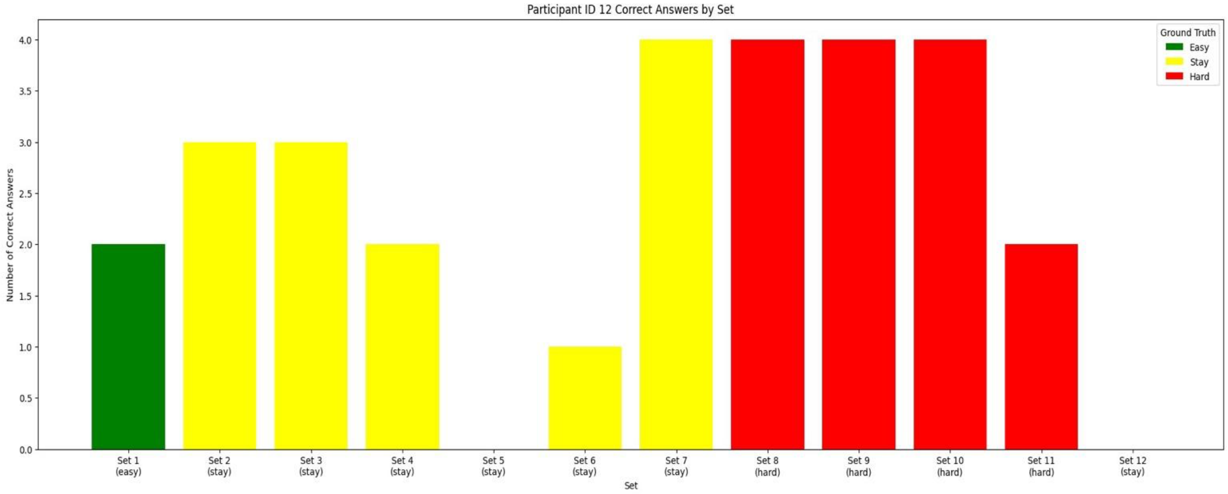
9 9


Participant ID 10 Correct Answers by Set
Number of Correct Answers
Number of Correct Answers
0.0
0.5
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.0
Set
Set 1 (easy)
Set 2 (stay)
Set 3 (easy)
Set 4 (stay)
Set 5 (hard)
Ground Truth
Easy
Stay
Hard

