PREDICTING STUDENT PERFORMANCE IN AN AUGMENTED REALITY LEARNING ENVIRONMENT USING EYE-TRACKING DATA

A Thesis Presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

MATTHEW DEAY

Dr. Kangwon Seo, Thesis Supervisor

May 2023

The undersigned, appointed by the dean of the Graduate School, have examined the thesis entitled

PREDICTING STUDENT PERFORMANCE IN AN

AUGMENTED REALITY LEARNING ENVIRONMENT

USING EYE-TRACKING DATA

presented by Matthew Deay,

a candidate for the degree of Master of Science,

and hereby certify that, in their opinion, it is worthy of acceptance.

Professor Kangwon Seo

Professor Jung Hyup Kim

Professor Danielle Oprean

ACKNOWLEDGEMENTS

I would first like to express the upmost gratitude to my thesis supervisor, Dr. Kangwon Seo, for his support throughout the research process. He was an exceptionally helpful and knowledgeable resource for me throughout this last semester and the guidance he has provided to me has been invaluable. Thank you, Dr. Seo!

I would also like to thank Dr. Jung Hyup Kim, Dr. Danielle Oprean, and the rest of the research team for their continued feedback and support. The development of the augmented reality lecture program as well as a majority of the eye-tracking data collection was done by the rest of the team before I joined the group. None of my work would have been possible without them. Thank you all for giving me the opportunity to work with you and I wish you all the best of luck as you continue to push the boundaries of educational technology!

Thank you to the rest of the Mizzou Industrial & Systems Engineering faculty and staff as well! This work has been a culmination of the lessons I've learned throughout the last five years that I've spent as a student in this department. I will continue to use the knowledge I've gained here throughout the rest of my career.

Lastly, I'd like to thank my friends and family for how they've helped me throughout the entirety of my academic career. It has been a long journey and I can't imagine how I would've made it without their love and support!

This research is supported by NSF IIS-2202108: Research on the Use of Real-Time Tracking and Eye-Tracking Technology for Integrating Metacognition and Augmented Reality into Undergraduate Engineering Laboratories.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS .... **ii**

LIST OF TABLES .... **v**

LIST OF FIGURES .... **vi**

ABSTRACT .... **viii**

Chapter 1 - Introduction **1**

1.1 Motivation 1

1.2 Research Overview 2

1.3 Objectives 3

Chapter 2 - Literature Review **4**

2.1 Augmented Reality Applications 4

2.2 Assessment of Augmented Reality in Education 6

2.3 Eye-Tracking Data Analysis 7

2.4 Analyzing Cognitive States in Augmented Reality 10

Chapter 3 - Augmented Reality Lab Design **12**

3.1 Equipment 12

3.2 Participants 12

3.3 Learning Content 13

3.4 Procedure 14

Chapter 4 - Methods **15**

4.1 Data Preprocessing 15

4.1.1 Output Data Problems 15

4.1.2 Importing & Cleaning Data 16

4.2 Student Data 18

4.3 Baseline Data Collection 20

4.4 Average Difference Method 23

4.4.1 Methodology 23

4.4.2 Statistical Analysis 25

4.4.3 Remove Models with 100% Accuracy 28

4.5 Average Euclidean Distance Method 30
4.5.1 Methodology 30
4.5.2 Statistical Analysis 32
4.5.3 Remove Modules with 100% Accuracy 33
4.6 Distraction Rate Method 35
4.6.1 Methodology 35
4.6.2 Statistical Analysis 40
4.6.3 Remove Modules with 100% Accuracy 42
4.6.4 Parameter Optimization 44
Chapter 5 - Results 47
5.1 Results Summary 47
5.2 Discussion of Results 48
5.3 Applications 50
5.3.1 Module-Based Feedback System 50
5.3.2 Attention Guidance System 51
Chapter 6 - Conclusions 54
6.1 Conclusions 54
6.2 Limitations 54
6.3 Future Studies 56
REFERENCES 57
APPENDIX 61

LIST OF TABLES

Table 4-1: Eye-tracking output data column descriptions. *16*

Table 4-2: Eye-tracking output data (student 1, lecture 1, module 7).............................. *16*

Table 4-3: Preprocessed student data (student 1, lecture 1).... *18*

Table 4-4: Distraction rate parameter optimization test results. *45*

Table 5-1: Results summary.. *47*

Table 5-2: Module-based feedback system attention levels.... *51*

LIST OF FIGURES

Figure 3-1: Microsoft HoloLens. *12*

Figure 3-2: Scene arrangement. *13*

Figure 3-3: Experimental layout. *14*

Figure 4-1: Data preprocessing R script file. *17*

Figure 4-2: Answer correctness by student. *19*

Figure 4-3: Answer correctness by module. *20*

Figure 4-4: X-coordinates vs baseline (student 2, module 2-2). *21*

Figure 4-5: Y-coordinates vs baseline (student 2, module 2-2). *22*

Figure 4-6: X-coordinates vs baseline (student 2, module 2-7). *23*

Figure 4-7: Y-coordinates vs baseline (student 2, module 2-7). *23*

Figure 4-8: Average x-coordinate difference vs module. *24*

Figure 4-9: Average x-coordinate difference R script. *25*

Figure 4-10: Average x-coordinate difference of modules with correct vs incorrect answers. *26*

Figure 4-11: Average x-coordinate difference simple logistic regression model. *27*

Figure 4-12: Average x-coordinate difference vs module (excluding modules with 100% accuracy). *29*

Figure 4-13: Average x-coordinate difference of modules with correct vs incorrect answers (excluding modules with 100% accuracy). *29*

Figure 4-14: Average x-coordinate difference simple logistic regression model (excluding modules with 100% accuracy). *30*

Figure 4-15: Average Euclidean difference vs module. *31*

Figure 4-16: Average Euclidean difference R script. *31*

Figure 4-17: Average Euclidean difference of modules with correct vs incorrect answers. *32*

Figure 4-18: Average Euclidean difference simple logistic regression model. *33*

Figure 4-19: Average Euclidean difference vs module (excluding modules with 100% accuracy). *34*

Figure 4-20: Average Euclidean difference of modules with correct vs incorrect answers (excluding modules with 100% accuracy). *34*

Figure 4-21: Average Euclidean difference simple logistic regression model (excluding modules with 100% accuracy). *35*

Figure 4-22: Euclidean difference compared to signal threshold of 1.5 (student 2, module 2-2). *37*

Figure 4-23: Euclidean difference compared to threshold of 1.5 (student 2, module 2-7). *38*

Figure 4-24: Moving average Euclidean difference compared to threshold of 1.5 (student 2, module 2-2). *39*

Figure 4-25: Moving average Euclidean difference compared to threshold of 1.5 (student 2, module 2-7). *39*

Figure 4-26: Distraction rate vs module. *40*

Figure 4-27: Distraction rate R script *40*

Figure 4-28: Average distraction rate of modules with correct vs incorrect answers. *41*

Figure 4-29: Distraction rate simple logistic regression model. *42*

Figure 4-30: Distraction rate vs module (excluding modules with 100% accuracy). *43*

Figure 4-31: Distraction rate of modules with correct vs incorrect answers (excluding modules with 100% accuracy). *43*

Figure 4-32: Distraction rate simple logistic regression model (excluding modules with 100% accuracy). *44*

Figure 5-1: Distraction rate distribution. *51*

Figure 5-2: Attention guidance system diagram. *53*

ABSTRACT

This paper investigates the use of eye-tracking data as a predictor of student performance in an augmented reality (AR) learning environment. 33 undergraduate students enrolled in an ergonomics course at the University of Missouri-Columbia participated in an AR biomechanics lecture consisting of 14 modules. Following each module students answered learning comprehension questions to test their understanding of the lecture material. An additional dataset was recorded for each module in which the participant perfectly follows the virtual instructor throughout the learning space. This dataset, referred to as the baseline, can be used as a comparison tool to gauge how well students follows the lecture material. Two methods are proposed to quantify the student's attention level for each module. The average difference method calculates the average distance between the student and baseline coordinates for each module. The distraction rate method expands upon the average difference method and aims to reduce the amount noise detected. This is done by incorporating a minimum distance threshold, a binary detection signal, and a moving average window. Both metrics are tested as factors in a set of logistic regression models to determine whether they can accurately predict student answer correctness. Average difference showed a correlation with student answer correctness, but with an underwhelming level of significance. Distraction rate outperformed average difference and proved to be a strong and statistically significant predictor of student answer correctness. Finally, two feedback systems are proposed which use distraction rate to detect when students have become distracted so that their attention can be regained through the use of module-based feedback or a real-time attention guidance system.

Chapter 1 - Introduction

1.1 Motivation

Following the lockdown caused by the COVID-19 pandemic, academic institutions of all levels were forced to transition to virtual learning environments. One of the greatest challenges faced by instructors during this transition was replicating the hands-on learning experience previously provided by in-person teaching. Augmented Reality (AR) platforms provide a unique solution to this problem as they promote spatial imagination and thinking which can boost the student's understanding of complex lecture material (Gurevych et al., 2021). If AR learning modules can be designed in a way that replicates the dynamic and engaging experience found in an in-person lab, it would significantly reduce gap between virtual and in-person learning experiences.

One of the key challenges in providing a comprehensive AR learning experience is to ensure students are focusing on the correct objects within the 3-dimensional learning space. Recent studies have found that students in AR learning environments are prone to cognitive overload and distraction (Akçayır & Akçayır, 2017). During in-person lab sessions, instructors can monitor their students and intervene if students become confused or stop paying attention. This type of interaction is not accounted for in a pre-recorded lecture, which further contributes to the gap between virtual and in-person learning environments. If the student's attention levels can be tracked and analyzed throughout the learning session, then feedback can be provided to them in real-time to attempt to regain their focus. Real-time feedback systems could be implemented to reduce premature termination of learning and potentially improve student performance. Research has shown an association between eye-tracking measurements and learning performance (Alemdag & Cagiltay, 2018). Fortunately, eye-tracking capabilities are becoming a common feature on AR headsets. This makes eye-tracking data analysis a potentially powerful method for monitoring attention levels and predicting student performance in AR learning environments.

1.2 Research Overview

The data used in this research was collected from a group undergraduate students enrolled in an ergonomics course at the University of Missouri-Columbia in the Fall semester of 2022. Each of the students participated in a total of 14 AR lecture modules. Following each of the modules, students would answer a multiple-choice question related to the material covered in that module. The student's eye-tracking data was recorded throughout the duration of the lectures. This data consisted of x and y-coordinate data along with corresponding timestamps.

In order to analyze the student's eye-tracking data, an additional dataset referred to as the baseline was collected. The coordinates of the baseline dataset closely follow the verbal instructions given by the virtual instructor and represent where the student's attention is supposed to be directed throughout the lecture. The baseline dataset can be compared to the student's coordinates to determine how well the student is following the lecture material. This comparison is done using two different attention monitoring metrics which have been developed. The first, referred to as average difference, is the average distance between the student and baseline coordinates throughout the module. The second metric, distraction rate, incorporates a minimum distance threshold, binary detection signal, and moving average window to reduce the amount of noise detected by the metric. Statistical tests are then conducted to determine whether there is a correlation between these metrics and student answer correctness. Next, multiple logistic regression models are fitted to evaluate whether these metrics can be used to accurately predict student answer correctness. Finally, the threshold and moving average window parameters used in the distraction rate calculations are tested to determine which parameter values yield the highest significance levels in the aforementioned statistical tests. As a result of this research, a module-based feedback system and an attention guidance system are recommended. The proposed frameworks for both of these systems are provided as well.

1.3 Objectives

The main objective of this research is to develop an attention monitoring metric capable of predicting student performance. The most important part of this process is to validate the relationship between eye-tracking data and student performance. This will be accomplished through the use of statistical testing and logistic regression models. If the proposed metrics are capable of accurately predicting student performance, then they can be implemented as a part of a real-time attention monitoring system. This system will provide feedback to students with the intention of reducing the negative effects of cognitive overload and preventing premature termination of learning.

Chapter 2 - Literature Review

2.1 Augmented Reality Applications

In recent years, augmented reality has emerged as an effective alternative to standard teaching and training practices. School subjects including chemistry, astronomy, physics, biology, mathematics, and geometry have all found new ways to implement AR learning as part of their curriculum (K. Lee, 2012). One study found that STEM subjects have seen particularly positive results in the form of student feedback as well as improvements in academic performance. The collaborative, interactive, and immersive nature of AR learning environments enriches students' learning experiences and thus contributes to their learning effectiveness (T. Lee et al., 2022). One potential reason for AR's effectiveness in STEM subjects is its ability to promote spatial intelligence. Spatial intelligence can be defined as the mental ability to understand and solve real-world problems. AR allows students to visualize and interact with objects in three dimensions, which helps further their understanding of complex problems. Research findings have shown that AR technology has a positive effect on spatial intelligence in mathematics (Ban Hassan Majeed & ALRikabi, 2022).

In one junior high school, experiments in AR-based applications, including a series of mathematics lessons on probability, found success as well. Results showed that mobile AR-based applications would be helpful for students' learning gains on the topic of probability. Additionally, students displayed positive attitudes towards the AR applications in this series of lessons(Cai et al., 2020). Martin-Gutierrez and Meneses Fernández (2014) implemented an AR program to assist mechanical engineering students in the subject of graphic engineering. They found that engineering students obtain better academic results and are more motivated when the new generation of technological tools are incorporated into the learning process. Another application of AR-based learning is in vocational higher education. Radosavljevic et al. (2020) compares the results of traditional learning and learning using AR in the part of the curriculum important for vocational skills. Results showed that AR helps to reduce the time of realizing a task as opposed to realizing it using printed materials. Kaur et al. (2022) developed an AR learning environment utilizing mobile and table-top design variants which was tested in a case study involving 60 undergraduate students of electronics and electrical engineering. Students who participated showed increased motivation and satisfaction.

AR applications in learning are not limited to higher levels of education. Lindgren et al. (2016) conducted a study where middle school students learned about gravity and planetary motion in an immersive, whole-body interactive simulation. Results of the study indicated that enactive concepts and experiencing critical ideas in physics through the whole body leads to significant learning gains, higher levels of engagement, and more positive attitudes towards science. Another study by Dunleavy et al. (2009) conducted multiple qualitative case studies across two middle schools (6^{th} and 7^{th} grade) and one high school (10^{th} grade). Teachers and students reported that the technology-mediated narrative and the interactive, situated, and collaborative problem-solving affordances of the AR simulation were highly engaging, especially among students who had previously presented behavioral and academic challenges for the teachers.

Medical training is another field where AR can have a positive impact. The shift towards online learning caused by COVID-19 highlights this fact as medical personnel were required to take added precautions to reduce exposure. In a review conducted by (Dhar et al., 2021), it was found that AR-based training provides a vast potential to prepare medical professionals effectively and efficiently for the real world of practice. AR can also be applied in business sectors such as tourism, museums, or gaming (K. Lee, 2012). The AR game-based learning environment developed by Chen et al. (2015) received positive feedback from its pilot study participants. Industrial applications can also be especially useful for providing non-expert users with helpful information about the functionality of complex automated systems (Heinz et al., 2019). AR-based training can even be used for emergency protocol training. Stigall et al. (2018) proposed an architecture and describes the design and implementation of an AR application to leverage the Microsoft HoloLens for building evacuation purposes. Pilot studies of the system demonstrated the effectiveness of the application in an emergency evacuation.

2.2 Assessment of Augmented Reality in Education

The recent popularity of augmented reality in education has resulted in an abundance of literature discussing the advantages and challenges of implementing it. In a review by Alzahrani (2020), multiple studies indicated that one of the most fundamental advantages of AR in education lies in its ability to support kinesthetic learning. This stems from how AR creates an interactive learning system that allows students to understand and memorize content through 3D visualizations. One of AR's most defining features is its ability to enhance the preexisting classroom environment. AR provides instructors with a way to strengthen students' understanding in the classroom by augmenting physical props with virtual annotations and illustrations (Saidin et al., 2015). In AR, there is an intimate relationship between virtual and physical objects. The physical objects can be enhanced in ways not normally possible such as providing dynamic information overlay, private and public data display, context sensitive visual appearance, and physically based interactions (Billinghurst, 2002). AR has also been shown to increase student attention levels. In an experiment conducted by Bos et al. (2019), user attention was monitored through an electroencephalography (EEG) sensor while performing an educational task using either AR or a traditional interface. An increase in student attention was identified during the interaction with the AR application, as opposed to its conventional counterpart. Other studies also report high levels of independent thinking, creativity, and critical analysis from students using AR compared to traditional learning (Bower et al., 2014).

One of the most prominent challenges of AR is cognitive overload. Students in AR environments may find difficulties with the large amount of information they encounter, the multiple technological devices they are required to use, and the complex tasks they must complete (Wu et al., 2013). Another similar issue is attention tunneling. Students reportedly experience higher attentional demands from AR systems. This results in the students ignoring important parts of the experience or feeling unable to properly perform team tasks (Radu, 2012). Some other challenges include technical problems, design difficulties, expensive technology, and that AR is difficult for students to use (Akçayır & Akçayır, 2017).

2.3 Eye-Tracking Data Analysis

Across a variety of learning environments, eye-tracking data analysis has managed to overcome limitations in the study of cognitive processes linked to learning and performance (Rodrigues & Rosa, 2017). An experiment conducted by Wang et al.

(2016) utilized eye-tracking measures such as total reading time, total fixation duration, number of fixations, and inter-scanning count to predict learning outcomes. They found that on dynamic, multimedia webpages the inter-scanning count between text and video zones had a significant negative correlation with retention scores. The total number of fixations also had a significant positive correlation with retention scores. Li et al. (2020) utilized eye-tracking data to train a machine learning to predict the difficulty level of e-learning problems. The results confirmed that eye movement, especially fixation duration, contains essential information on the difficulty of problems and is sufficient to build machine learning models to predict problem difficulty level. Mayer (2010) investigated the link between eye-fixation measures and learning outcomes. Out of six case studies, four concluded that there is evidence of a link between perceptual processing of relevant portions of graphic information and measures of cognitive performance on an intellectually demanding task. Chettaoui et al. (2023) applied predictive modeling to identify the synergies between eye-gaze features and students' learning performance. The obtained results suggest that combining eye-gaze tracking with learning traces and behavior attributes may support an accurate prediction of students' learning performance.

In another study, eye movement data was recorded from 40 students who watched lecture videos. Using an artificial intelligence algorithm, researchers were able to predict student performance with an error of less than 5% (Sharma et al., 2020). Eye-tracking data can also measure how well students are paying attention to lecture material. Sharma et al. (2015) was able to detect the difference between students who engage with their teacher or collaborating partner through the interface/display and students who only engage with the material. Eye-tracking data can also be analyzed alongside other cognitive sensor data to predict student performance. Khedher et al. (2019) used both eye-tracking and electroencephalography data to train a K-Nearest Neighbor classification algorithm to accurately discriminate between students who successfully resolved a problem-solving task and students who did not.

In another method proposed by Buettner et al. (2018), eye-tracking based pupillometry was used to capture pupil diameter data and calculate user performance expectations via a Random Forest Algorithm. The results showed a good classification accuracy of user performance after only 40 seconds (5% of the mean total runtime). Peterson Joshua and Pardos (2015) also analyzed the predicting power of pupillometry in addition to over 40 other high-level gaze features. They found that certain gaze features are strong predictors of performance, but less so of learning gains, while pupil diameter is marginally predictive of learning gains, but not performance.

Eye-tracking data has proven to be an effective indicator of attention levels within virtual reality (VR) learning environments as well. Asish et al. (2022) proposed an automated system based on machine learning to classify students based on their distraction level using eye gaze data. Results showed that a Random Forest algorithm was capable of classifying student attention as one of three levels (low, medium, high) with an accuracy of 98.88%. Eye-tracking data can also be used to visualize student eye gaze patterns in real-time, giving teachers useful insights on student attention levels. Rahman et al. (2020) proposed six gaze-visualization techniques for a VR-embedded teacher's view and conducted a user study to compare them. The results suggested that a short particle trail representing eye-tracking trajectory is promising.

2.4 Analyzing Cognitive States in Augmented Reality

Strictly within AR learning environments, eye-tracking hasn't been extensively tested as an indicator of student performance or attention levels. Regardless, the studies on the subject have yielded promising results thus far. In a paper by Dzsotjan et al. (2021), they discuss their ongoing construction of an AI framework to quantify and predict the learning gain of the user, examining the predictive potential of gaze data collected during the app usage. Experimental results showed that a support vector machine yields the highest accuracy, and the K-Nearest Neighbor and Random Forest Classifiers found success as well.

Besides eye-tracking, there are several other techniques which have been implemented in AR learning environments to monitor students' cognitive states. Brain activity sensors such as EEG and functional near-infrared spectroscopy (fNIRS) have both been used to estimate attentional states in AR (Vortmann, 2019). Skin conductance (or electrodermal activity – EDA) is another indicator that can be monitored via skin biosensors. The data collected from these sensors has proven to be an effective way to track student engagement during AR lab activities (Soltis et al., 2020).

As previously discussed, it is common for students to become overwhelmed in AR learning environments due to the large quantity and variety of content being presented to them. Attention guiding systems are an effective way to ensure users can efficiently find the desired information within the AR space. Biocca et al. (2006) conducted an experiment in which an attention funnel and other conventional AR attention directing techniques were implemented. Results showed a 65% increase in user search consistency, 22% increase in search speed, and an 18% decrease in mental workload. Such systems have applications outside of education as well. Renner and Pfeiffer (2017) designed a smart glasses-based assistance system for a manual assembly station which incorporated several attention guiding techniques, some of which incorporated eye-tracking data. Considering how prominent the issue of cognitive overload is in AR learning environments, attention guiding systems should be considered to attempt to mitigate its negative effects on learning.

Chapter 3 - Augmented Reality Lab Design

3.1 Equipment

The experimental AR lectures took place in the ergonomics lab of the Industrial & Systems Engineering department at the University of Missouri-Columbia. The lab room consists of an open room with a single moveable desk where the student can write down calculations as they answer questions related to the lecture material. The AR headset used in this experiment is the Microsoft HoloLens. The HoloLens allows students the freedom to turn their head in any direction as they experience the AR lecture. It is also capable of recording eye-tracking coordinate data via the orientation of the headset. This is not the same as tracking pupil movements, but still indicates approximately where the student is looking at any given moment.

Figure 3-1: Microsoft HoloLens.

3.2 Participants

The participants of this experiment were undergraduate students at the University of Missouri-Columbia enrolled in an ergonomics course in the Industrial & Systems Engineering department. There were 33 total participants with an average age of 21.75

(standard deviation of 4.27). Data collection took place over a three-week span during the Fall 2022 semester.

3.3 Learning Content

The lecture material covered in this lesson is on the subject of biomechanics. Biomechanics problems provide an excellent opportunity to take advantage of the AR learning environment with 3-dimensional animated figures. These figures aim to help the student visualize the problem context, which is sometimes difficult to represent in two dimensions. The scene arrangement for each of the modules consists of five panels where text, calculations, and data tables can be displayed. In front of the panels, the virtual instructor will move throughout the 3-dimensional space to guide the student's attention towards the current point of interest.

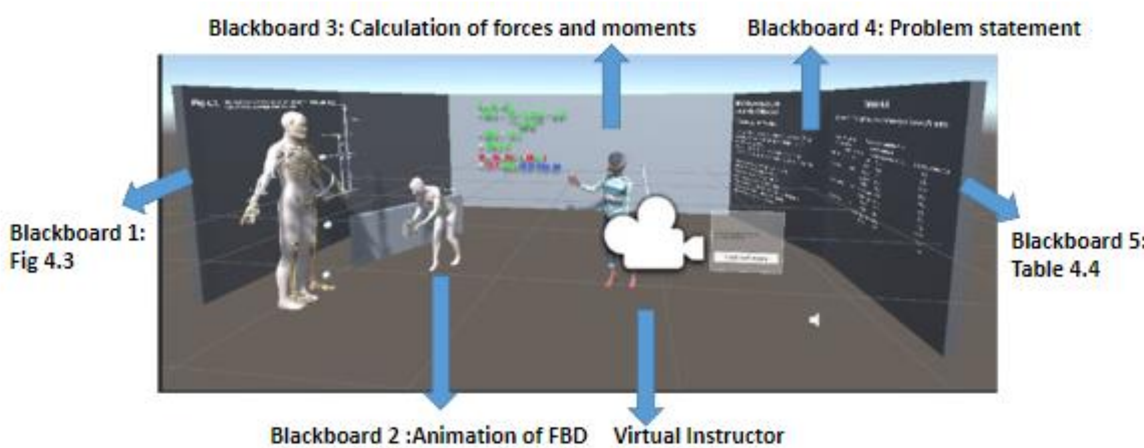

Figure 3-2: Scene arrangement.

The learning content is split into two different lectures. Both lectures consist of seven modules, all of which are followed by a quiz question to assess the students' comprehension of the material. The difficulty level of the learning content is increased in the second lecture. The first lecture focuses on basic biomechanics concepts. Lecture 2 expands upon the material from the first lecture with example problems which walk through complex calculations.

3.4 Procedure

The participant begins each lecture at station 1 with a moveable table for answering questions. As the participant progresses through the modules, they move across the room with the table which is attached to a Q-Track real-time positioning sensor. This sensor indicates the location of the table and initiates the next module whenever the student moves to the next location. Each time they complete a module, they are then given as much time as they need to complete the quiz question for that module. They are allowed to look around the virtual space during this time to view relevant tables and figures.

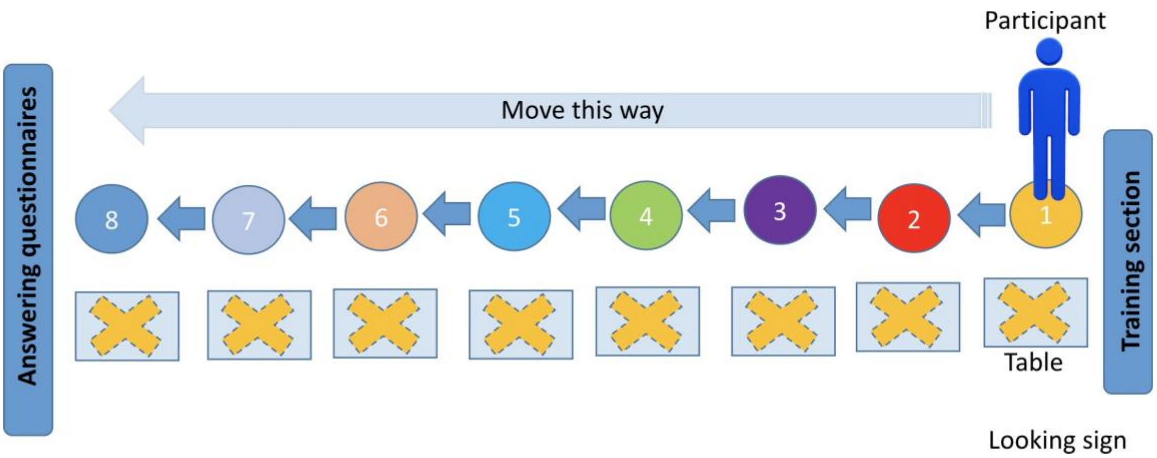

Figure 3-3: Experimental layout.

Chapter 4 - Methods

4.1 Data Preprocessing

4.1.1 Output Data Problems

There are several problems with the raw eye-tracking output data that prevent an effective analysis. The data must be preprocessed in order to address these issues. The most significant problem is that there is an inconsistent number of data points collected per second by the Microsoft HoloLens eye-tracking system. Some 1-second periods contain as many as 30 data points. Others contain few or even no data points at all. To compare different students with one another, the dataset needs to be structured, and therefore modifications must be made.

Another issue that needs to be addressed are the periods of time without any data points. Missing data points occur when the student is not looking at any of the five content panels within the virtual space. Students who look down at the table to view the quiz questions during the module are one cause of this issue. Regardless of how they occurred, the missing data points need to be accounted for and filled in with a null value so that a continuous timeline exists.

Finally, some additional columns are included in the output data that can be excluded for the purpose of this analysis. The distance between the student and the panel, the z coordinates (which is essentially the same as the distance), which panel the student is looking at, and the date will all be removed. The remaining columns are the x-coordinates, y-coordinates, and time. Table 4-1 includes a description of each of the output dataset columns.

Column	Name	Description
V1-V3	Timestamp	These columns contain the date and time for the instant each data point was recorded.
V6	Panel	This column notes which panel the student was looking at. There are five possible panels (Left Panel, Center Left Panel, Center Panel, Center Right Panel, and Right Panel).
V8	X-coordinate	The x-coordinate values range from approximately -3 to 3, with the left side of the leftmost panel being at negative 3 and the right side of the rightmost panel at positive 3. The center point of the center panel is 0.
V9	Y-coordinate	The y-coordinate values range from approximately -1 to 1 with the bottom of the panel being at negative 1 and the top of the panel at positive 1. The vertical center of the panel is 0.
V10	Z-coordinate	The z-coordinate corresponds to the distance from the student to the panel. No distance units are specified by the program.
V12	Distance	The distance between the student and the panel (same as the z-coordinate). No distance units are specified by the program.
Others	n/a	Formatting rows which can be removed.

Table 4-1: Eye-tracking output data column descriptions.