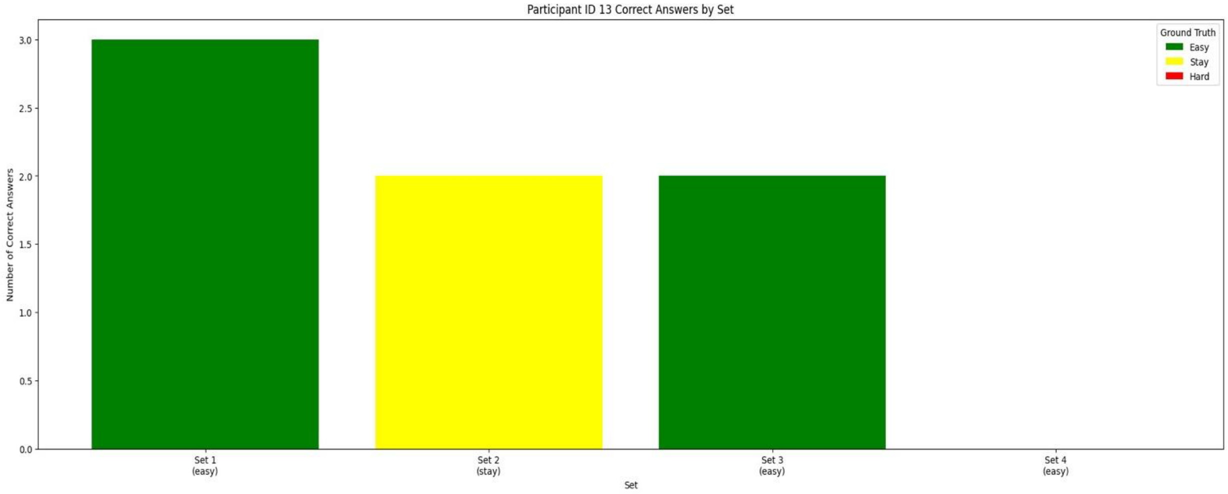
10 10



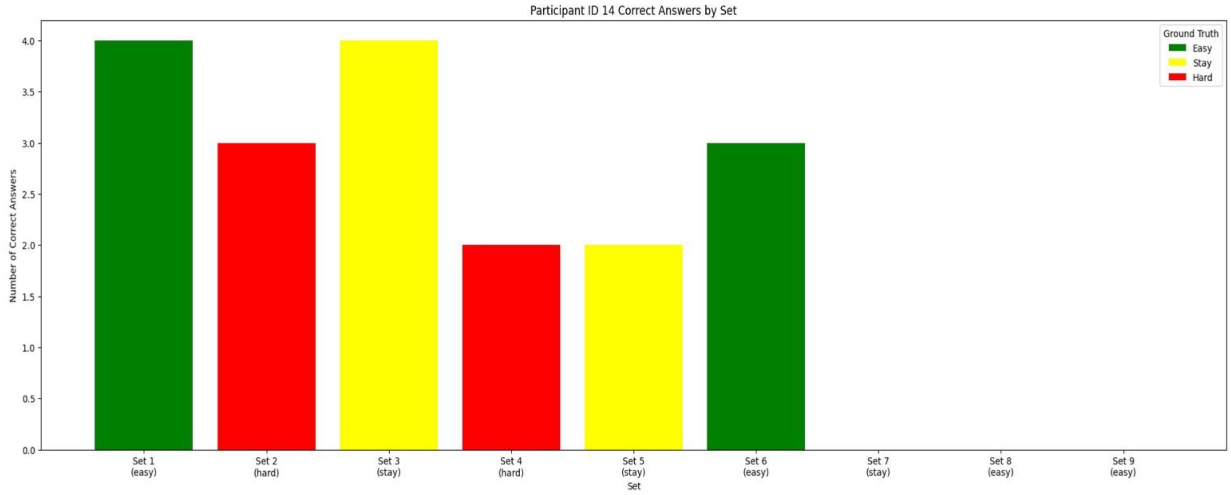
11 11



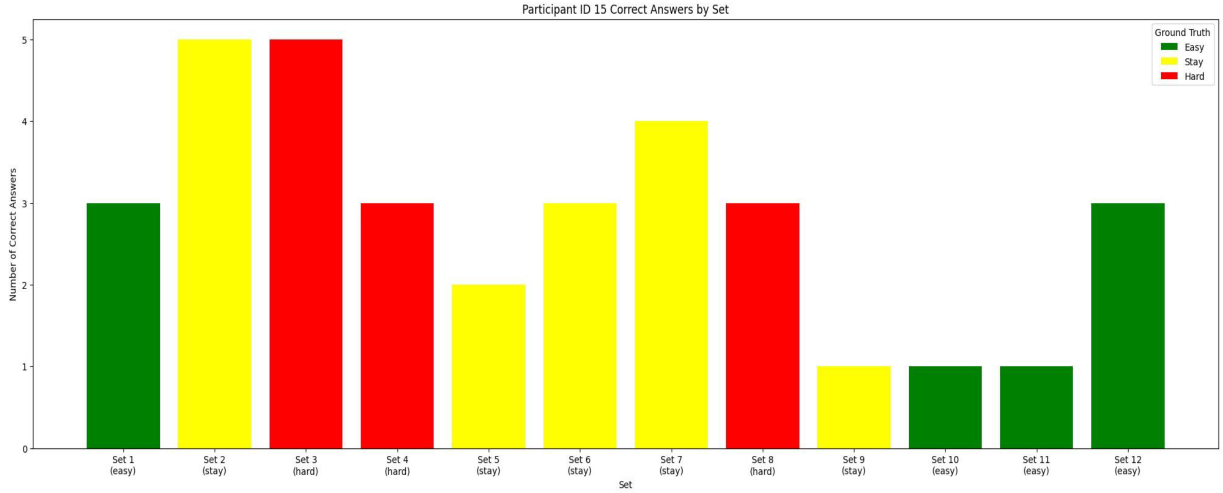
12 12



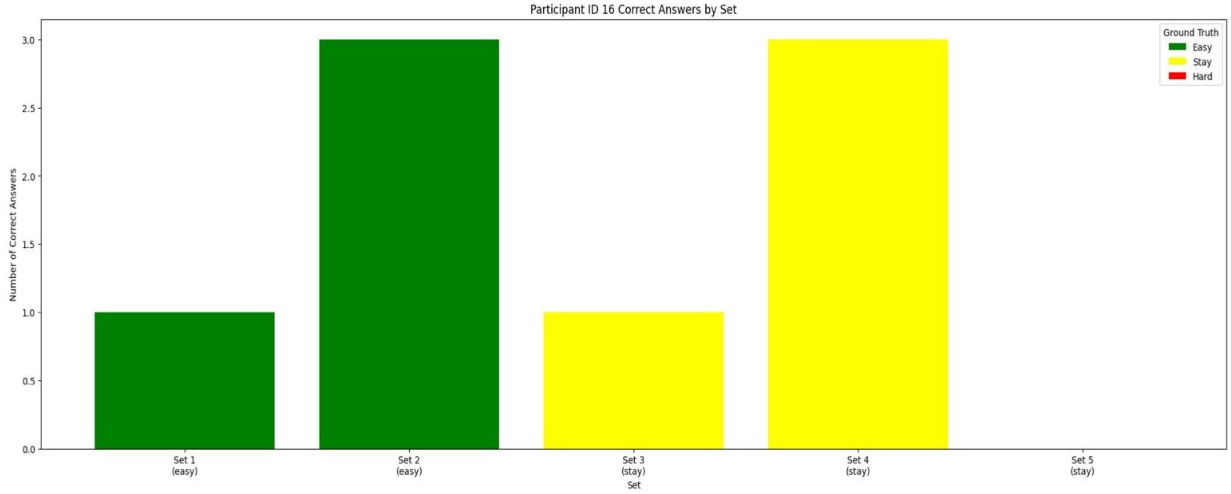
13 13



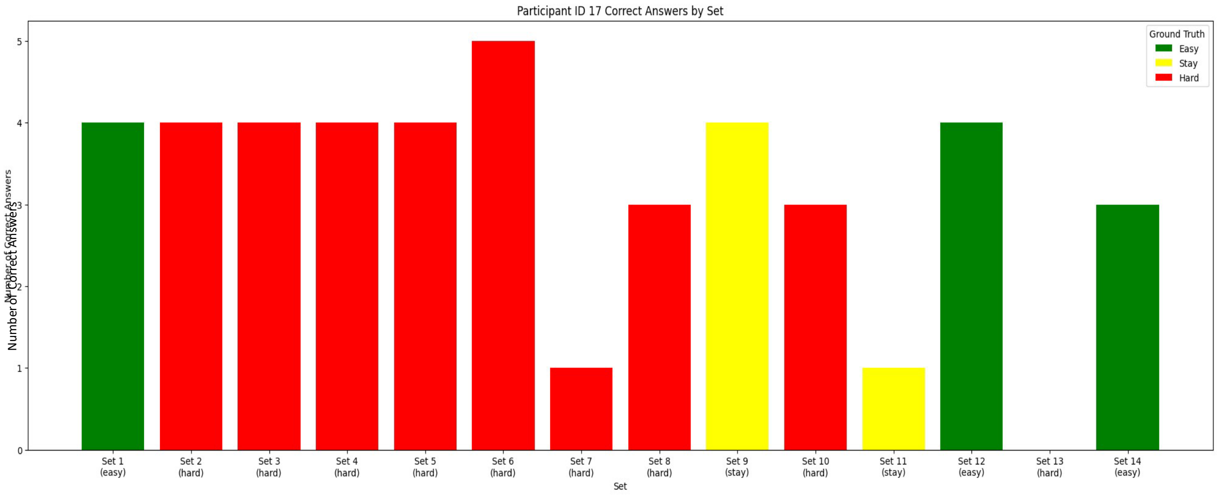
14 14



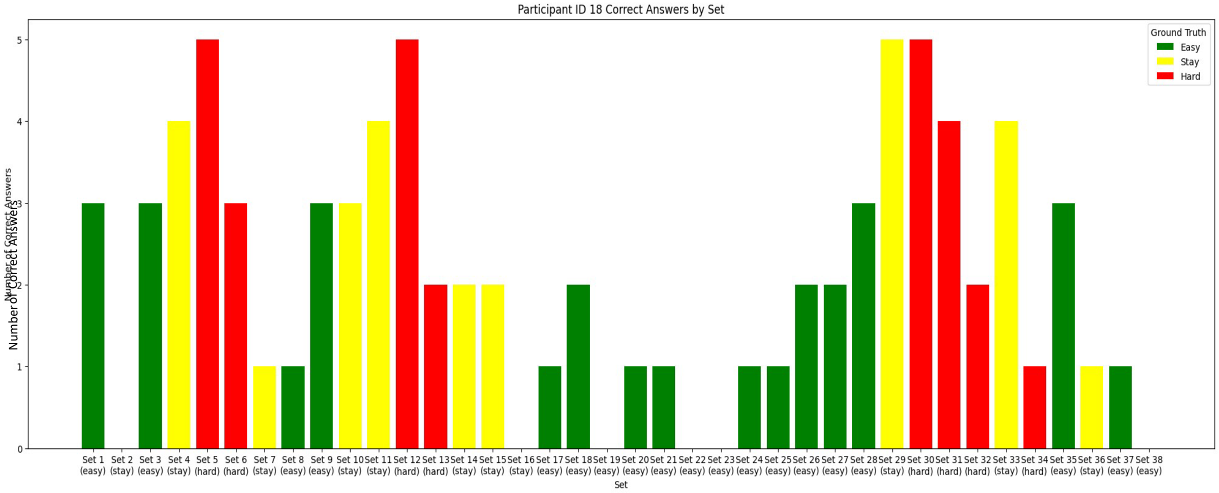
15 15



16 16


Participant ID 17 Correct Answers by Set
Ground Truth
Easy
Stay
Hard
Number of Correct Answers
Number of Correct Answers
0
1
2
3
4
5
Set 1 (easy)
Set 2 (hard)
Set 3 (hard)
Set 4 (hard)
Set 5 (hard)
Set 6 (hard)
Set 7 (hard)
Set 8 (hard)
Set 9 (stay)
Set 10 (hard)
Set 11 (stay)
Set 12 (easy)
Set 13 (hard)
Set 14 (easy)
Set

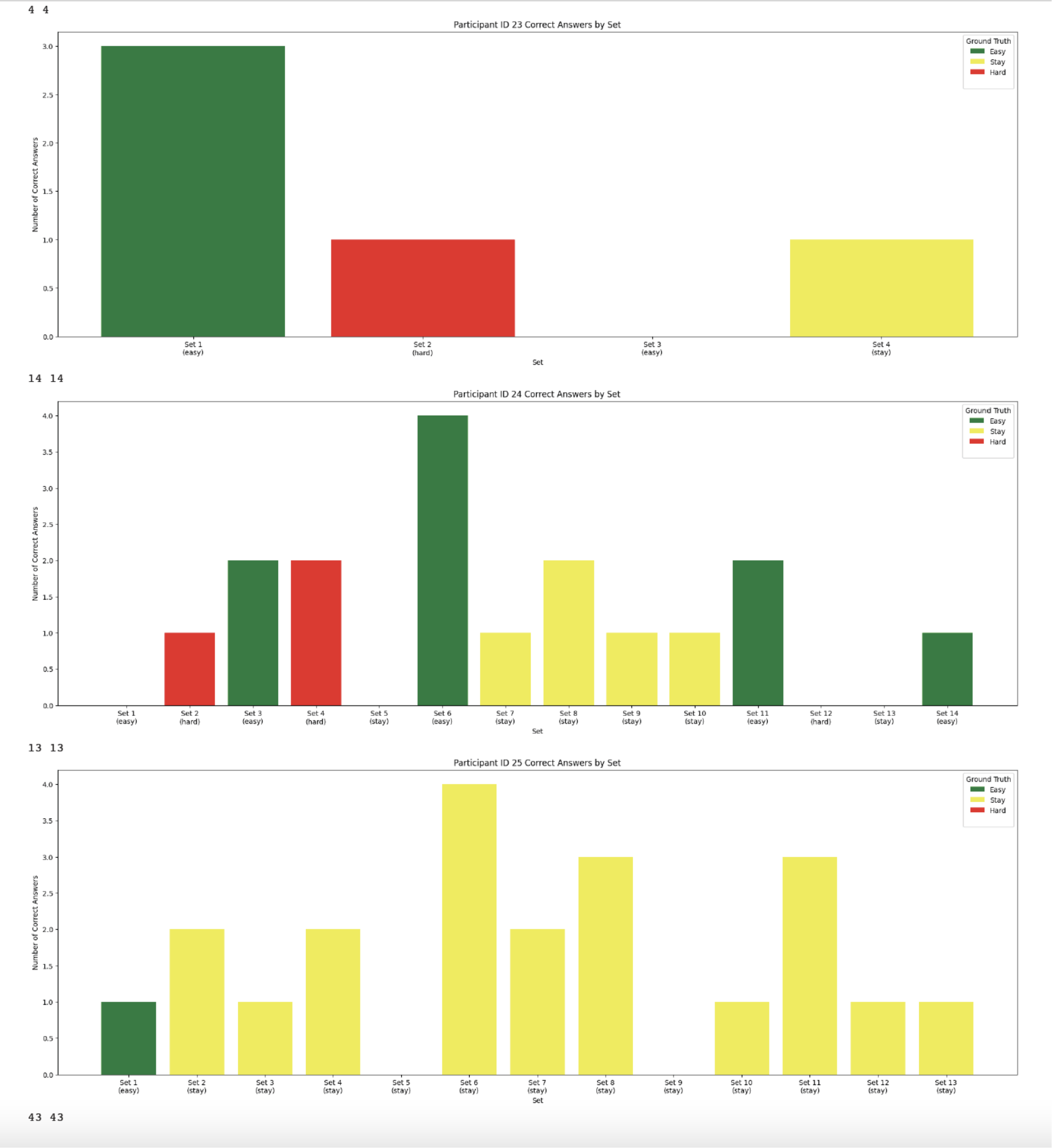
17 17


Participant ID 18 Correct Answers by Set
Ground Truth
Easy
Stay
Hard
Number of Correct Answers
Number of Correct Answers
0
1
2
3
4
5
Set 1 (easy)
Set 2 (stay)
Set 3 (easy)
Set 4 (stay)
Set 5 (hard)
Set 6 (hard)
Set 7 (stay)
Set 8 (easy)
Set 9 (easy)
Set 10 (stay)
Set 11 (stay)
Set 12 (hard)
Set 13 (hard)
Set 14 (stay)
Set 15 (stay)
Set 16 (stay)
Set 17 (easy)
Set 18 (easy)
Set 19 (easy)
Set 20 (easy)
Set 21 (easy)
Set 22 (easy)
Set 23 (easy)
Set 24 (easy)
Set 25 (easy)
Set 26 (easy)
Set 27 (easy)
Set 28 (easy)
Set 29 (stay)
Set 30 (hard)
Set 31 (hard)
Set 32 (hard)
Set 33 (stay)
Set 34 (hard)
Set 35 (easy)
Set 36 (stay)
Set 37 (easy)
Set 38 (easy)
Set

18 18

Appendix I Bar Plot Prediction – Performance ML

```
Participant ID 23:
Total number of correct answers: 5
Total number of sets: 4
Total number of questions in each set: [5, 2, 5, 1]
Number of correct answers in each set: [3, 1, 0, 1]
Target model values for each set: ['easy', 'hard', 'easy', 'stay']
Participant ID 24:
Total number of correct answers: 17
Total number of sets: 14
Total number of questions in each set: [4, 3, 4, 5, 1, 5, 5, 5, 5, 5, 5, 5, 2, 4]
Number of correct answers in each set: [0, 1, 2, 2, 0, 4, 1, 2, 1, 1, 2, 0, 0, 1]
Target model values for each set: ['easy', 'hard', 'easy', 'hard', 'stay', 'easy', 'stay', 'stay', 'stay',
'stay', 'easy', 'hard', 'stay', 'easy']
Participant ID 25:
Total number of correct answers: 21
Total number of sets: 13
Total number of questions in each set: [1, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 2]
Number of correct answers in each set: [1, 2, 1, 2, 0, 4, 2, 3, 0, 1, 3, 1, 1]
Target model values for each set: ['easy', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay',
'stay', 'stay', 'stay', 'stay']
Participant ID 26:
Total number of correct answers: 110
Total number of sets: 43
Total number of questions in each set: [5, 4, 2, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5, 5, 5,
5, 5, 5, 5, 5, 5, 1, 5, 5, 5, 5, 5, Number of correct answers in each set: [2, 1, 0, 5, 0, 0, 0, 4, 4, 3,
3, 3, 4, 3, 3, 2, 3, 0, 3, 3, 3, 2, 3, 1, 4, 5, 0, 2, 1, 3, 3, 4, 1, 3, Target model values for each set:
['easy', 'hard', 'stay', 'easy', 'hard', 'stay', 'easy', 'stay', 'stay', 'stay', 'stay', 'stay', 'stay', 'hard',
Participant ID 27:
Total number of correct answers: 18
Total number of sets: 8
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 2]
Number of correct answers in each set: [5, 4, 2, 2, 1, 1, 2, 1]
Target model values for each set: ['easy', 'hard', 'hard', 'hard', 'easy', 'easy', 'easy', 'stay']
Participant ID 28:
Total number of correct answers: 16
Total number of sets: 6
Total number of questions in each set: [5, 5, 5, 5, 5, 1]
Number of correct answers in each set: [5, 3, 2, 4, 2, 0]
Target model values for each set: ['easy', 'hard', 'hard', 'stay', 'hard', 'hard']
Participant ID 29:
Total number of correct answers: 45
Total number of sets: 14
Total number of questions in each set: [5, 5, 5, 5, 5, 4, 5, 5, 5, 5, 5, 5, 5, 5]
Number of correct answers in each set: [5, 3, 5, 3, 2, 2, 5, 4, 4, 3, 3, 3, 2, 1]
Target model values for each set: ['easy', 'hard', 'hard', 'hard', 'hard', 'hard', 'easy', 'hard', 'hard',
'hard', 'hard', 'hard', 'hard', 'hard']
Participant ID 30:
Total number of correct answers: 86
Total number of sets: 30
Total number of questions in each set: [1, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 4, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,
5, 5, 5, 5, 5, 5, 5, 4]
Number of correct answers in each set: [1, 3, 4, 2, 1, 3, 4, 4, 0, 4, 4, 4, 2, 3, 4, 4, 2, 2, 2, 1, 3, 5,
4, 4, 5, 3, 2, 0, 3, 3]
Target model values for each set: ['easy', 'easy', 'hard', 'hard', 'hard', 'easy', 'easy', 'stay', 'stay',
'easy', 'stay', 'stay', 'stay', 'stay', Participant ID 31:
Total number of correct answers: 21
Total number of sets: 8
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 5]
Number of correct answers in each set: [5, 5, 3, 4, 1, 2, 1, 0]
Target model values for each set: ['easy', 'hard', 'hard', 'hard', 'hard', 'hard', 'hard', 'hard']
Participant ID 32:
Total number of correct answers: 32
Total number of sets: 13
Total number of questions in each set: [2, 5, 5, 5, 5, 5, 1, 5, 5, 5, 5, 5, 1]
Number of correct answers in each set: [2, 4, 3, 3, 1, 1, 1, 2, 4, 3, 4, 3, 1]
Target model values for each set: ['easy', 'easy', 'hard', 'hard', 'stay', 'hard', 'hard', 'easy', 'stay',
'stay', 'stay', 'stay', 'stay']
Participant ID 33:
Total number of correct answers: 17
Total number of sets: 8
Total number of questions in each set: [5, 5, 5, 5, 5, 5, 5, 2]
Number of correct answers in each set: [4, 2, 3, 2, 1, 3, 2, 0]
Target model values for each set: ['easy', 'hard', 'easy', 'hard', 'stay', 'easy', 'stay', 'stay']
Participant ID 34:
Total number of correct answers: 20
Total number of sets: 12
4/1/24, 11:37 AM
https://zh4df8t67mc-496ff2e9c6d22116-0-
colab.googleusercontent.com/outputframe.html?vrz=colab_20240328-
060159_RC01_620078024 Page 2 of 5
Total number of questions in each set: [5, 3, 5, 5, 5, 5, 5, 5, 5, 1, 5, 1]
Number of correct answers in each set: [3, 0, 1, 2, 2, 4, 3, 3, 2, 0, 0, 0]
Target model values for each set: ['easy', 'hard', 'easy', 'hard', 'easy', 'stay', 'stay', 'easy', 'hard',
'hard', 'easy', 'stay']
```