4.1.2 Importing & Cleaning Data

The statistical analysis programming language R will be used to preprocess the data along with the integrated development environment RStudio. The raw data is imported into RStudio one module at a time. There were 33 participants who each completed 14 lecture modules, resulting in a total of 462 observations. An example of the raw data after it is imported into RStudio as a data frame is shown in Table 4-2.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
1	[10/22/2022	12:59:45	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	0.1,	3.7),	Distance:	3.647739	}
2	[10/22/2022	12:59:45	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	0.1,	3.7),	Distance:	3.629482	}
3	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	0.1,	3.7),	Distance:	3.613884	}
4	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	0.1,	3.7),	Distance:	3.602836	}
5	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.2,	0.1,	3.7),	Distance:	3.594952	}
6	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.2,	0.0,	3.7),	Distance:	3.587662	}
7	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	0.0,	3.7),	Distance:	3.583257	}
8	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.580386	}
9	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.579888	}
10	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.581107	}
11	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.582490	}
12	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.583268	}
13	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.583115	}
14	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.581873	}
15	[10/22/2022	12:59:46	PM]	{	Name:	CenterPanel,	Point:	(−0.1,	−0.1,	3.7),	Distance:	3.580450	}

Showing 1 to 15 of 9,002 entries, 13 total columns

Table 4-2: Eye-tracking output data (student 1, lecture 1, module 7).

Once imported, each dataset is cleaned and restructured. The first step is to remove unnecessary columns and rename the significant ones. After this, the time column is restructured so that it begins from 0 and only includes seconds. Finally, the average x and y-coordinate for each 1-second period are calculated. These values will be used moving forward so that there are an equal number of data points for each second. For any second which does not have any data points, the "NA" value will be used as a placeholder. The final preprocessed data is organized by lecture so that each student has two preprocessed data files. Each file contains all 7 modules which are a part of the corresponding lecture. The data preprocessing R script file is shown in Figure 4-1 along with the preprocessed data for student 1 in Table 4-3.

```r
for (f in 1:length(student.data.files)) {
  #select the individual file to be processed
  student <- read.table(file = student.data.files[f])
  #Name important columns and remove unnecessary characters
  student <- tidyr::separate(data=student, col=V8, sep=c(1,-1),
                             into=c(NA,"xco",NA))
  student <- tidyr::separate(data=student, col=V9, sep=-1,
                             into=c("yco",NA))
  #Create a new time column which starts from 0
  student$V2 <- as.POSIXct(student$V2, format="%H:%M:%OS")
  student$time <- student$V2 - student$V2[1]
  #Remove all columns except time, xco, and yco
  student <- dplyr::select(student, c(14,8,9))
  #Convert columns to numeric values
  student$xco <- as.numeric(student$xco)
  student$yco <- as.numeric(student$yco)
  student$time <- as.numeric(student$time)

  #Convert x and y coordinate data to 1-second averages
  xavg <- rep(NA, max(student$time)+1)
  yavg <- rep(NA, max(student$time)+1)
  for (t in 0:max(student$time)) {
    xavg[t+1] <- mean(student$xco[which(student$time==t)])
    yavg[t+1] <- mean(student$yco[which(student$time==t)])
  }
```

Figure 4-1: Data preprocessing R script file.

	xco1	yco1	xco2	yco2	xco3	yco3	xco4	yco4	xco5
1	0.093103448	−0.15172414	0.09285714	0.000000000	0.00000000	0.000000000	−0.003703704	0.01111111	−0.17037037
2	0.100000000	−0.10666667	0.08518519	0.033333333	−0.04400000	−0.276000000	−0.283333333	0.20666667	−1.15333333
3	0.100000000	−0.10000000	0.05000000	0.100000000	0.02222222	−0.070370370	0.110714286	−0.43571429	−1.37333333
4	0.100000000	−0.10000000	0.15517241	0.127586207	0.02333333	0.023333333	1.903333333	0.36666667	−0.35333333
5	0.100000000	−0.10000000	−0.06428571	−0.435714286	0.08333333	−0.003333333	2.600000000	0.63666667	−1.36666667
6	0.090000000	−0.11666667	−0.10000000	0.025000000	0.00000000	0.003333333	2.206666667	0.47000000	−2.86000000
7	0.293333333	−0.16333333	−0.10000000	0.010000000	0.00000000	0.023333333	−0.575862069	0.28965517	−2.81666667
8	0.133333333	−0.18000000	−0.10666667	0.100000000	0.03000000	0.030000000	−0.596666667	0.33666667	−2.50740741
9	0.100000000	−0.20000000	−0.10666667	0.720000000	0.27333333	−0.486666667	−0.300000000	−0.80000000	−0.34666667
10	0.100000000	−0.20000000	−0.33000000	0.840000000	0.05666667	−0.196666667	−0.736842105	0.26842105	−0.74666667
11	−0.140000000	−0.20000000	−0.51333333	0.796666667	0.08333333	0.053333333	−0.560000000	−0.24000000	−0.28666667
12	0.026666667	−0.20000000	−0.15000000	0.453333333	1.45333333	0.300000000	−0.688235294	0.10588235	1.90000000
13	0.036666667	−0.20000000	−0.22000000	0.426666667	2.56333333	0.476666667	−0.537500000	0.30000000	2.68000000
14	0.100000000	−0.20000000	0.13333333	0.396666667	2.95000000	0.290000000	NA	NA	2.66666667
15	0.100000000	−0.20000000	0.07500000	0.141666667	2.70000000	0.400000000	NA	NA	0.15333333

Showing 1 to 14 of 501 entries, 14 total columns

Table 4-3: Preprocessed student data (student 1, lecture 1).

4.2 Student Data

Before analyzing the eye-tracking data, the dataset must first be checked to determine whether there appears to be any trends within the data. Regression models were trained using eye-tracking data to predict whether students will be more or less likely to answer questions correctly. In order for this type of analysis to be effective, it is necessary to understand the different trends present in the answer correctness data so that they can be accounted for during the regression analysis. Figure 4-2 displays the proportion of correct versus incorrect answers for each individual student. Students 13 and 32 both had missing answer data and are excluded from analysis. Students answered a majority of questions correctly with an overall accuracy of 87.4%. Four students answered every single question correctly. This is not ideal because in order to train a regression model there needs to be a sufficient amount of data with both possible outputs. Since a majority of students were correct, it is more difficult for a model to identify factors which accurately predict answer correctness. Besides the lack of answer disparity, the answer data appears to be normal and there don't seem to be any trends between students.

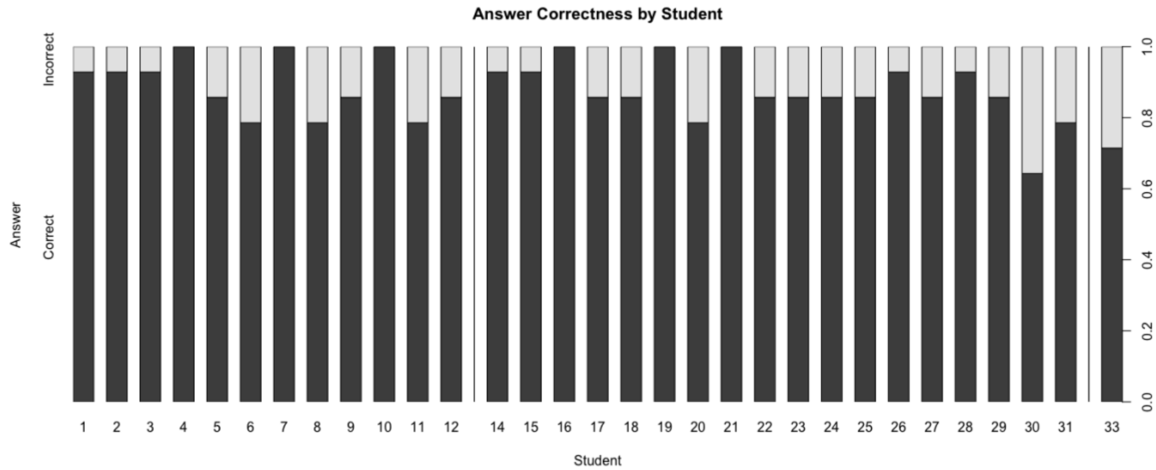

Figure 4-2: Answer correctness by student.

The next comparison is the effect of the different modules on student answer correctness. In Figure 4-3, each module is plotted in order versus the proportion of students who answered the follow-up quiz question correctly. Upon a visual analysis, it appears that there is a negative correlation between student answer correctness and the lecture modules. This correlation happens to be intentional as the difficulty level of the second lecture was increased with respect to the first. This trend needs to be accounted for in order to create a model that is capable of accurately predicting student answer correctness.

Another important feature of the answer data is that there are four modules (1-1, 1-3, 1-7, and 2-1) in which all students answered questions correctly. This is an important fact to consider when conducting a logistic regression analysis to predict the likelihood of a correct vs incorrect answer. It can be very difficult for a regression model to account for factors which result in a 100% likelihood of a certain outcome. In some cases, it is beneficial to treat such data as outliers and remove them from the dataset when training the model.

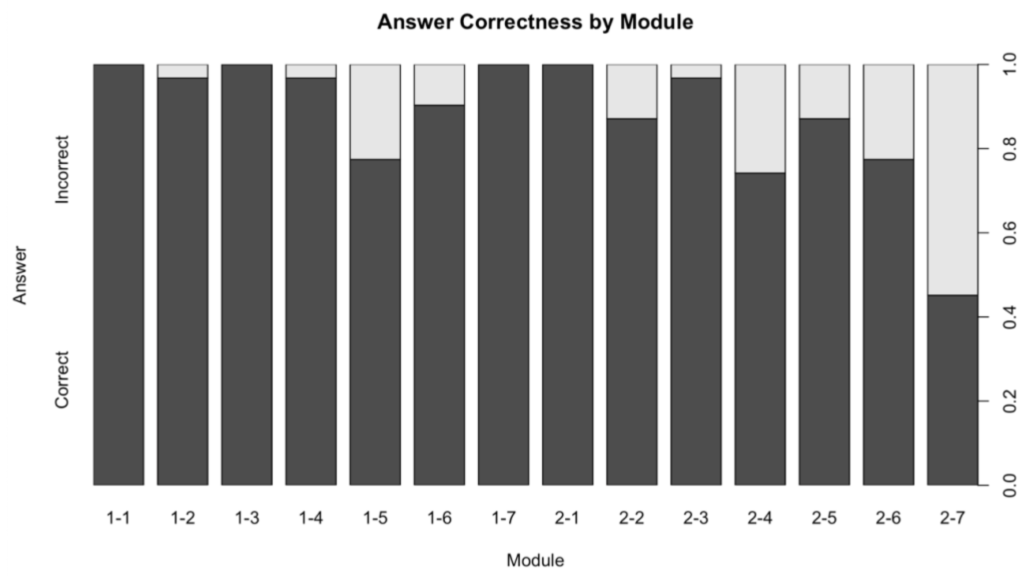

Figure 4-3: Answer correctness by module.

4.3 Baseline Data Collection

Initially, the use of a machine learning algorithm was considered as a method of predicting student answer correctness. Unfortunately, due to the uneven split between student answer correctness, this type of model would be ineffective. Instead, the alternate approach chosen for this analysis begins with identifying the ideal eye-tracking coordinates throughout the lecture material. This set of ideal coordinates, referred to as the baseline dataset, closely follows the virtual instructor's location and only deviates when specifically told to do so. In order to record this dataset, an additional participant was trained on where the different objects of interest can be found throughout each module. Multiple trial runs were recorded for each module until a satisfactory baseline dataset was collected. The baseline dataset can be used as a comparison tool to detect how closely each student followed the virtual instructor. It is hypothesized that students who accurately follow the instructions of the virtual instructor will be more likely to discern key information and answer follow-up questions correctly.

Before creating an algorithm to compare the student and baseline datasets, they can be compared using a coordinate plot to search for trends. Based on the hypothesis, students whose eye-tracking coordinates are closer to the baseline coordinates will answer questions correctly. In contrast, students whose eye-tracking coordinates do not match the baseline should answer incorrectly. An example of the first case is shown in Figures 4-4 and 4-5. These timeseries plots show the x and y-coordinates for student 2 compared to the baseline coordinates in module 2-2. In both plots, the student appears to accurately follow the baseline coordinates, excluding a few brief deviations. In this case, the student answered the question correctly, which agrees with the proposed hypothesis that students who accurately follow the baseline dataset will have a greater likelihood of answering questions correctly.

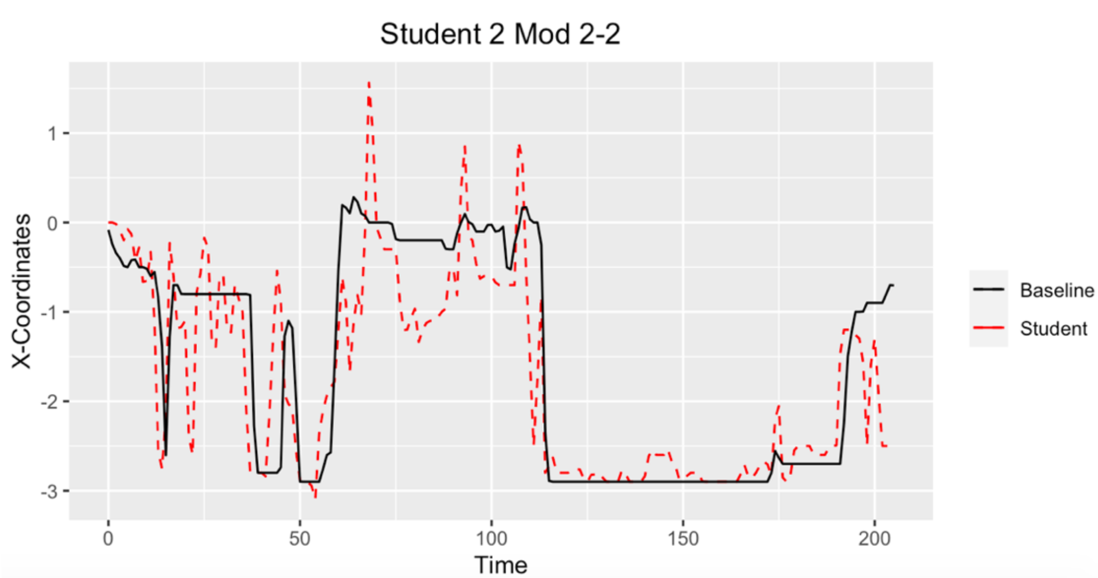

Figure 4-4: X-coordinates vs baseline (student 2, module 2-2).