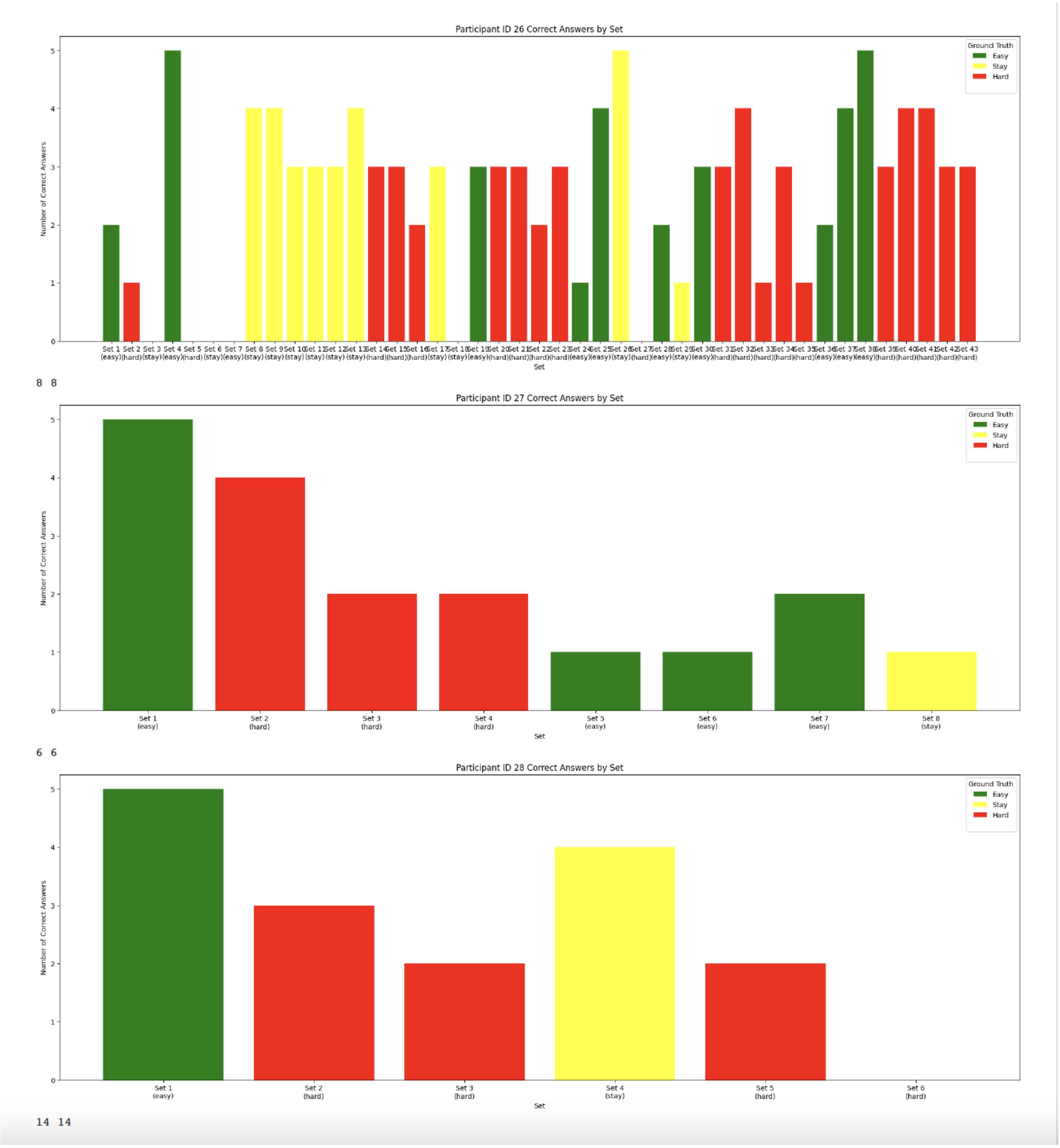


```
4 4
```

```
14 14
```

```
13 13
```

```
43 43
```

Participant ID 26 Correct Answers by Set
Number of Correct Answers
0
1
2
3
4
5
Set 1 (easy)
Set 2 (hard)
Set 3 (stay)
Set 4 (easy)
Set 5 (hard)
Set 6 (stay)
Set 7 (easy)
Set 8 (stay)
Set 9 (stay)
Set 10 (stay)
Set 11 (stay)
Set 12 (stay)
Set 13 (stay)
Set 14 (hard)
Set 15 (hard)
Set 16 (hard)
Set 17 (stay)
Set 18 (stay)
Set 19 (easy)
Set 20 (hard)
Set 21 (hard)
Set 22 (hard)
Set 23 (hard)
Set 24 (easy)
Set 25 (easy)
Set 26 (stay)
Set 27 (hard)
Set 28 (easy)
Set 29 (stay)
Set 30 (easy)
Set 31 (hard)
Set 32 (hard)
Set 33 (hard)
Set 34 (hard)
Set 35 (hard)
Set 36 (easy)
Set 37 (easy)
Set 38 (easy)
Set 39 (hard)
Set 40 (hard)
Set 41 (hard)
Set 42 (hard)
Set 43 (hard)
Set
Ground Truth
🟩 Easy
🟨 Stay
🟥 Hard

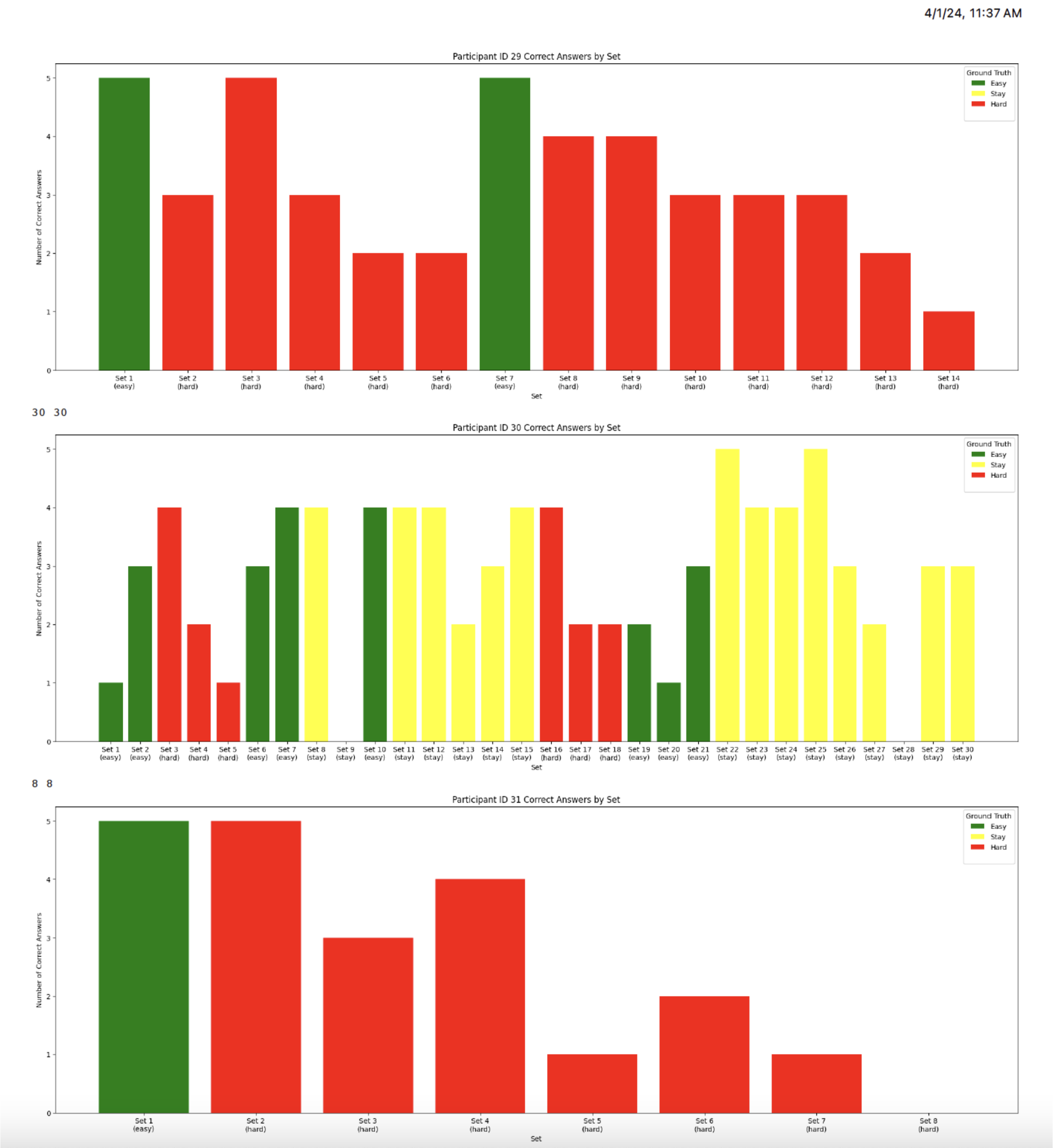
8 8

Participant ID 27 Correct Answers by Set
Number of Correct Answers
0
1
2
3
4
5
Set 1 (easy)
Set 2 (hard)
Set 3 (hard)
Set 4 (hard)
Set 5 (easy)
Set 6 (easy)
Set 7 (easy)
Set 8 (stay)
Set
Ground Truth
🟩 Easy
🟨 Stay
🟥 Hard

6 6

Participant ID 28 Correct Answers by Set
Number of Correct Answers
0
1
2
3
4
5
Set 1 (easy)
Set 2 (hard)
Set 3 (hard)
Set 4 (stay)
Set 5 (hard)
Set 6 (hard)
Set
Ground Truth
🟩 Easy
🟨 Stay
🟥 Hard