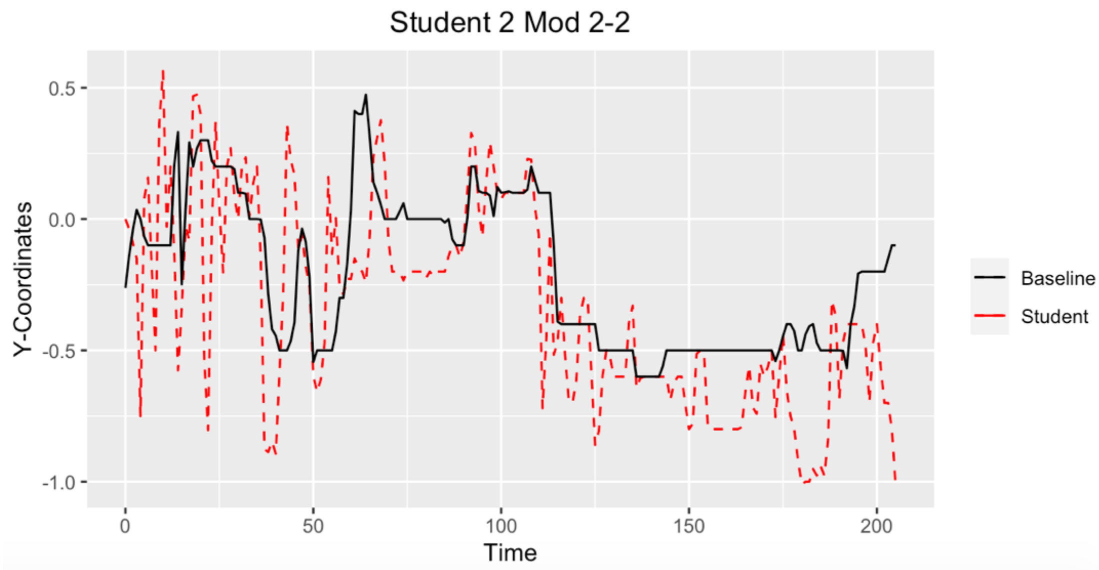

Figure 4-5: Y-coordinates vs baseline (student 2, module 2-2).

In Figures 4-6 and 4-7, the plots show the x and y-coordinates of student 2 compared to the baseline for module 2-7. In this case, the student was not able to accurately follow the baseline and the student answered the question incorrectly. Once again, the proposed hypothesis was correct. These two examples are useful for visualizing the baseline comparisons, although they do not provide nearly enough evidence to draw any conclusions. The rest of the data will need to be analyzed as well in order to determine whether there is a meaningful relationship between how accurately the student follows the baseline dataset and the probability of a correct answer. Additionally, rather than visually analyzing the data for each individual module, comparison metrics will need to be developed to numerically represent how accurately the student follows the baseline. Two metrics are developed in order to accomplish this: average difference and distraction rate. The following sections describe how they are calculated and how they can be used to predict student performance.

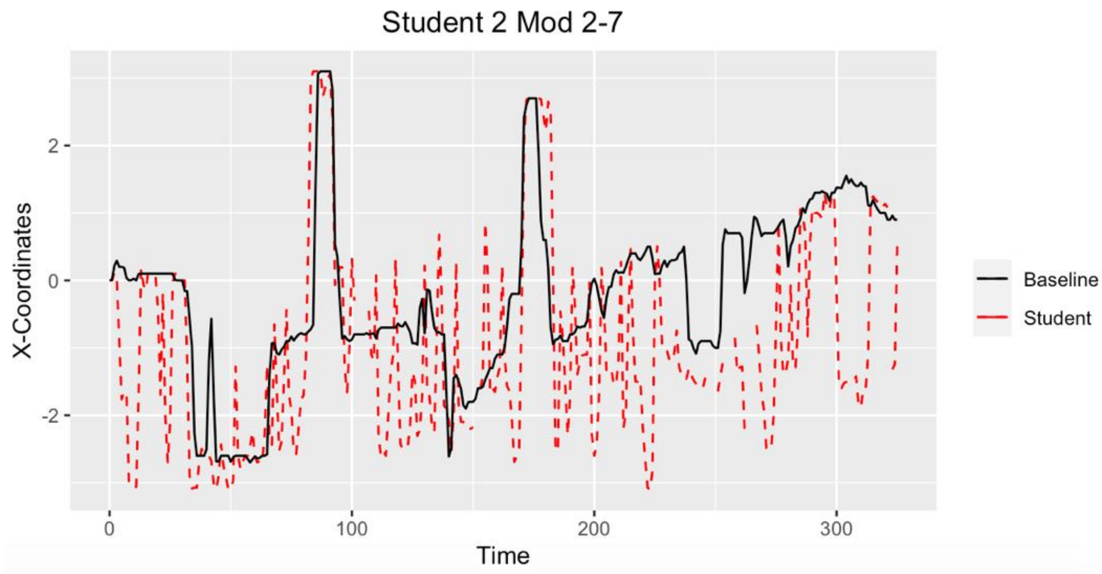

Figure 4-6: X-coordinates vs baseline (student 2, module 2-7).

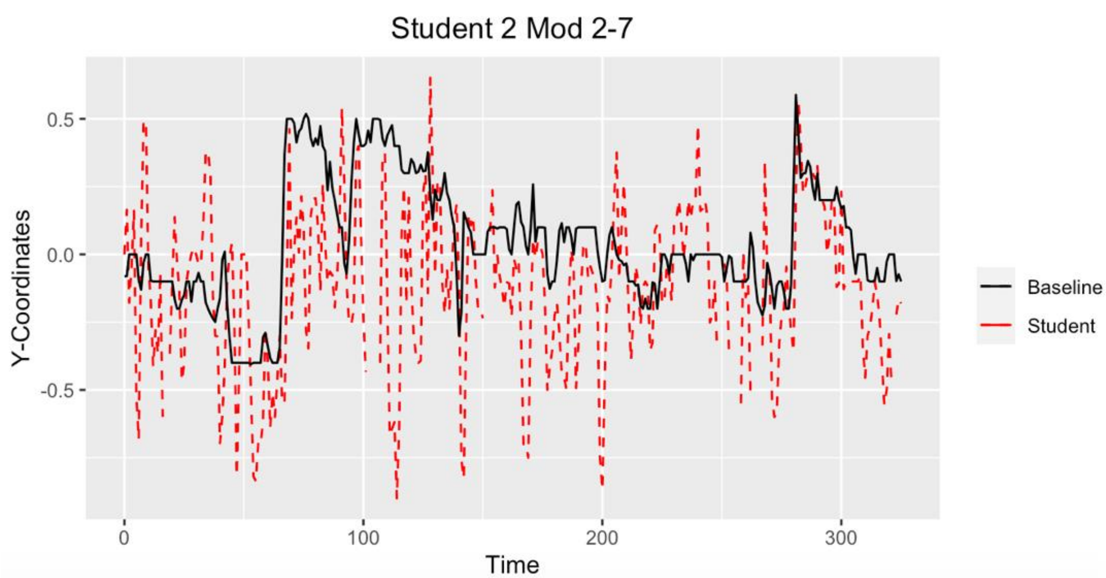

Figure 4-7: Y-coordinates vs baseline (student 2, module 2-7).

4.4 Average Difference Method

4.4.1 Methodology

The average difference method is fairly straightforward. There will be one value calculated for each module which represents the students' attention level. The first step is to calculate the difference between the student and baseline coordinates for each 1-second interval. Next, add together the difference from each 1-second interval to get the total difference across the entire module. Finally, divide the sum by the total number of observations (equivalent to the runtime minus the number of missing values) to normalize the data. The resulting value is the average difference for that module. This metric should indicate how well the student followed the instructions of the virtual instructor throughout each module. The average x-coordinate difference will first be considered. The reason for this decision is that the virtual content within the AR learning environment covers a much greater horizontal distance since the five content panels are positioned side-to-side. Because of this, the x-coordinate data will provide a greater insight as to whether or not the student is focusing on the correct locations.

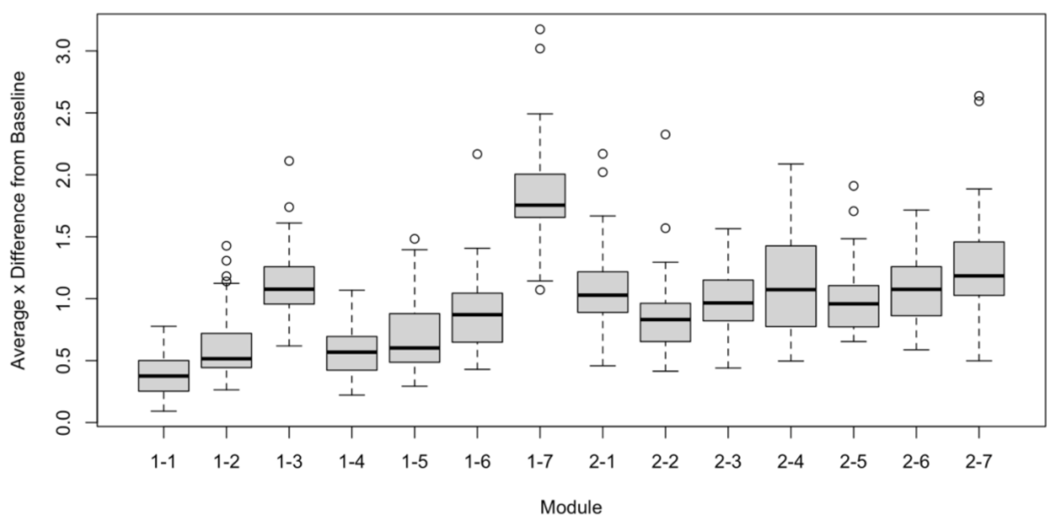

Figure 4-8: Average x-coordinate difference vs module.

The statistical analysis programming language R will once again be used to conduct this analysis through the integrated development environment RStudio. The first step will be to import the preprocessed student and baseline data files for each lecture. For each student, one module will be processed at a time and its average difference will be calculated. Since there are 33 students who each completed 14 modules, there will be a total of 462 observations. The R script file which is used to calculate the average x-coordinate difference for lecture 1 is shown in Figure 4-9. The resulting data is also plotted based on module in Figure 4-8. This chart helps visualize how difficult it was for students to follow the virtual instructor in each module.

```r
#calculate the average euclidean difference between baseline and student data for each student
library(gtools)

#import baseline data and student data file list
base.df <- data.frame(read.csv("../ETDA Project/Processed Baseline Data/Lecture 1.csv"))
base.df <- base.df[-c(1:2,4,6,8,10,12,14,16)]
student.data.files <- list.files("../ETDA Project/Processed Student Data/Lecture 1",
                                 full.names=T, include.dirs=T)
student.data.files <- mixedsort(student.data.files)

#create an empty data frame to store the x coordinate differences between the student and baseline data
x.diff.df <- data.frame(matrix(nrow = length(base.df), ncol = length(student.data.files)))
rownames(x.diff.df) <- c("Lecture 1-1","Lecture 1-2","Lecture 1-3","Lecture 1-4",
                         "Lecture 1-5","Lecture 1-6","Lecture 1-7")
colnames(x.diff.df) <- paste("Student", 1:33, sep = "_")

for (student in 1:length(student.data.files)) {
  #import the student data for student i
  std.df <- data.frame(read.csv(student.data.files[student]))
  std.df <- std.df[-c(1:2,4,6,8,10,12,14,16)]

  #create a new data frame of the x coordinate difference between the student and baseline data
  diff.df <- abs(base.df-std.df)

  #Calculate the euclidean distances and store the average value in the data frame
  for (mod in 1:(length(base.df))) {
    x.diff.df[mod,student] <- mean(diff.df[,mod], na.rm = TRUE)
  }
}
#store the x diff data frame in a csv file
write.csv(x.diff.df, file = "../ETDA Project/Sum of Differences Analysis/Lecture 1 x Diff.csv")
```

Figure 4-9: Average x-coordinate difference R script.

4.4.2 Statistical Analysis

The first statistical test conducted is a two-sample, one-sided, equal-variance, t-test comparing the average x-coordinate difference from modules which students answered correctly to modules which students answered incorrectly. With a p-value of 0.02226, the results from this test show that there is a significant increase in average x difference between the two populations. This indicates a correlation between average x-coordinate difference and student answer correctness. The data is plotted in Figure 4-10, where it becomes apparent that there isn’t a substantial difference between the two populations despite the results of the t-test.

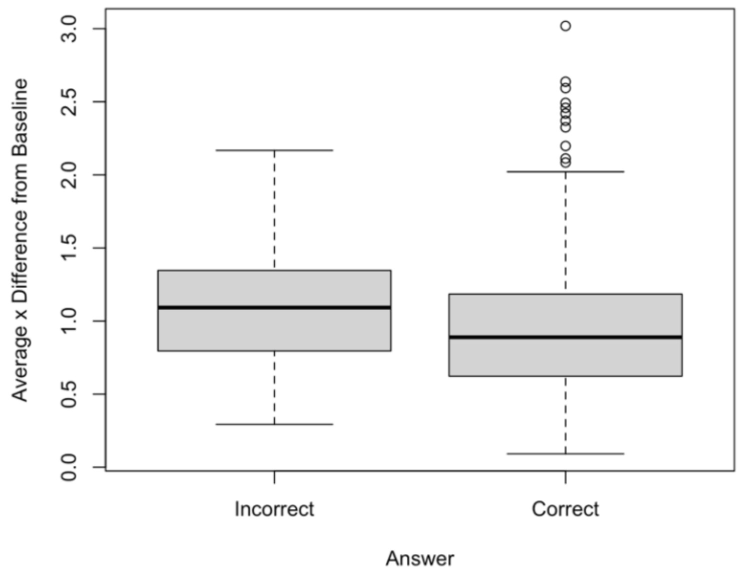

Figure 4-10: Average x-coordinate difference of modules with correct vs incorrect answers.

For the next part of the statistical analysis, the average x-coordinate difference will be fitted as a predictor of student answer correctness in a simple logistic regression model. This model does not consider any other factors. In the resulting model, average x-coordinate difference has a parameter estimate of -0.5745 with a p-value of 0.0463. Based on this analysis, average x-coordinate difference can be considered a significant predictor of student answer correctness. The negative parameter estimate also indicates that as the average x-coordinate difference increases, the probability of a correct answer decreases. In other words, if the student does not accurately follow the virtual instructor, then they will have a lower chance of correctly answering the question for that module. The resulting model is plotted in Figure 4-11. The plot also includes all observations of student answer correctness vs average x-coordinate difference.

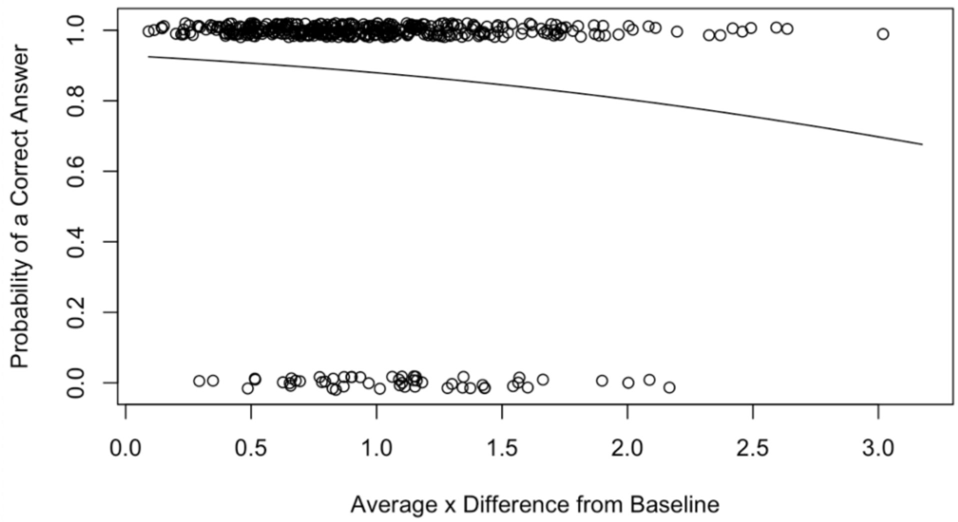

Figure 4-11: Average x-coordinate difference simple logistic regression model.

The final statistical test will be a mixed-effects logistic regression model. In this model, average x-coordinate difference will be considered as a factor in addition to module (fixed factor) and student (random factor). By including these factors in the model, the variation they cause within the dataset will be accounted for. This should result in a prediction model which is more accurate at predicting student answer correctness than the simple logistic regression model. Two different logistic regression functions will be used, the glmmPQL function and the glmer function. The glmmPQL function utilizes penalized quasi-likelihood, which is a flexible and widely implemented parameter estimation method. The glmer function instead uses the Laplace approximation method, which tends to be more accurate than PQL, but also slower and less flexible (Bolker et al., 2009).

The glmmPQL model resulted in a parameter estimate of -0.8973 for average x-coordinate difference. The parameter estimate has a p-value of 0.0393, which is significant. For the glmer model, average x-coordinate difference has a parameter estimate of -0.5906 and a p-value of 0.226. The algorithm also failed to converge, which suggests that the prediction variables may not convey enough information to accurately predict the outcome variable. There could also be issues with the algorithm's parameters which could potentially be adjusted to resolve this issue. Regardless, both algorithms resulted in negative parameter estimates and the glmmPQL parameter estimate was significant. When considering modules and students as sources of variation, average x-coordinate difference appears to have a negative correlation with answer correctness, albeit without an overwhelming significance level.

4.4.3 Remove Models with 100% Accuracy

One of the main concerns noted earlier about the dataset is its lack of answer disparity. Students answered a majority of the questions correctly which makes it difficult for a logistic regression model to make accurate predictions. For this reason, the same statistical analysis will be repeated after removing all modules containing 100% answer correctness from the dataset (modules 1-1, 1-3, 1-7, and 2-1). This could potentially increase the significance of average x-coordinate difference as a predictor of answer correctness. It should also improve the accuracy of any predictive models. The updated average x-coordinate difference vs module chart is shown in Figure 4-12.

The results of the t-test showed a large increase in significance compared to the analysis which included the modules with 100% accuracy. The p-value increased from 0.02226 to 0.0001757. This is a much more conclusive result, showing that modules in which students answered questions correctly have a significantly lower average difference than modules with incorrect answers. The average x-coordinate difference for modules with correct vs incorrect answer excluding modules with 100% accuracy is plotted in Figure 4-13.

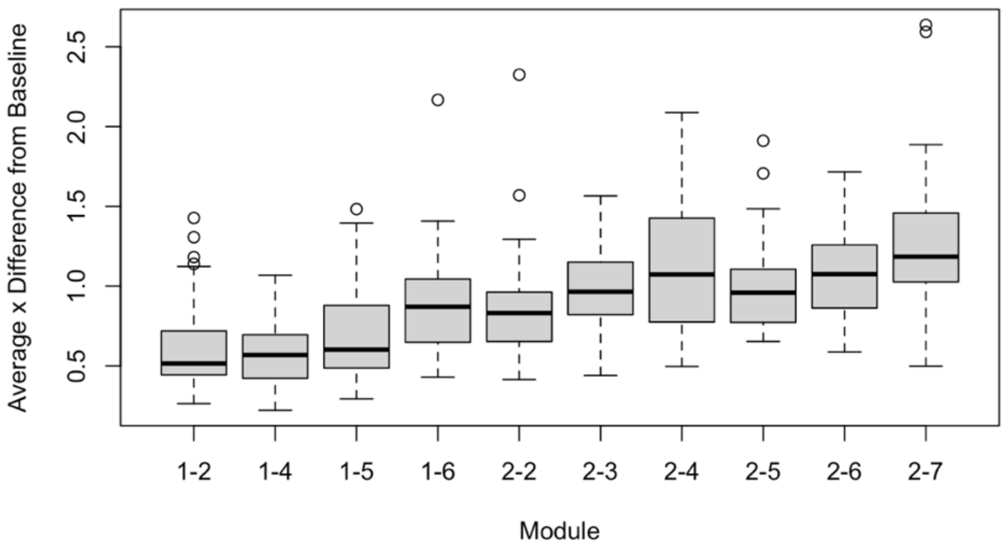

Figure 4-12: Average x-coordinate difference vs module (excluding modules with 100% accuracy).

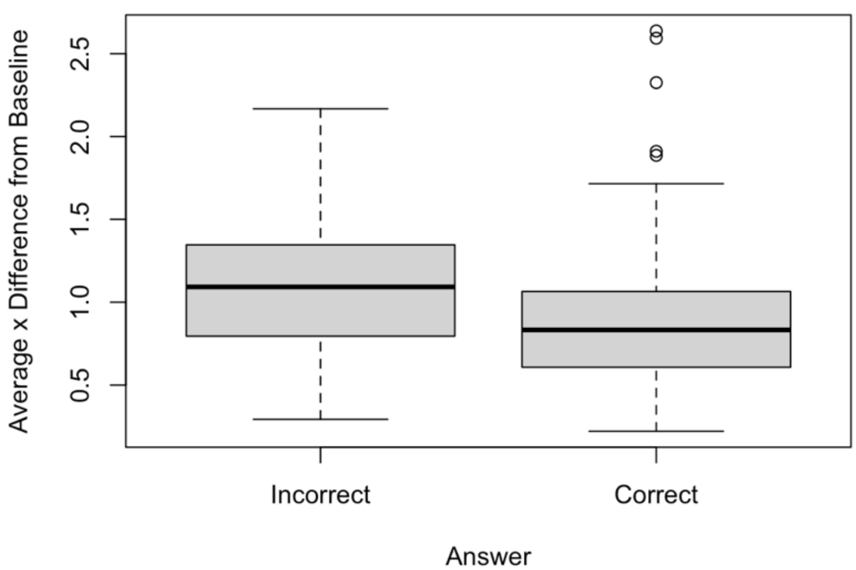

Figure 4-13: Average x-coordinate difference of modules with correct vs incorrect answers (excluding modules with 100% accuracy).

The simple logistic regression model saw a large increase in significance as well with the p-value increasing from 0.0463 to 0.000685. Additionally, the parameter estimate for average x-coordinate difference was -1.2209, which is more than double the magnitude of the original analysis (-0.5745). By removing the modules with 100% accuracy, the predictive power of the model has increased significantly. The model is plotted below in Figure 4-14.

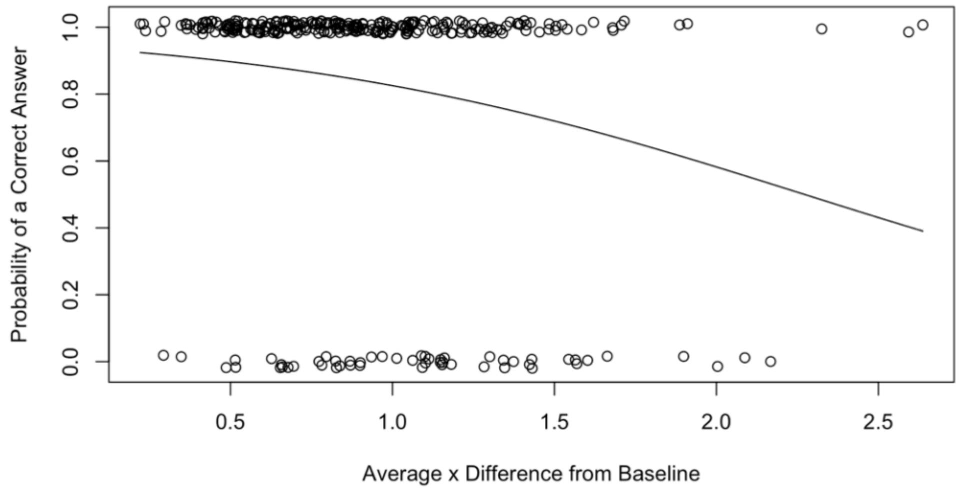

Figure 4-14: Average x-coordinate difference simple logistic regression model (excluding modules with 100% accuracy).

After having removed the modules with 100% answer correctness, there appears to have been a negative effect on the mixed effects models. In the case of the glmmPQL model, the parameter estimate for average x difference was -0.6728 with a p-value of 0.1487. This is a large decrease in significance from the model which included every module. The glmer model failed to converge as it did previously, but this time it produced a model that appears to be problematic as all parameter values have concerningly low p-values ($2*10^{-16}$). For this reason, the results of this model will be ignored (see the appendix for model output).

4.5 Average Euclidean Distance Method

4.5.1 Methodology

Rather than only including x-coordinate data, the x and y-coordinates can be considered simultaneously by calculating the Euclidean distance between the student and baseline coordinates for each 1-second interval. The process of computing the average Euclidean distance is similar to how the average x-coordinate difference was calculated. First, the difference of both the x and y-coordinates are calculated. Next, these two values squared, combined, and then square rooted. At this point, the rest of the calculation is the same as before. The resulting metric is the average Euclidean distance between the student and baseline. The resulting data is plotted based on module in Figure 4-16. The R script file used to conduct these calculations for lecture 1 is shown in Figure 4-15.

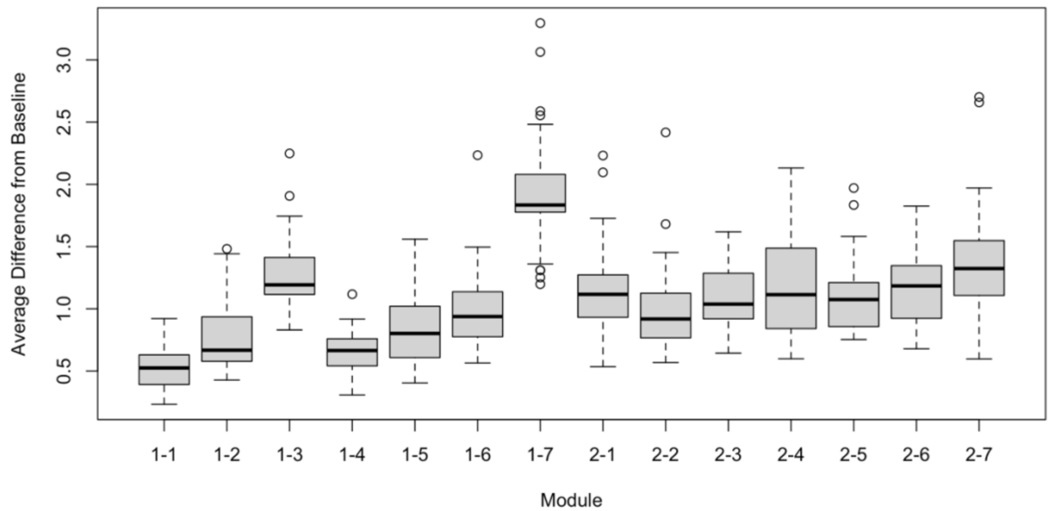

Figure 4-15: Average Euclidean difference vs module.

```r
#calculate the average euclidean difference between baseline and student data for each student
library(gtools)

#import baseline data and student data file list
base.df <- data.frame(read.csv("../ETDA Project/Processed Baseline Data/Lecture 1.csv"))
base.df <- base.df[-c(1:2)]
student.data.files <- list.files("../ETDA Project/Processed Student Data/Lecture 1",
                                 full.names=T, include.dirs=T)
student.data.files <- mixedsort(student.data.files)

#create an empty data frame to store the average differences between the student and baseline data
avg.diff.df <- data.frame(matrix(nrow = length(base.df)/2,
                                 ncol = length(student.data.files)))
rownames(avg.diff.df) <- c("Lecture 1-1","Lecture 1-2","Lecture 1-3","Lecture 1-4",
                           "Lecture 1-5","Lecture 1-6","Lecture 1-7")
colnames(avg.diff.df) <- paste("Student", 1:33, sep = "_")

for (student in 1:length(student.data.files)) {
  #import the student data for student i
  std.df <- data.frame(read.csv(student.data.files[student]))
  std.df <- std.df[-c(1:2)]

  #create a new data frame of the squared difference between the student and baseline data
  diff.df <- abs(base.df-std.df)
  diff.df <- diff.df^2

  #Calculate the euclidean distances and store the average value in the data frame
  for (mod in 1:(length(base.df)/2)) {
    diff.df[,(length(base.df)+mod)] <- diff.df[,2*mod-1] + diff.df[,2*mod]
    diff.df[,(length(base.df)+mod)] <- sqrt(diff.df[,(length(base.df)+mod)])
    avg.diff.df[mod,student] <- mean(diff.df[,(length(base.df)+mod)], na.rm = TRUE)
  }
}
#store the avg diff data frame in a csv file
write.csv(avg.diff.df, file = "../ETDA Project/Sum of Differences Analysis/Lecture 1 Avg Diff.csv")
```

Figure 4-16: Average Euclidean difference R script.

4.5.2 Statistical Analysis

The same t-test used for the average x-coordinate difference can also be applied to the average Euclidean distance. In this case, the resulting p-value was 0.03226 which is significant, although slightly worse than the average x-coordinate difference. This test indicates that there is a potential correlation between average Euclidean distance and student answer correctness. The data is plotted in figure 4-17, where it is once again apparent that there isn't a large difference between the two populations.

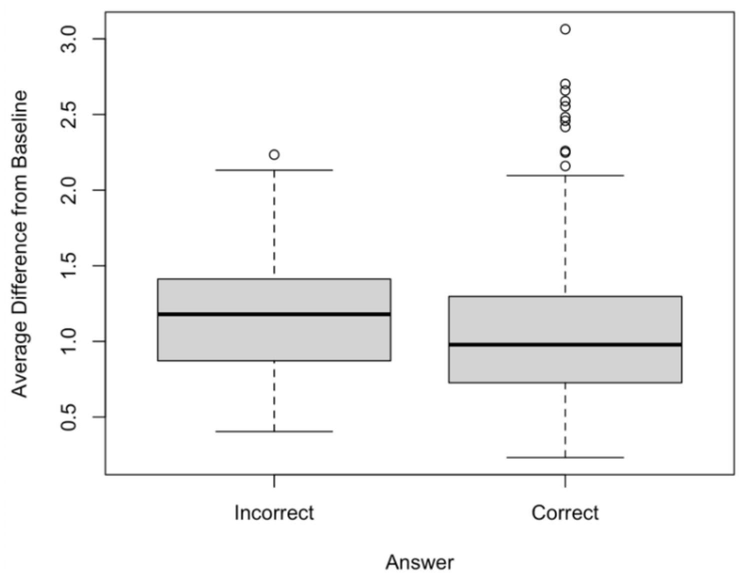

Figure 4-17: Average Euclidean difference of modules with correct vs incorrect answers.