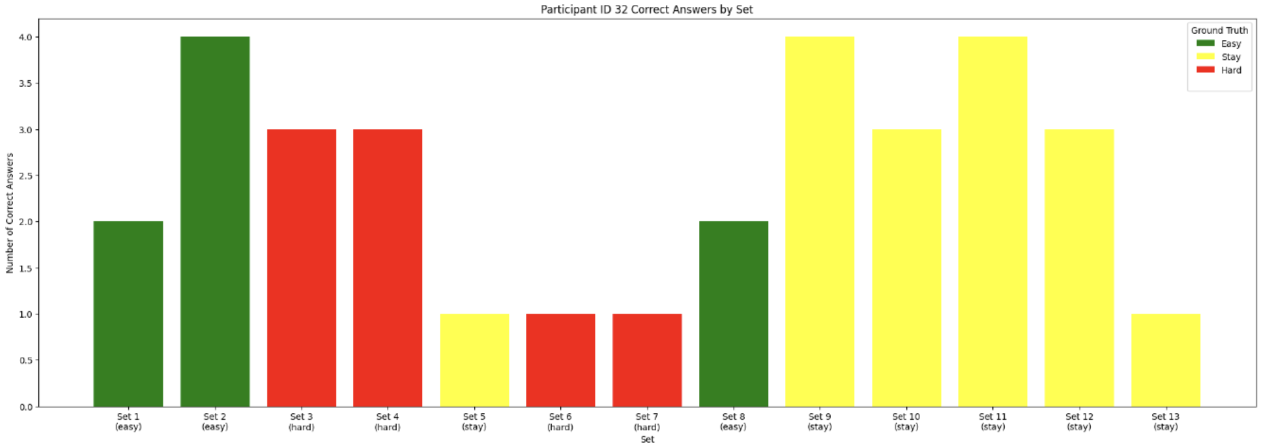
14 14



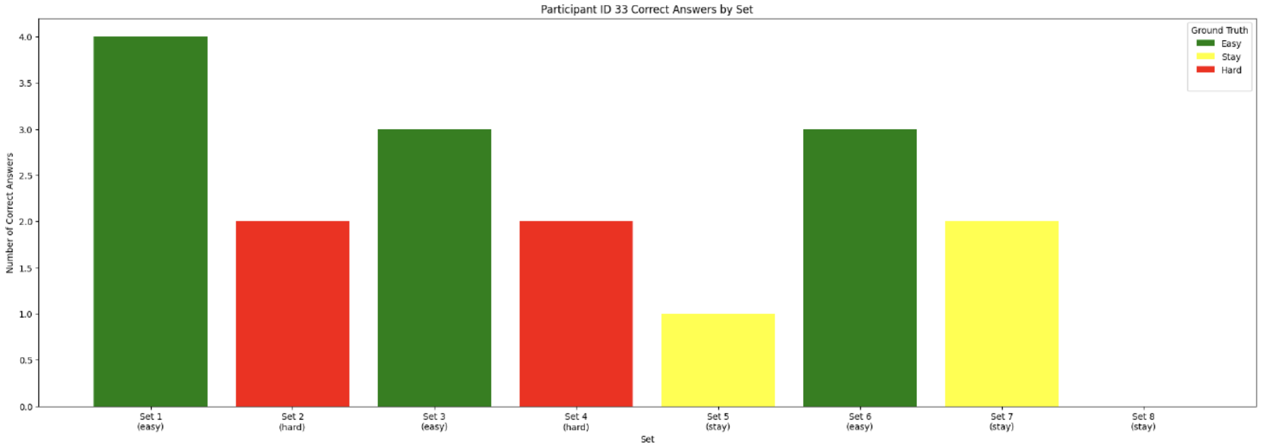
4/1/24, 11:37 AM

30 30

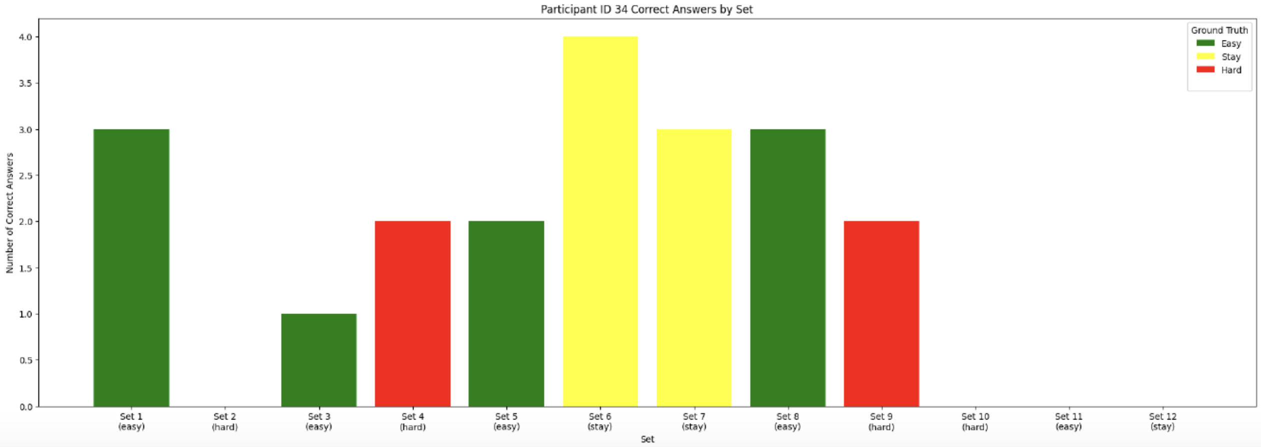
8 8


Participant ID 32 Correct Answers by Set
Number of Correct Answers
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Set 1 (easy)
Set 2 (easy)
Set 3 (hard)
Set 4 (hard)
Set 5 (stay)
Set 6 (hard)
Set 7 (hard)
Set 8 (easy)
Set 9 (stay)
Set 10 (stay)
Set 11 (stay)
Set 12 (stay)
Set 13 (stay)
Set
Ground Truth
Easy
Stay
Hard

8 8

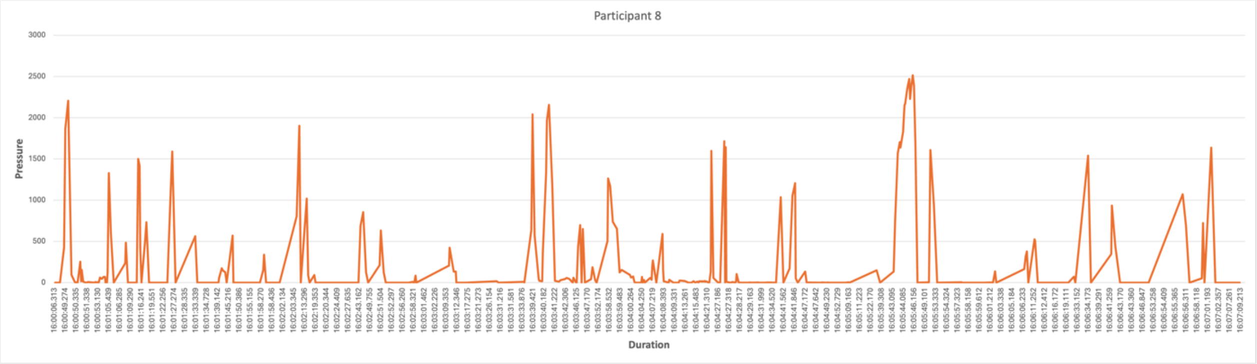

Participant ID 33 Correct Answers by Set
Number of Correct Answers
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Set 1 (easy)
Set 2 (hard)
Set 3 (easy)
Set 4 (hard)
Set 5 (stay)
Set 6 (easy)
Set 7 (stay)
Set 8 (stay)
Set
Ground Truth
Easy
Stay
Hard

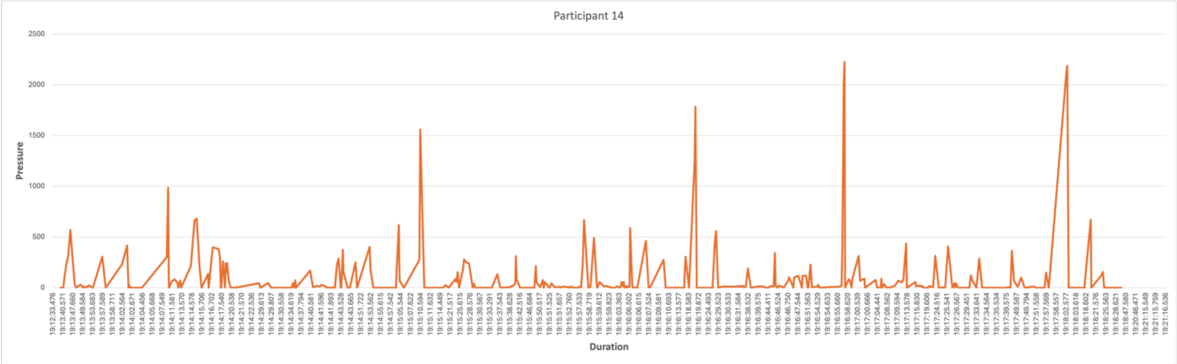
12 12

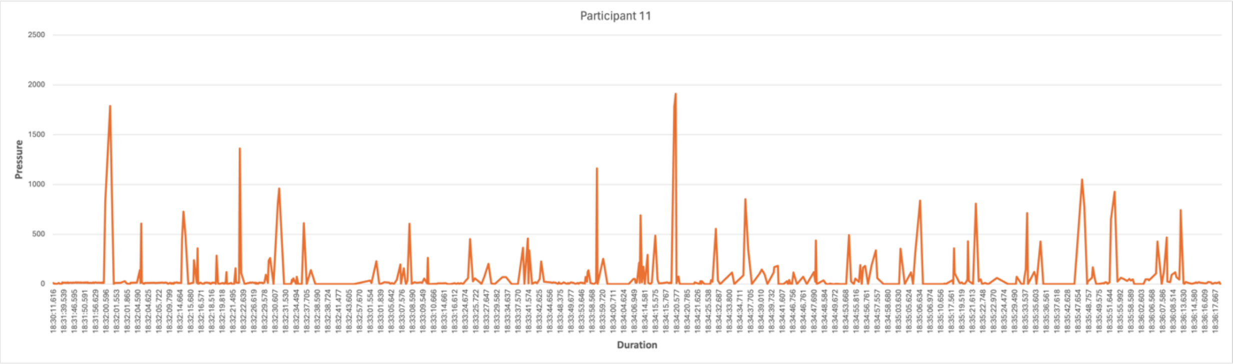

Participant ID 34 Correct Answers by Set
Number of Correct Answers
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Set 1 (easy)
Set 2 (hard)
Set 3 (easy)
Set 4 (hard)
Set 5 (easy)
Set 6 (stay)
Set 7 (stay)
Set 8 (easy)
Set 9 (hard)
Set 10 (hard)
Set 11 (easy)
Set 12 (stay)
Set
Ground Truth
Easy
Stay
Hard