Once again, a simple logistic regression model is fitted with average Euclidean distance as a predictor of student answer correctness. The resulting parameter estimate was -0.5431 with a p-value of 0.0663. This is not a significant p-value, although it is close. Also, the parameter estimate is negative, indicating a negative correlation between average Euclidean distance and student answer correctness. The model is plotted in Figure 4-18 with all observations of student answer correctness vs average Euclidean distance included as well.

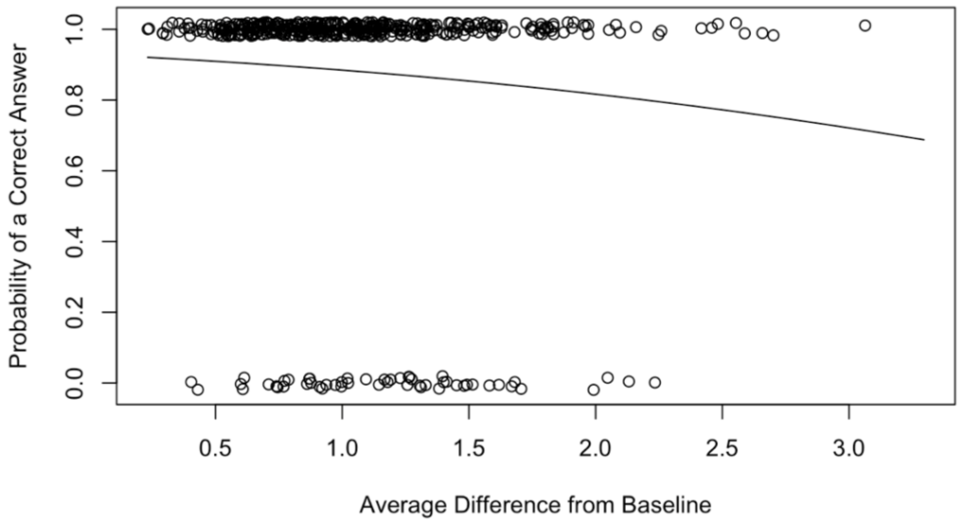

Figure 4-18: Average Euclidean difference simple logistic regression model.

The two mixed-effect logistic regression models will now be fitted to the average Euclidean difference data to determine whether it can be used as a significant predictor when considering modules and students as factors as well. The glmmPQL algorithm resulted in a parameter estimate of -0.8594 for average Euclidean distance with a p-value of 0.0518. The glmer algorithm produced a parameter estimate of -0.5604 with a p-value of 0.258. The glmer algorithm once again failed to converge. Neither of the parameter estimates were significant, although the glmmPQL model's parameter estimate was quite close. Both parameter estimates were also negative, which is indicative of a negative correlation between average Euclidean distance and student answer correctness.

4.5.3 Remove Modules with 100% Accuracy

Modules with 100% answer correctness will now be removed from the dataset and the statistical analysis will be repeated. The updated average Euclidean difference vs module chart is shown in Figure 4-19.

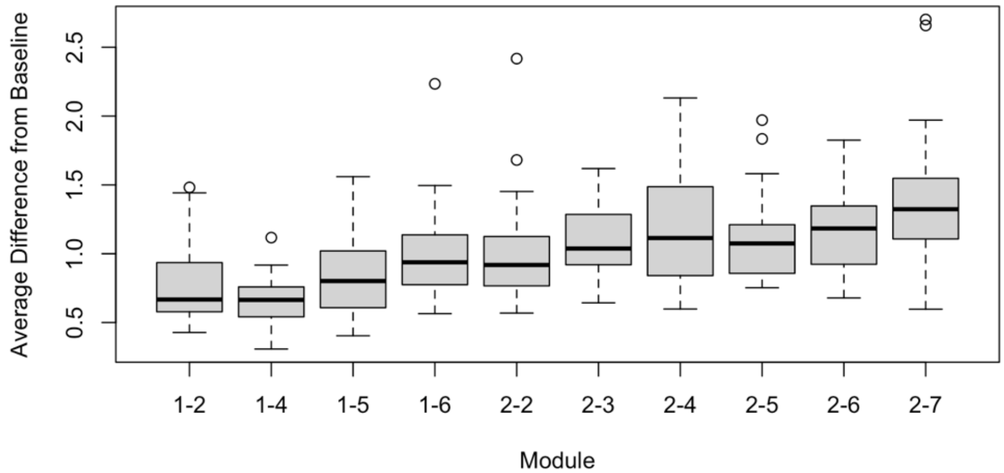

Figure 4-19: Average Euclidean difference vs module (excluding modules with 100% accuracy).

The results of the t-test were once again improved after removing modules with 100% accuracy. In this case, the p-value was decreased from 0.03226 to 0.0002757. The average Euclidean difference for modules with correct vs incorrect answer excluding modules with 100% accuracy is plotted in Figure 4-20.

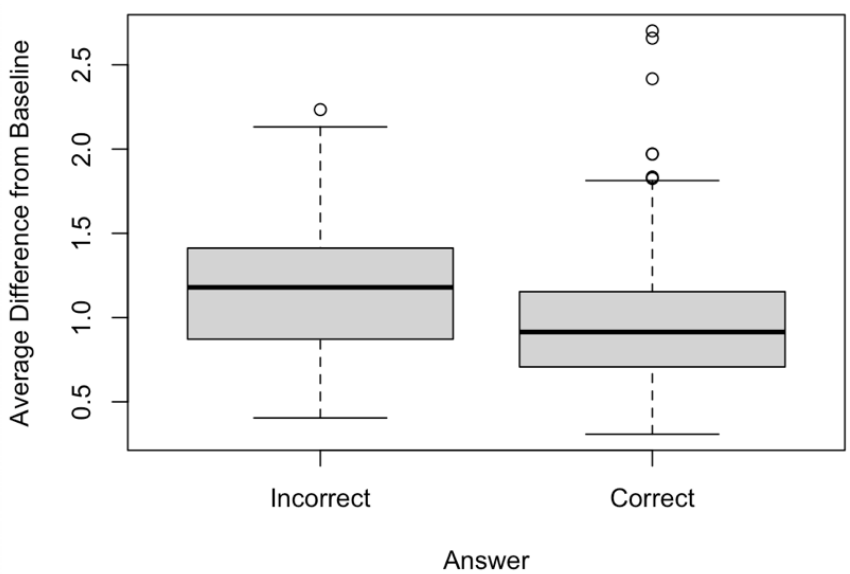

Figure 4-20: Average Euclidean difference of modules with correct vs incorrect answers (excluding modules with 100% accuracy).

The simple logistic regression model increased in significance as well, with the p-value decreasing from 0.0663 to 0.000993. The parameter estimates for average difference also increased in magnitude, changing from -0.5431 to -1.2047. The simple logistic regression model is plotted in Figure 4-21. It is clear that in the case of average Euclidean distance, removing modules with 100% accuracy significantly increases the significance of answer correctness prediction models.

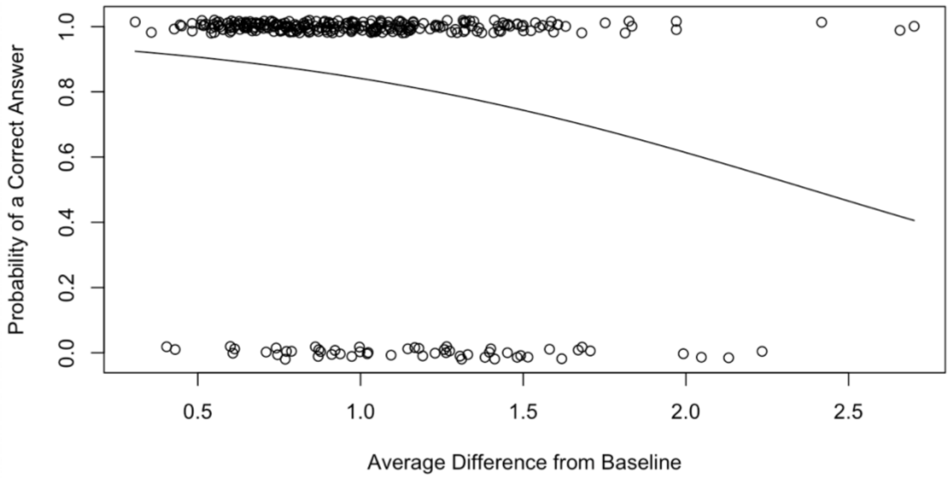

Figure 4-21: Average Euclidean difference simple logistic regression model (excluding modules with 100% accuracy).

The results of the mixed effects models were not significant. In the glmmPQL model, average Euclidean distance had a parameter estimate of -0.6355 with a p-value of 0.1788. The glmer model results appeared to be inconclusive once again. The module produced a similar output to when the average x coordinate metric was tested after removing all modules with 100% accuracy. The results will once again be omitted since they do not appear to have any sort of significance (see appendix for model output).

4.6 Distraction Rate Method

4.6.1 Methodology

The Average Difference method has shown a correlation with student answer correctness but lacks the desired effectiveness as a factor in prediction models. The reason for this is the significant number of false signals which are detected by the average difference method. This is partially due to the nature of the Microsoft HoloLens eye-tracking system. Rather than tracking pupil movements, the system only records the orientation of the headset. In theory, students could turn their eyes without reorienting their head which leads to a discrepancy between the eye-tracking coordinates and the student's actual point of attention. For this reason, it can be reasonably assumed that if the baseline coordinates are within a certain range of the student's coordinates, then the student is paying attention.

Another noise factor which is not accounted for by the average difference method are deviations from the baseline which only last for a short amount of time. There are several reasons why students may need to briefly look away from the virtual instructor. The student could decide to check one of the data tables to see where a certain value came from, or they could be taking a look at one of the animated figures. These objects within the virtual space are the exact reason why AR learning environments have an advantage over in-person learning. Students should be encouraged to look at these virtual objects throughout the lecture, and it is certainly feasible that a student could take a quick look at one of them while still paying attention to the virtual instructor.

In order to account for these two noise factors, modifications will need to be made to the average difference method to increase the significance of the metric. First, there will be a minimum distance threshold which will need to be surpassed for any data point to be considered a significant deviation from the baseline. Additionally, rather than recording the difference between the student and baseline dataset, a binary signal will be recorded. This is intended to remove any insignificant levels of deviation. Students who have become distracted are not any more or less distracted based on how far away from the baseline they are. Instead, any observation of a student who is exhibiting signs of distraction should be counted equally. A value of 1 will indicate that the difference between the student and baseline dataset was larger than the threshold and a value of 0 indicates that the difference between the student and the baseline was not larger than the threshold. In theory, this change should increase the significance of the metric since only large deviations from the baseline will be accounted for.

Example plots which visualize these adjustments are shown in Figures 4-22 and 4-23. These plots display the Euclidean difference between the student and baseline datasets for second throughout modules 2-2 and 2-7 for student 2. Both plots include a threshold line of 1.5. Any points above this line are considered positive signals (1), and any points below the line are negative (0). In module 2-2, there were 19 signals detected and in module 2-7, there were 108.

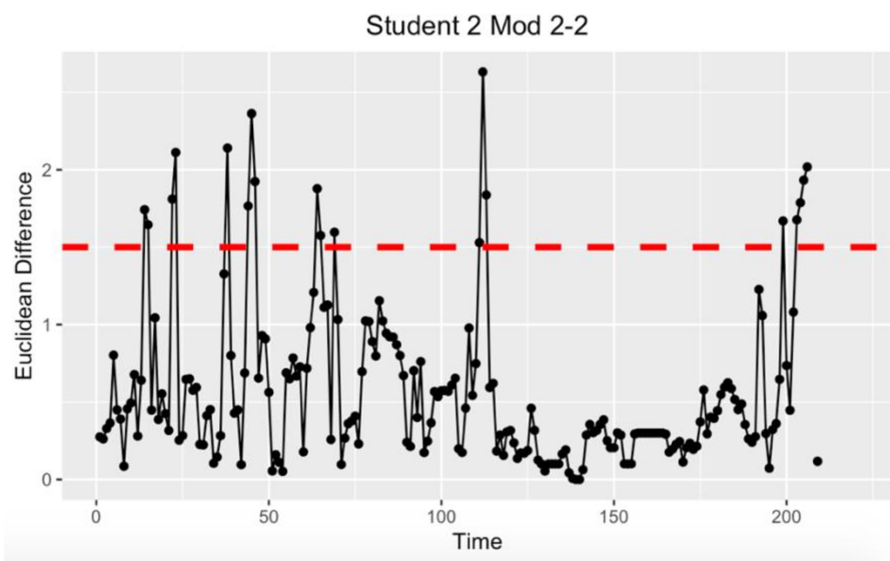

Figure 4-22: Euclidean difference compared to signal threshold of 1.5 (student 2, module 2-2).

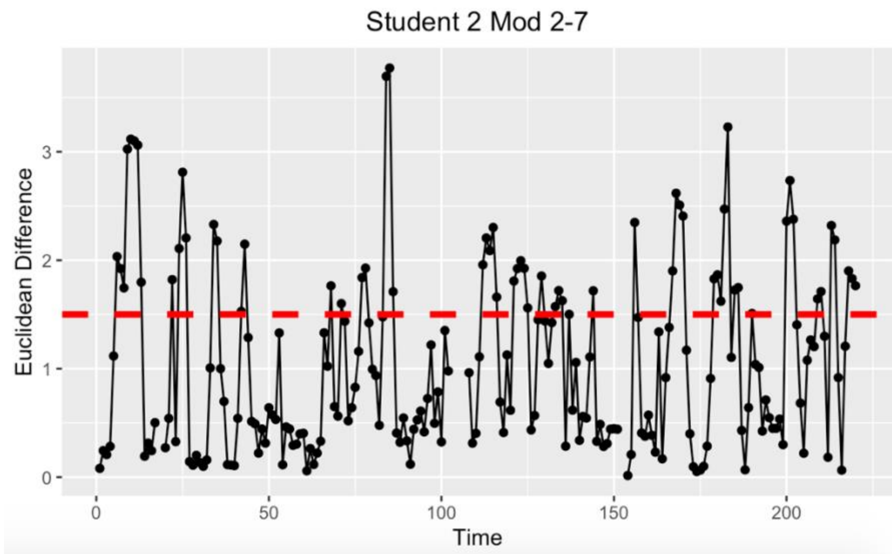

Figure 4-23: Euclidean difference compared to threshold of 1.5 (student 2, module 2-7).

The final change that will be made is to consider the moving average difference rather than the difference at each individual second. In order to compute the moving average, the Euclidean distance from the current 1-second interval will be combined with the values from the 4 previous intervals and averaged. The total amount of data points included in the average is referred to as the moving average “window”. In this case, the window is 5 seconds long. By implementing this change, signals will only be detected when the student is looking away from the virtual instructor for a significant period of time. Prolonged deviations from the baseline should be a much stronger indicator of student distraction than short-term deviations. The moving average difference from the previous examples are plotted in Figures 4-24 and 4-25. In both cases, the number of signals detected was significantly reduced as there were only 2 signals detected in module 2-2 (previously 19) and 75 detected in module 2-7 (previously 108).

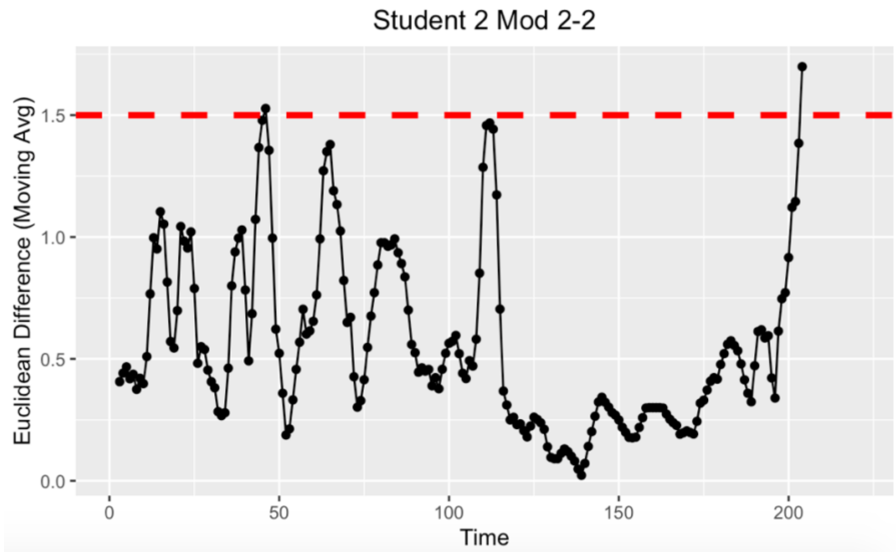

Figure 4-24: Moving average Euclidean difference compared to threshold of 1.5 (student 2, module 2-2).

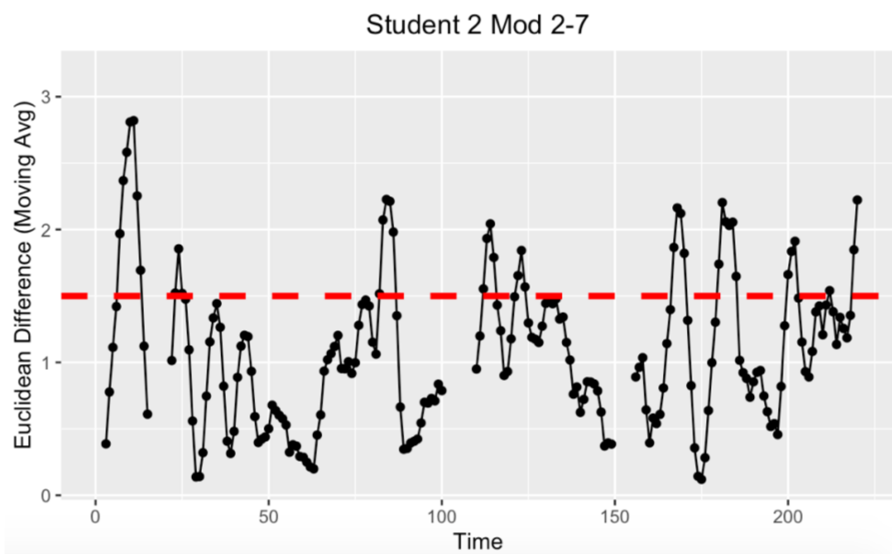

Figure 4-25: Moving average Euclidean difference compared to threshold of 1.5 (student 2, module 2-7).

Now that the number of signals per module can be calculated, this value will need to be normalized in order to account for the variation in runtime between each module. This can be done by simply dividing the number of signals detected by the total number of observations (equivalent to the runtime minus the number of missing data points). The resulting value is the proportion of the module in which the student is not accurately following the baseline coordinates. This value can be referred to as the Distraction Rate.

The distraction rate vs module comparison chart can be found in Figure 4-26. The R script file used to compute this value for each module is also shown in Figure 4-27.

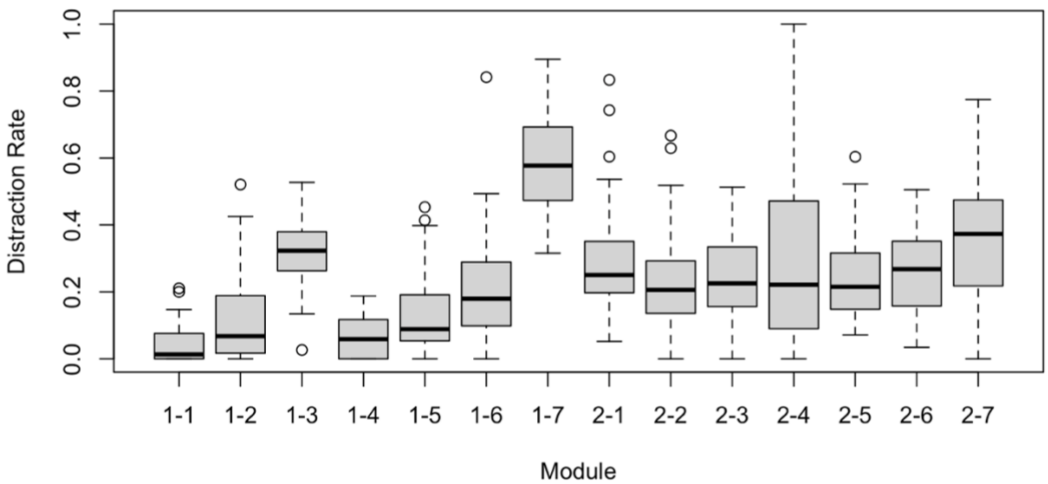

Figure 4-26: Distraction rate vs module.

```r
#create an empty data frame to store the number of signals detected in each lecture
signal <- data.frame(matrix(0,nrow = length(base.df)/2, ncol = length(student.data.files)))
cols <- c("1-1","1-2","1-3","1-4","1-5","1-6","1-7")
rownames(signal) <- cols
colnames(signal) <- rep(1:33)

for (student in 1:length(student.data.files)) {
  #import the student data for student i
  std.df <- data.frame(read.csv(student.data.files[student]))
  std.df <- std.df[-c(1:2)]

  #create a new data frame of the squared difference between the student and baseline data
  diff.df <- abs(base.df-std.df)
  diff.df <- diff.df^2

  #Calculate the euclidean distances between the student and baseline data
  for (mod in 1:(length(base.df)/2)) {
    diff.df[,(length(base.df)+mod)] <- diff.df[,2*mod-1] + diff.df[,2*mod]
    diff.df[,(length(base.df)+mod)] <- sqrt(diff.df[,(length(base.df)+mod)])
  }

  #remove separated x and y data
  diff.df <- diff.df[-c(1:14)]
  colnames(diff.df) <- cols

  #create a new column for each lecture that is the moving average with a window size of k
  for (col in cols) {
    diff.df[[paste0(col, "_ma")]] <- rollmean(diff.df[[col]], k = a, fill = NA)
  }

  #calculate the probability of the moving average being greater than 1.5 throughout each module
  for (col in cols) {
    signal[col,student] <- mean(diff.df[[paste0(col, "_ma")]] > b, na.rm = TRUE)
  }
}
```

Figure 4-27: Distraction rate R script

4.6.2 Statistical Analysis

The same statistical tests which were used to analyze the average different metrics can also be used to gauge how well distraction rate can predict student performance. The first statistical test will be the t-test. Here, the distraction rate for modules that students answered correctly will be compared to the distraction rate for modules that were answered incorrectly. The resulting p-value from this test is 0.003508, which is more significant than the average difference metrics when modules with 100% accuracy are included. The distraction rates for modules with correct vs incorrect answers are plotted in Figure 4-28.

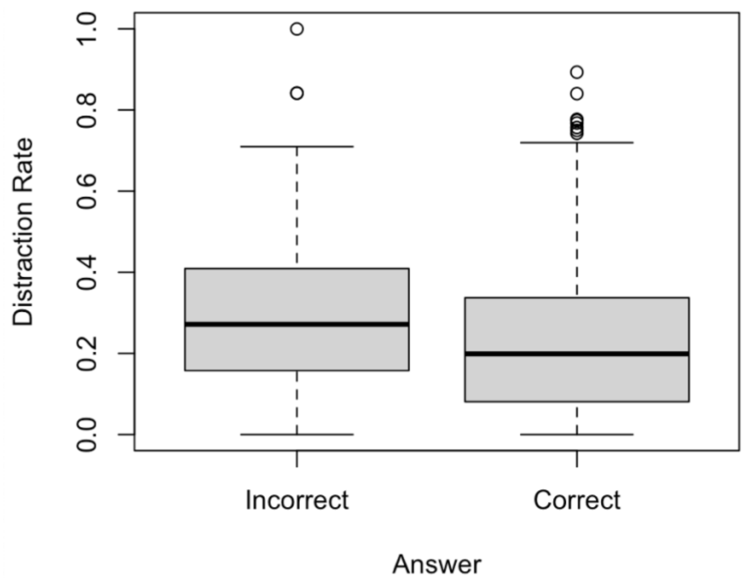

Figure 4-28: Average distraction rate of modules with correct vs incorrect answers.

The simple logistic regression model can also be fitted using distraction rate as a predictor of answer correctness. The resulting parameter estimate is -1.8457, which has a greater magnitude than any of the previous models. The p-value for distraction rate is also 0.00794, which is significant. From these results, it is clear that distraction rate has excellent potential for predicting student answer correctness. The simple logistic regression model is plotted in Figure 4-29.

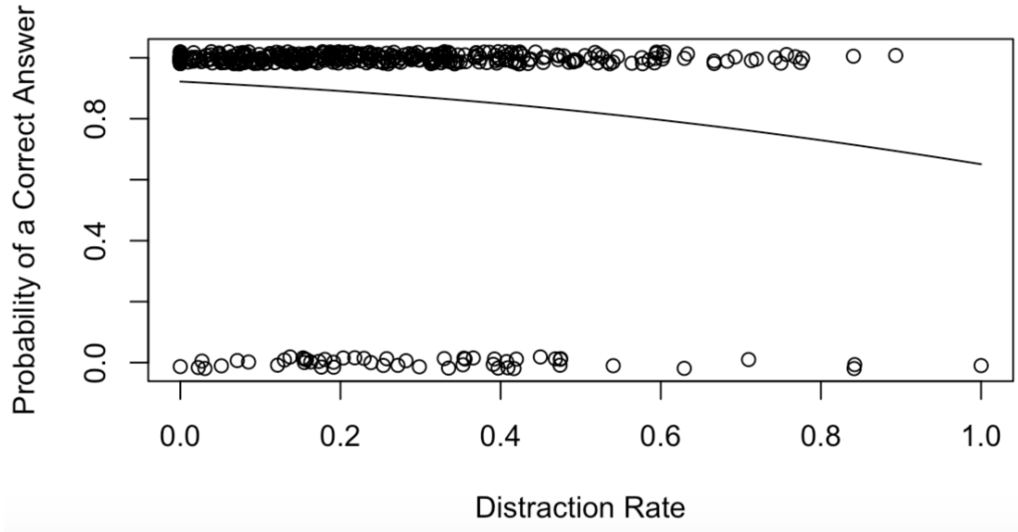

Figure 4-29: Distraction rate simple logistic regression model.

For the mixed-factor logistic regression models, the distraction rate appears to have a much greater prediction power than either of the average difference method metrics. The glmmPQL algorithm resulted in a parameter estimate of -3.4822 and a p-value of 0.0002. The parameter estimate for the glmer algorithm was -2.6767 with a p-value of 0.0127. The glmer algorithm failed to converge once again, but other than that, the results from both analyses are consistent, and distraction rate can be considered an accurate predictor of answer correctness when the variability from modules and students are considered. Additionally, the magnitude the distraction rate parameter estimates in both models are much greater than the parameter estimates for average difference (none of which were greater than 1).

4.6.3 Remove Modules with 100% Accuracy

Now, the modules with 100% accuracy will be removed from the data set to see if it has a positive effect on the significance of distraction rate as a predictor of answer correctness. The distraction rate vs module chart can be found in Figure 4-30.

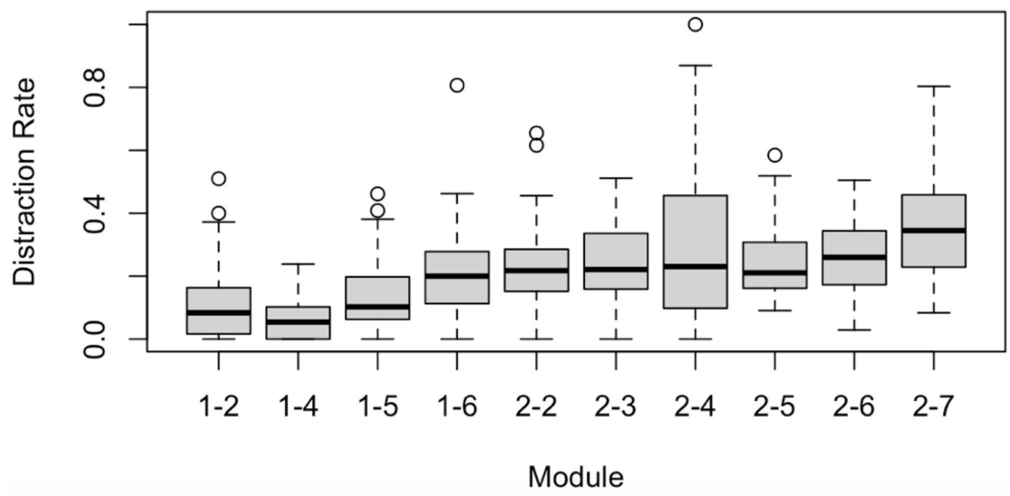

Figure 4-30: Distraction rate vs module (excluding modules with 100% accuracy).

The t-test will be conducted first. In this case, the p-value is $8.881*10^{-6}$, which is more significant that the original p-value of 0.003508. Removing all modules with 100% accuracy has once again resulted in an increased significance. The comparison of distraction rates for modules with correct vs incorrect answers when excluding modules with 100% accuracy is shown in figure 4-31.