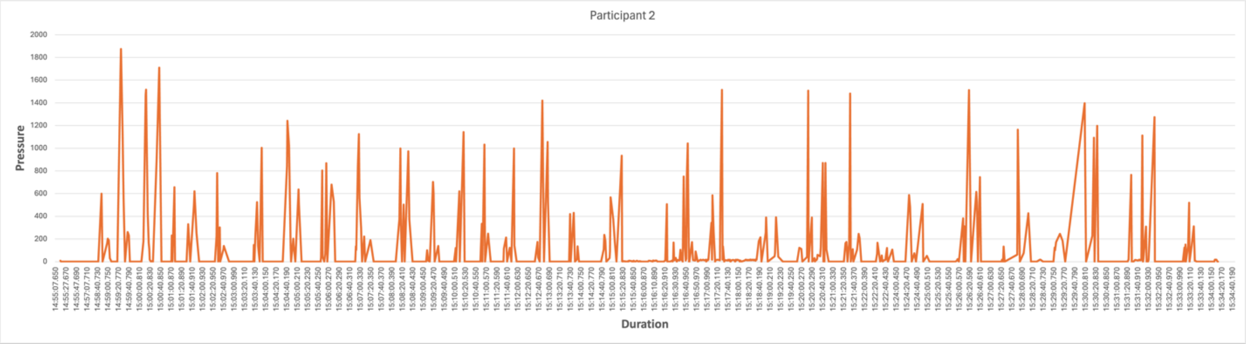
Appendix J Pressure Patterns

Light Touch 1700 - 2100

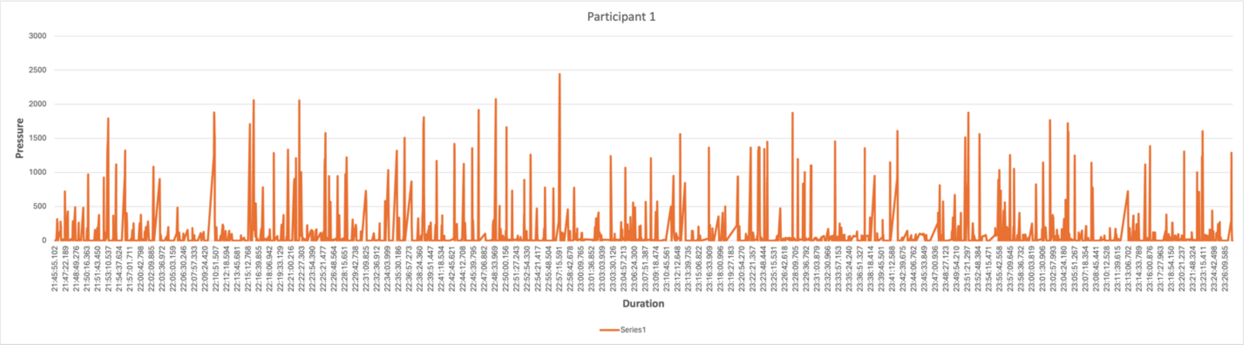


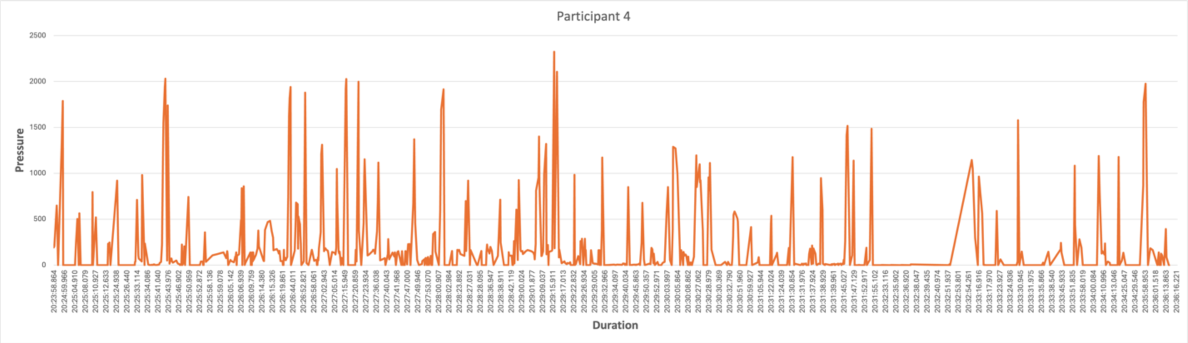


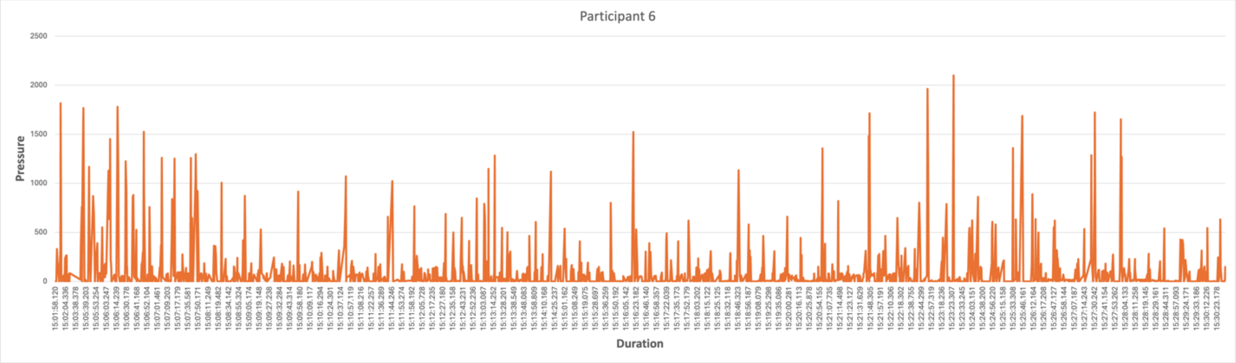


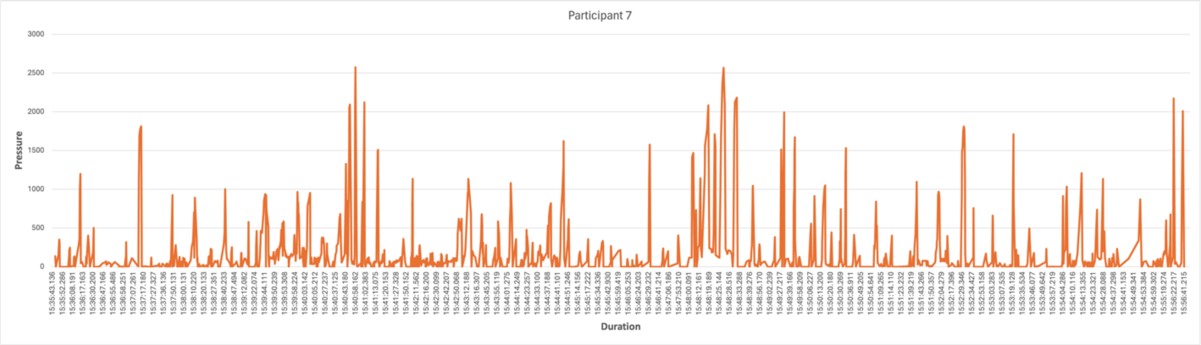


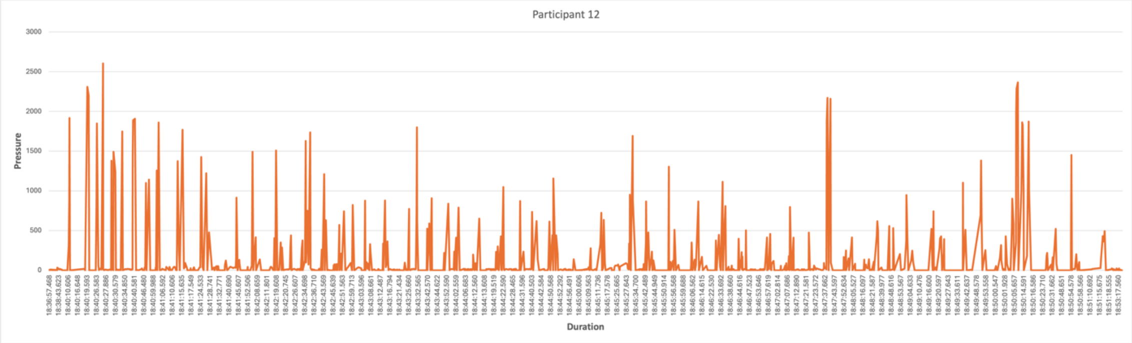
Medium Touch 2100 - 2700



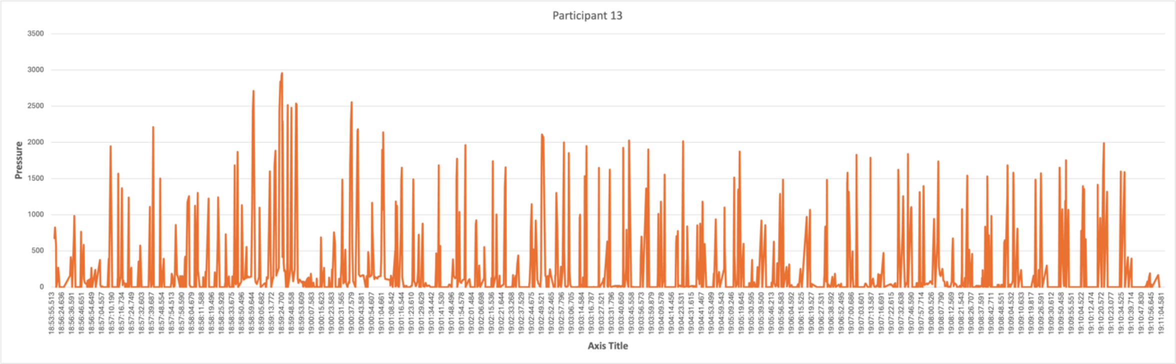


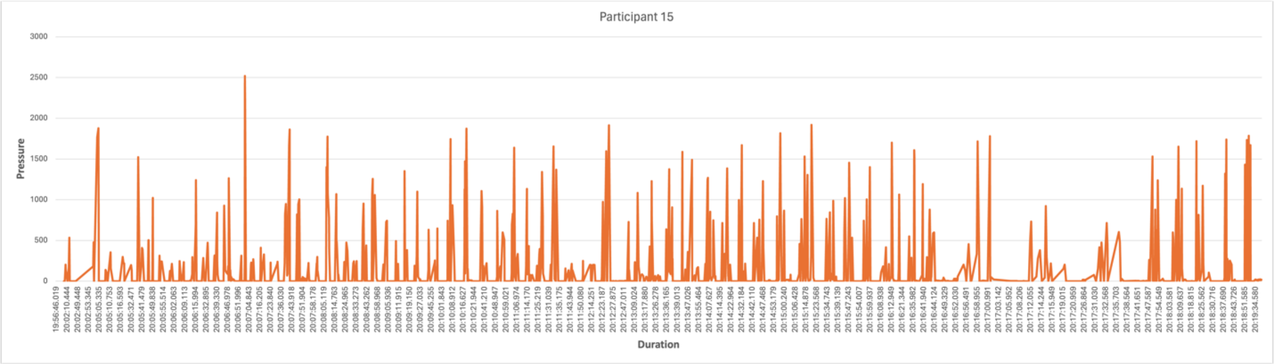


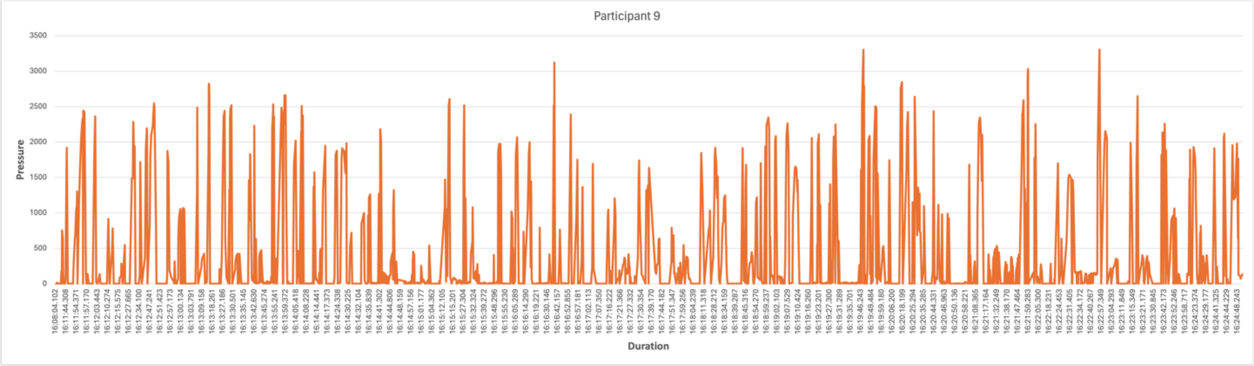


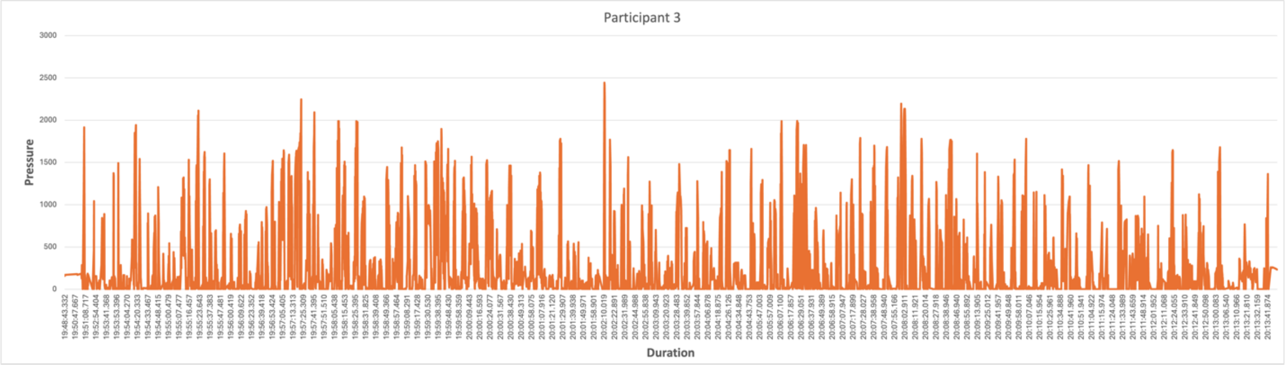

Participant 12
Pressure
3000
2500
2000
1500
1000
500
0
Duration
18:36:57.468
18:38:43.623
18:40:10.606
18:40:16.648
18:40:19.593
18:40:26.583
18:40:27.886
18:40:30.579
18:40:34.850
18:40:40.581
18:40:46.480
18:40:59.988
18:41:06.592
18:41:10.606
18:41:15.635
18:41:17.549
18:41:24.533
18:41:28.741
18:41:32.771
18:41:40.690
18:41:45.607
18:41:52.506
18:42:08.659
18:42:11.801
18:42:19.608
18:42:20.745
18:42:26.607
18:42:34.698
18:42:36.710
18:42:43.569
18:42:45.639
18:42:51.563
18:42:59.713
18:43:03.596
18:43:08.661
18:43:12.487
18:43:16.794
18:43:21.434
18:43:25.660
18:43:32.565
18:43:42.570
18:43:44.622
18:43:52.590
18:44:02.559
18:44:06.683
18:44:12.560
18:44:13.608
18:44:19.519
18:44:22.590
18:44:28.465
18:44:31.596
18:44:40.503
18:44:42.584
18:44:50.568
18:44:52.292
18:44:56.491
18:45:00.606
18:45:09.563
18:45:11.736
18:45:17.578
18:45:25.665
18:45:27.643
18:45:34.700
18:45:42.489
18:45:44.949
18:45:50.914
18:45:56.568
18:45:59.688
18:46:06.562
18:46:14.615
18:46:22.530
18:46:33.692
18:46:38.660
18:46:44.616
18:46:47.523
18:46:53.646
18:46:57.619
18:47:02.814
18:47:07.686
18:47:12.890
18:47:21.581
18:47:23.572
18:47:37.662
18:47:43.597
18:47:52.634
18:48:05.527
18:48:16.097
18:48:21.587
18:48:39.977
18:48:48.616
18:48:53.567
18:49:04.633
18:49:10.476
18:49:16.600
18:49:20.597
18:49:27.643
18:49:33.611
18:49:42.637
18:49:48.578
18:49:53.558
18:50:00.547
18:50:01.928
18:50:05.657
18:50:14.591
18:50:16.586
18:50:23.710
18:50:31.662
18:50:48.651
18:50:54.578
18:50:58.566
18:51:10.692
18:51:15.675
18:51:18.555
18:53:17.560

Hard Touch 2500 - 3300

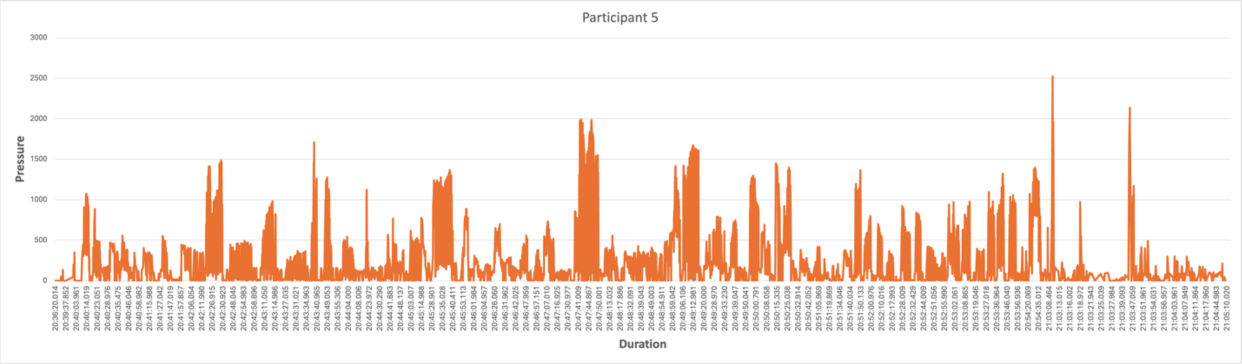


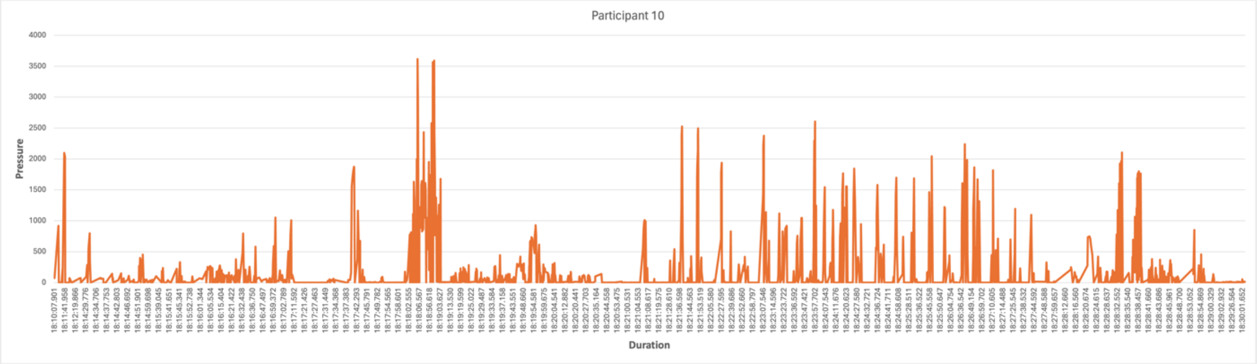


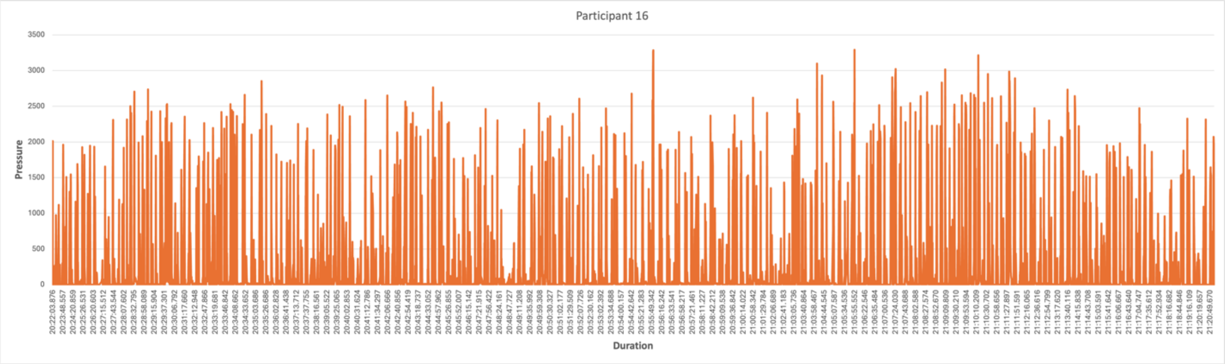




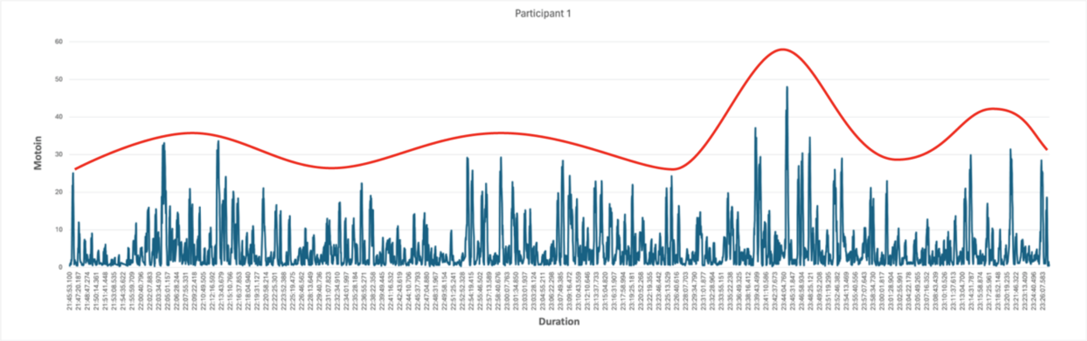
Super Touch 2600 - 3600

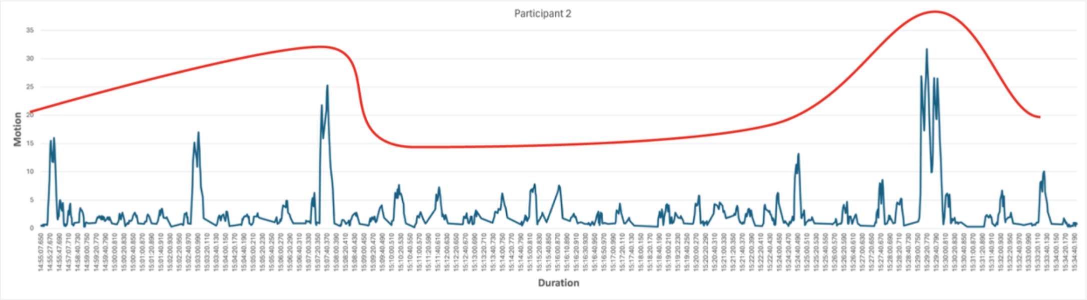




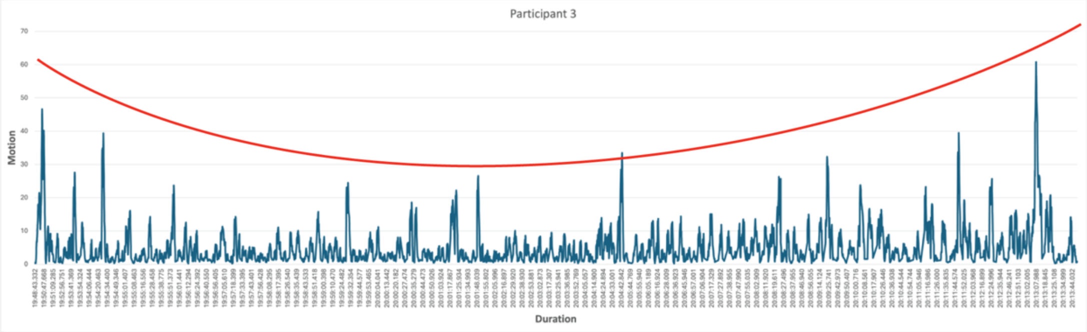


Appendix K Motion Patterns

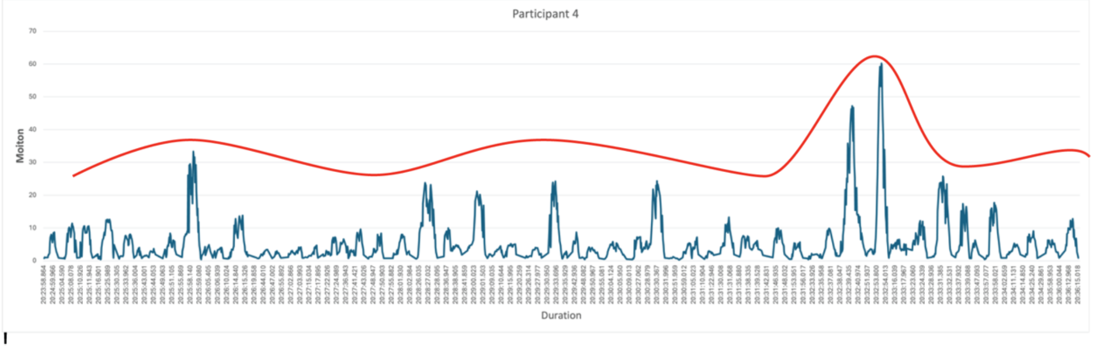




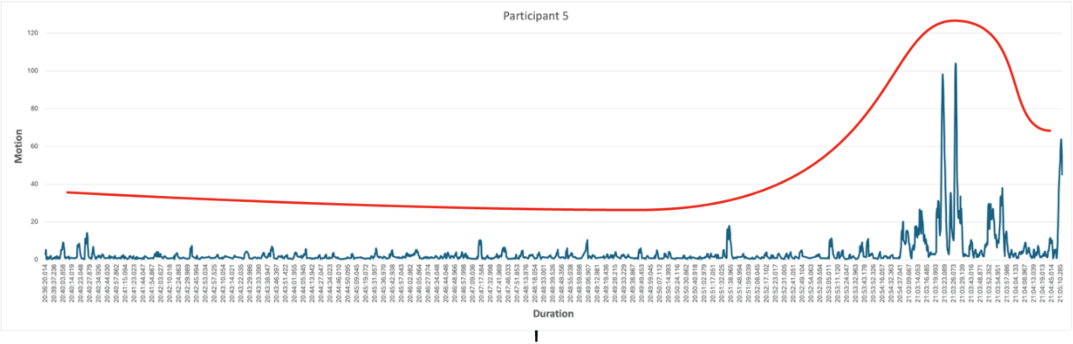
!



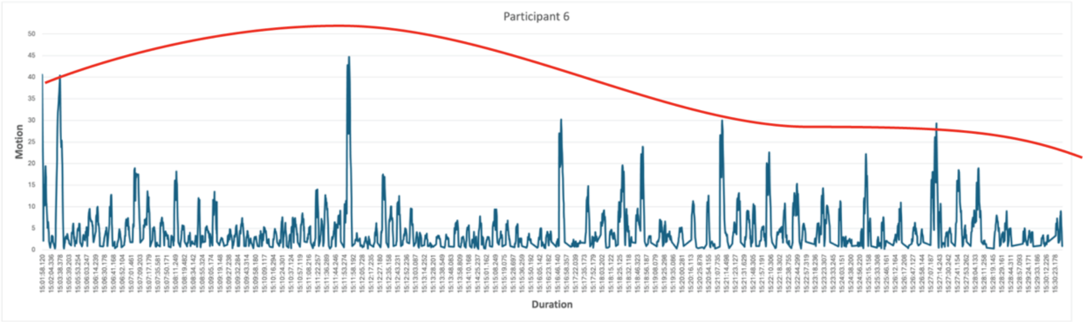
!



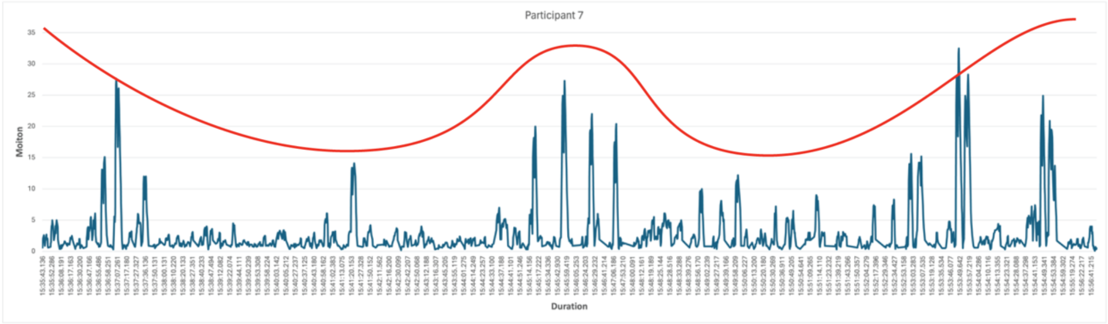
!

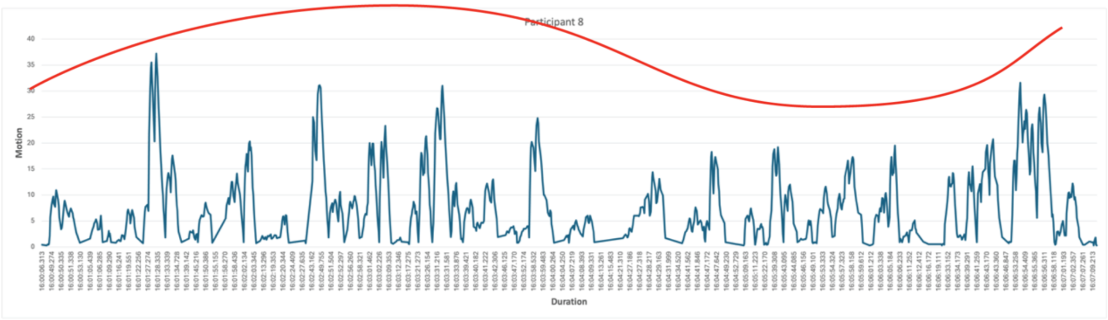

Participant 5
Motion
120
100
80
60
40
20
0
20:36:20.014
21:05:10.285
Duration

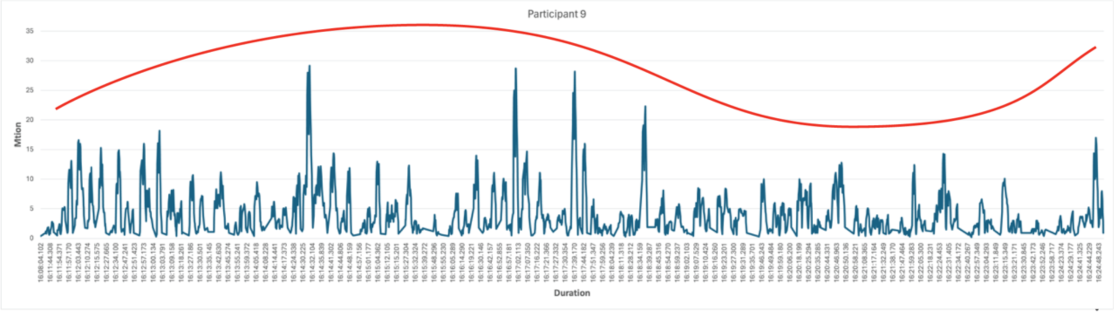
!

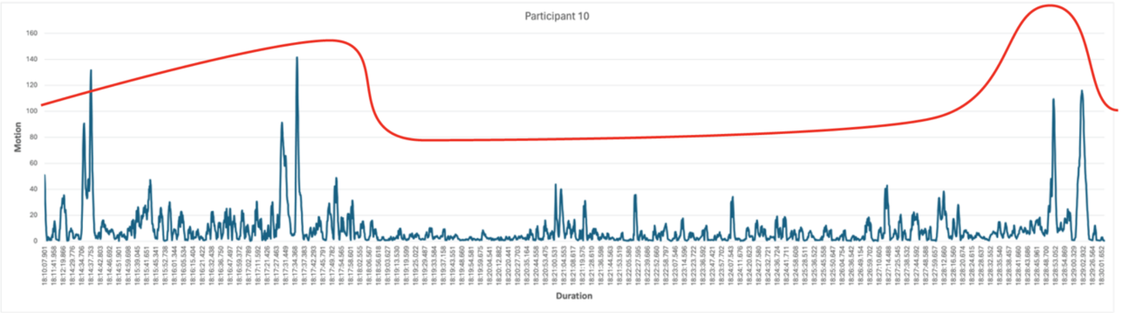

Participant 6
Motion
50
45
40
35
30
25
20
15
10
5
0
15:01:58.120
15:30:23.178
Duration

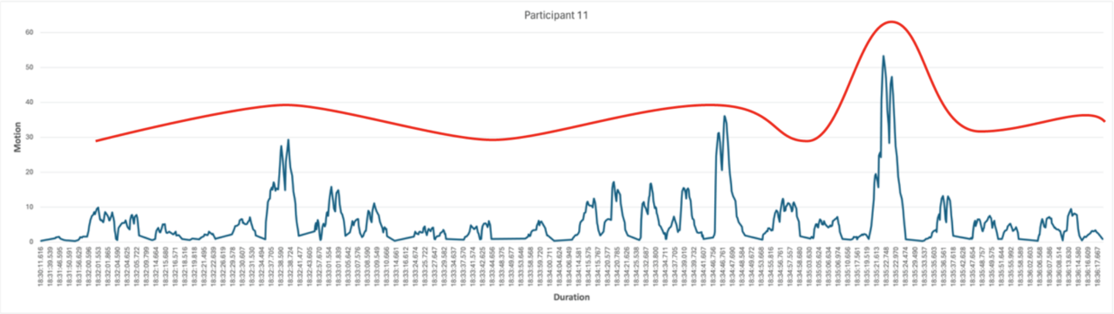
!


Participant 7
Motion
35
30
25
20
15
10
5
0
15:35:43.136
15:56:41.215
Duration

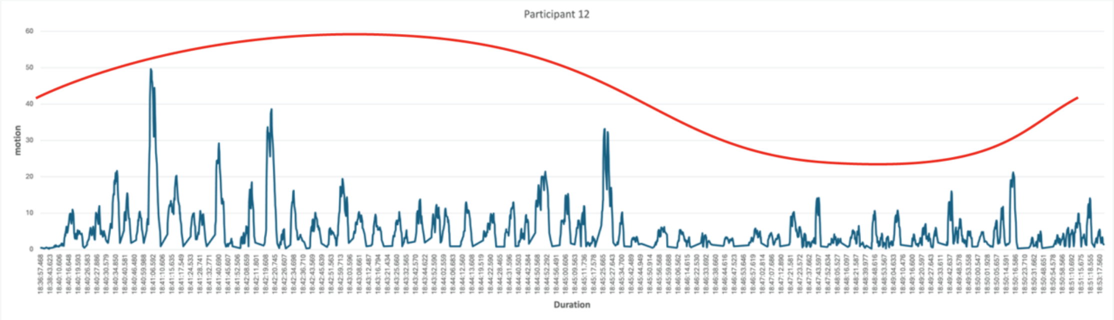

Participant 8
Motion
40
35
30
25
20
15
10
5
0
16:00:09.313
16:07:09.213
Duration

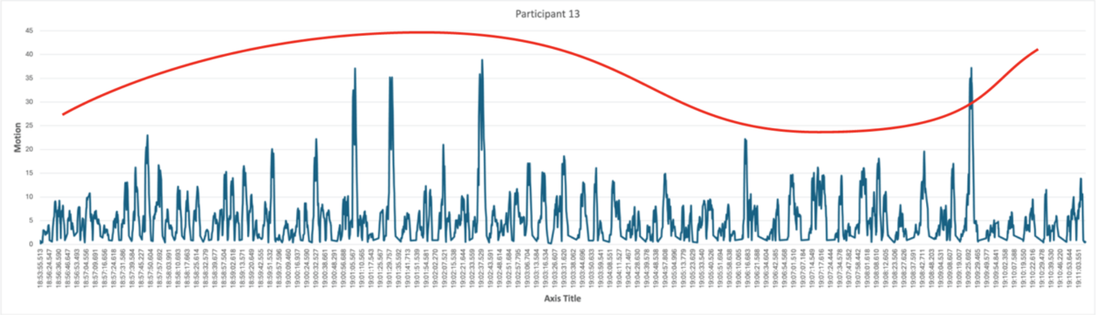

Participant 9
Mtion
35
30
25
20
15
10
5
0
Duration


Participant 10
Motion
160
140
120
100
80
60
40
20
0
Duration

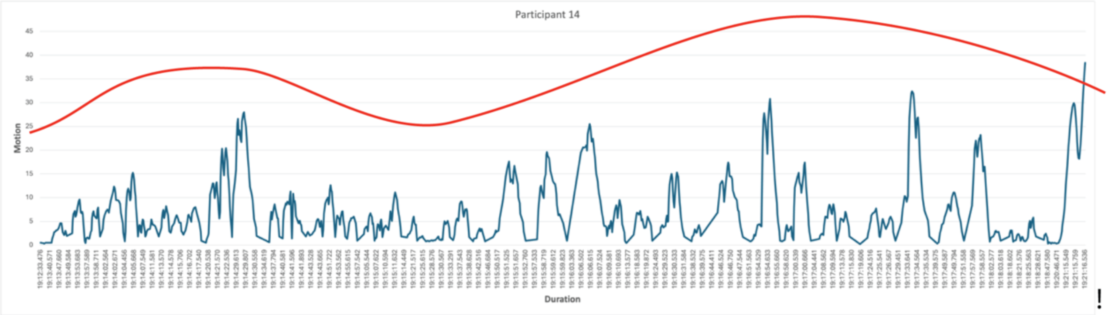

Participant 11
Motion
60
50
40
30
20
10
0
Duration

!

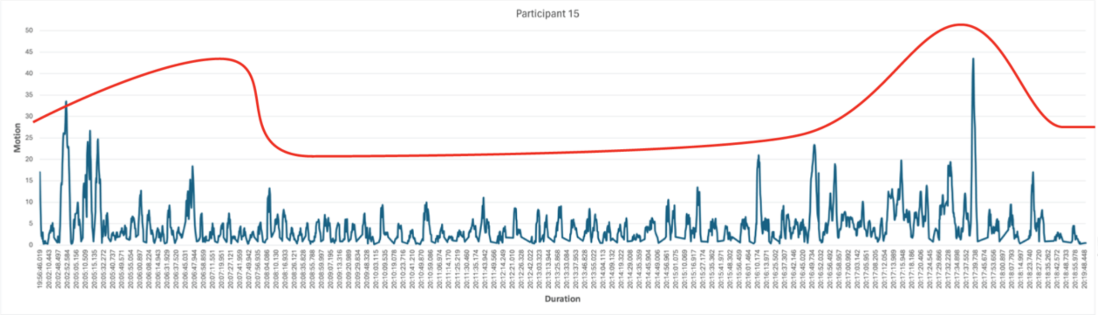

Participant 12
motion
60
50
40
30
20
10
0
Duration


Participant 13
Motion
45
40
35
30
25
20
15
10
5
0
Axis Title

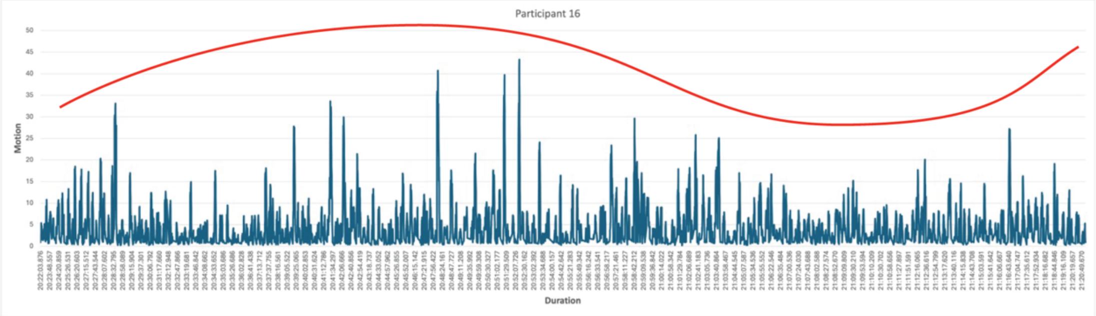
!


Participant 14
Motion
45
40
35
30
25
20
15
10
5
0
Duration

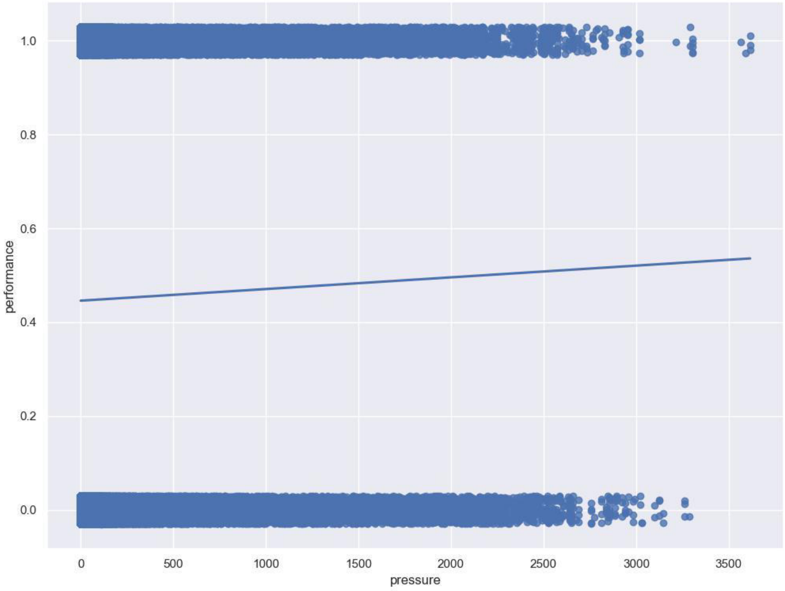
!


Participant 15
Motion
50
45
40
35
30
25
20
15
10
5
0
Duration

!


Participant 16
Motion
50
45
40
35
30
25
20
15
10
5
0
Duration

Appendix L Gradient Descent Graph



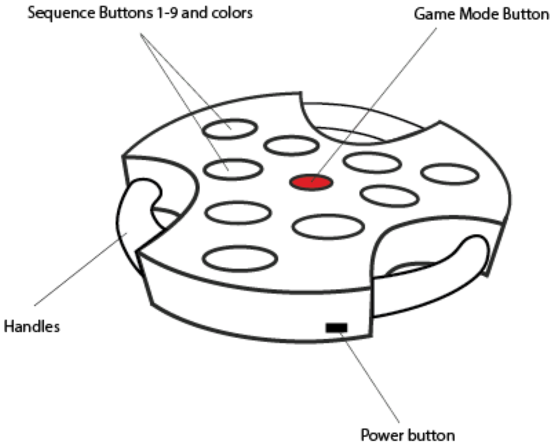
pressure over performance indicates the regression line's slope and the y-intercept.

Appendix M - Instruction Manual for Paper Prototype Study

2
3
1
4
5
6
7
8
9
Operations and Rule of the Game

Setting up the Toy

The following is a set of operations for the smart toy. Generally, the toy needs to be powered, turned on and connected to the internet to begin. Once the toy is connected you can begin to play by pushing the center 'Game Mode' button on the toy.



Follow these steps to connect:

1. In a browser window go to the Network Connect page located at: https://smarttoy.gerryderksen.com
2. Login using your network credentials for username and password
3. Be sure to plug-in a 9-volt battery located at the bottom of the toy
4. Slide the power button to ON position.
5. When the toy appears in the sync item list on the Network Connect page (step 1) select it and the push sync button below to connect the toy.
6. Push the center "Game Mode" button on the toy when you see the toy is connected icon.

How the games are played

The toy will automatically start with the 'Simon says' game. A sequence of colors will be displayed from the edge of the buttons and wait for the player to repeat the sequence. Once five sequences are completed correctly, the toy will switch to math sequences indicated by the number illuminating in the center of each button. You will know the math sequence is correct when the Game Mode button will turn off after each attempt is correctly answered.

The goal of the game

Players are attempting to answer 5 color sequences and 5 math sequences correctly in a row. If they are not correct the game will repeat the sequence until they are. All colors and numbers will light up when this is accomplished. The sequences will then move the next level of difficulty for you to challenge yourself. The games are only limited by the number of possible color or number sequences without repeating.

Trouble shooting problems

Always check the connection settings on the website https://smartToy.gerryderksen.com

Can't connect to network

If the toy does not connect to the internet immediately after set-up, select the systems preference icon and check to see that the toy recognizes the network your phone or laptop are on. If the networks are different, change the toy or your device network so they are the same. For your device, see the manual or website for network settings on how to change networks.

How do I change the toy's network?

To change the network for the toy; select it in the sync item list and edit the network settings by selecting the pen icon also on the right of the toy name. Select the correct network that matches the device you are viewing the app. To change your device network settings, refer to your owner's manual.

Cant turn on the toy

If the toy is not on check the battery located on the bottom of the toy. The battery storage is inside the toy and can be accessed by opening the battery cover. Use the lever to pull up on the cover to open. Check the battery connection to ensure it is on or check that the battery is fully charged or has power. Recharge or replace the battery with a 9v similar to the one in the opening. Now turn on the power button located on the side of the toy.

How can I change the starting game?

Once the toy is connected to the internet and the first game sequence begins, press and hold the game mode button for 3 seconds. When you release the button, it will remain on indicating it is in the math game mode. The following sequence will be the first math sequence. The toy will remember this selection and continue to start with the math game until these steps are followed again to return it to the default color game.

How can I change the level of the game?

If the game is in the math sequence press and hold the game mode button for three seconds. This will reset the game to level 1. If the game is in the color sequence press and hold the game mode button for three seconds. The game will first change to the math sequences. Press and hold the game mode button a second time and the game will reset to level 1.