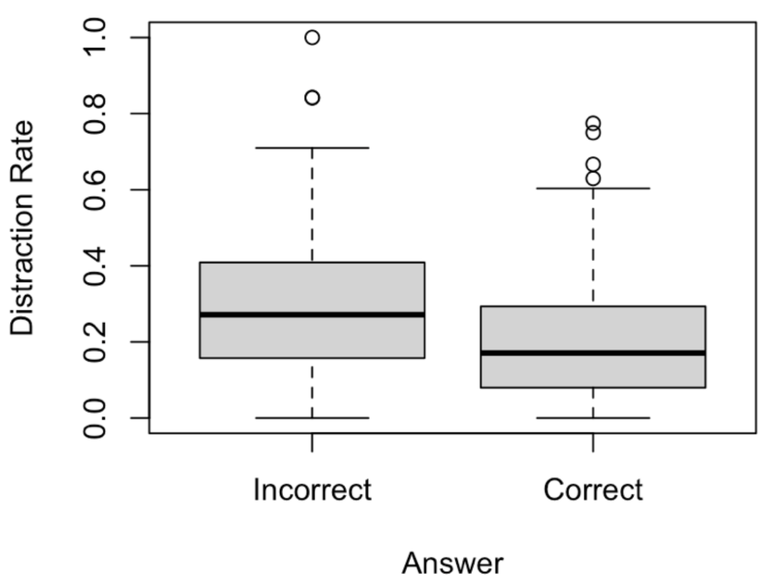

Figure 4-31: Distraction rate of modules with correct vs incorrect answers (excluding modules with 100% accuracy).

Next, distraction rate will be used as a factor in a simple logistic regression model. In this case, distraction rate has a parameter estimate of -3.2943, which by far the largest magnitude for a parameter estimate thus far. Based on the plot in Figure 4-32, it is clear that distraction rate has a very large influence on the probability of a student answering a question correctly. Additionally, the p-value for distraction rate in this model is $6.64 * 10^{-5}$, which is significant.

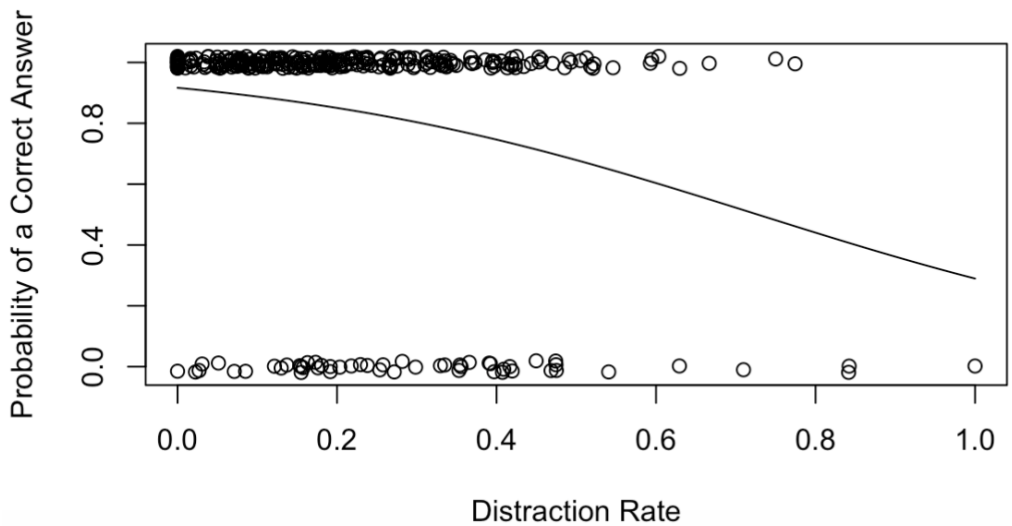

Figure 4-32: Distraction rate simple logistic regression model (excluding modules with 100% accuracy).

The results of the mixed effects linear regression models were not improved by the removal of modules with 100% accuracy. In the glmmPQL model, the parameter estimate for distraction rate was -2.8955 with a p-value of 0.0043. In the glmer model, the parameter estimate for distraction rate was -2.6684 with a p-value of 0.0248. Both of these significance values are lower than when the modules with 100% answer correctness were included. The parameter estimate decreased in the case of the glmmPQL model and remained relatively similar in the glmer model.

4.6.4 Parameter Optimization

With the implementation of the moving average window and minimum distance threshold, there are now decisions which must be made regarding the levels of these parameters. Increasing or decreasing the size of the moving average window or the threshold distance could have a significant impact on how accurately distraction rate is able to predict student answer correctness. To determine the optimal parameters for the obtained dataset, a range of possible parameter settings will be tested. The initial parameters are a minimum distance threshold of 1.5 and a moving average window of 5 seconds. The threshold will be adjusted between 1, 1.5, and 2 and the moving average window will be adjusted from 3 to 6 seconds. In total, there are 12 different combinations that will each be tested for significance. The t-test, simple logistic regression, and mixed-effects logistic regression models will all be considered. For the t-test and simple logistic regression models, modules with 100% answer correctness will be removed. They will not be removed for the mixed-effects model due to the negative effect it has shown on the significance of the model. Only the glmmPQL algorithm will be used for this analysis since the glmer function failed to converge in several previous cases. The results of the analysis are presented in Table 4-4.

Window	Threshold	T-test Significance	Simple logistic regression parameter estimate	Simple logistic regression significance	Mixed effects model parameter estimate	Mixed effects model significance
3	1	0.0000168	-3.1188	0.000075	-2.933274	0.0021
	1.5	0.0000262	-3.4459	0.000138	-3.254082	0.0022
	2	0.0001811	-3.5446	0.000814	-3.243266	0.0088
4	1	0.0000159	-2.9895	0.000072	-2.681724	0.0027
	1.5	0.0000131	-3.3023	0.000094	-3.313651	0.0007
	2	0.0000994	-3.4639	0.000529	-3.270048	0.0046
5	1	0.0000160	-2.8675	0.000072	-2.684645	0.0015
	1.5	0.0000089	-3.2943	0.000066	-3.482155	0.0002
	2	0.0000731	-3.3180	0.000547	-3.160450	0.0041
6	1	0.0000174	-2.7263	0.000077	-2.500014	0.0015
	1.5	0.0000198	-3.0234	0.000115	-2.924802	0.0010
	2	0.0001301	-3.1101	0.000784	-2.857706	0.0068

Table 4-4: Distraction rate parameter optimization test results.

Based on the parameter optimization analysis results, the initial parameter settings (window size of 5 seconds and a threshold of 1.5) appear to be the optimal choice. The initial settings have the best possible significance (p-value) across all tests. They also result in the mixed effects model parameter estimate with the greatest magnitude (-3.482155). The only category in which the initial settings are not optimal is the simple logistic regression parameter estimate. Even in that case, the difference between the initial settings and the best possible combination of parameters is only 0.2503 (which is only a 7.06% decrease in magnitude).

Chapter 5 - Results

5.1 Results Summary

The results of the various statistical tests can be compared, and it becomes clear that distraction rate is the best performing metric in all categories. For the t-test, distraction rate was clear of both average difference metrics when including and excluding modules with 100% answer correctness. The outcome was the same with the simple logistic regression model, where not only did distraction rate outperform in significance, but the magnitude of its parameter estimate was much larger. The greater magnitude indicates a much stronger relationship between distraction rate and answer correctness. Finally, in the case of the two mixed effects logistic regression models, distraction rate was also superior. Distraction rate had lower p-values than either average difference metric. Distraction rate's parameter estimates were much greater in magnitude as well. The complete list of statistical test results can be found in Table 5-1.

	Including Modules with 100% Accuracy			Excluding Modules with 100% Accuracy		
	Average X Difference	Average Euclidean Distance	Distraction Rate	Average X Difference	Average Euclidean Distance	Distraction Rate
T-test	0.02226	0.03226	0.003508	0.0001715	0.0002725	8.881E-06
Simple LR Parameter Estimate	-0.5745	-0.5431	-1.8457	-1.2209	-1.2047	-3.2943
Simple LR p-value	0.0463	0.0663	0.00794	0.000685	0.000993	0.0000664
glmmPQL Parameter Estimate	-0.8973	-0.8594	-3.4822	-0.6728	-0.6355	-2.8955
glmmPQL p-value	0.0393	0.0518	0.0002	0.1487	0.1788	0.0043
glmer Parameter Estimate	-0.5906	-0.5604	-2.6767	n/a	n/a	-2.6684
glmer P-value	0.226	0.258	0.0127	n/a	n/a	0.0248

Table 5-1: Results summary.

Next, the results from before and after removing modules with 100% answer correctness can be compared. In the case of the t-test and simple logistic regression models, all attention monitoring metrics saw improvements in significance and parameter estimate magnitude. The mixed effects models, however, did not improve in the same manor. In fact, all parameter estimates decreased in magnitude in addition to becoming less significant. The glmer model was also not able to successfully produce a model for either of the average difference methods.

The comparison between the average x-coordinate difference and average Euclidean difference is worth noting as well. There did not seem to be a noticeable difference between the two metrics. Average x-coordinate difference performed slightly better in all tests, although the difference was miniscule compared to the gap between them and distraction rate.

5.2 Discussion of Results

The distinguishing factor which caused distraction rate to outperform average difference is its ability to filter out false signals. These false signals stem from how the Microsoft HoloLens tracks headset orientation rather than pupil movements. Students are capable of focusing on the virtual instructor even if their headset isn’t perfectly aligned with the baseline. These small to medium sized deviations are insignificant but are still accounted for by the average difference method. The minimum distance threshold is a simple change which disregards any deviations which aren’t large enough to provide insight as to whether the student is paying attention or not. The average difference method also lends itself to the issue of unintentionally weighted signals. When a student has become distracted, the extent to which their vision deviates from the baseline is irrelevant. The only information that needs to be recorded is whether or not the student is distracted. The binary detection signal address this issue by evenly weighting each observation which is beyond the minimum distance threshold.

Another source of false signals which distraction rate is able to filter out are short-term deviations from the baseline. Tables and figures are included within the AR space which provide students with supplementary information as they follow along the problem-solving process. Animated 3-dimensional figures also help students visualize complex problems and are one of most significant benefits of AR learning environments. When students glance at these objects during a lecture, it is not an indication of distraction unless they ignore the virtual instructor for an extended period of time. The moving average window which is included in the distraction rate calculation reduces the influence of brief deviations from the baseline. Instead, only prolonged differences in eye-tracking coordinates are detected. This further increases the significance of distraction rate and makes it a much stronger predictor of student answer correctness.

Another aspect of the results which needs to be considered is the effect of removing modules with 100% answer correctness. The justification for removing these modules is that the lack of an even split between correct and incorrect answers could lead to problems when fitting regression models. Removing the four modules with 100% answer correctness (1-1, 1-3, 1-7, and 2-1) could help alleviate this issue while still providing the algorithm with enough information to accurately predict answer correctness. In the case of the t-test and simple logistic regression model, an improvement in significance was found as expected. The mixed effects models, however, resulted in a much lower significance. This is likely because modules are included as factors in this model, and when specific modules are removed, the information which they provide to the model is lost. This could potentially skew the estimated effects of the different modules and take away from the overall predictive power of the model.

5.3 Applications

5.3.1 Module-Based Feedback System

The distraction rate attention monitoring method has the potential to increase the learning gains of students if it can be effectively utilized in a real-time feedback system. The current method calculates the distraction rate by analyzing the data from one full module at a time and must wait until the complete dataset is available before calculations can occur. Therefore, if the current system is to be implemented, then it will only be able to provide results after the module has been completed. This wouldn't require the program to be modified in any way, the only addition that would need to be made is a program that can upload the eye-tracking data to the analysis software immediately following the completion of each module. After the resulting distraction rate is calculated, then it could be used to provide the student with feedback.

The feedback that will be provided to students will be based on a set of three different attention level categories: low, medium, and high. Students whose distraction rate is below the minimum threshold will be given positive feedback and encouraged to maintain their current attention levels. If the student is between the minimum and maximum thresholds, then they will be allowed to continue to on with the next lecture but will also be reminded to follow the virtual instructor. For students whose distraction rate is above the maximum threshold, they will be asked to repeat the previous module as well as being reminded to follow the virtual instructor. Table 5-2 provides a list of the three categories along with the corresponding recommended distraction rate thresholds and feedback messages. For reference, the distraction rate distribution is also provided in Figure 5-1.

Attention Level	Distraction Rate Range	Feedback
High	Less than 0.2	"Excellent job following the lecture material. Keep it up as you continue onto the next module!"
Medium	Between 0.2 and 0.6	"Remember to follow the virtual instructor throughout the AR learning environment so that you don't miss out on any key information!"
Low	Greater than 0.6	"Our system has detected that you may not have been able to keep up with the virtual instructor during the previous module. You will be given another chance to rewatch the same module to make sure you didn't miss any key information!"

Table 5-2: Module-based feedback system attention levels.

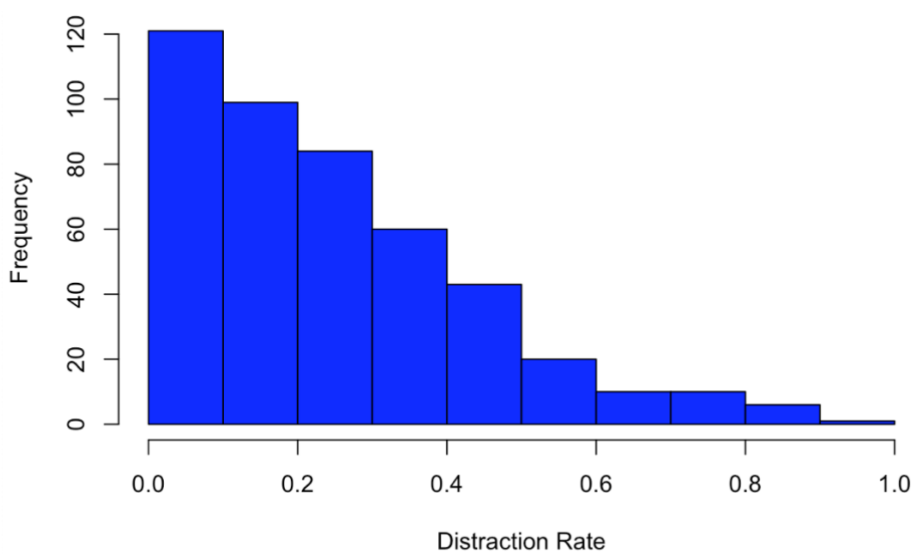

Figure 5-1: Distraction rate distribution.

5.3.2 Attention Guidance System

Rather than provide feedback after the module has already been completed, the system could also attempt to guide the student's attention in real-time whenever they become distracted. In order to achieve this, there are a few changes that would need to be made to the distraction rate calculation method in order to make it capable of providing feedback in real-time. First and foremost, the HoloLens eye-tracking data would need to be continuously uploaded into the analysis software throughout the duration of the AR lecture. This process would need to occur with little or no delay between when the students' eye-tracking data is recorded and when it is uploaded to the analysis software. If this can be done, then the data analysis would be similar to the current distraction rate calculation process. First, the live data would be compared to the baseline dataset by calculating the Euclidean distance between the student and the baseline. Next, it will be converted to a moving average just as before. Then, the moving average difference can be monitored in real-time.

Once the eye-tracking monitoring system is in place, the feedback system will need to be set up. Whenever the moving average difference surpasses the designated threshold, the student will be given a signal to redirect their attention to the virtual instructor. The signal system will need to be implemented as a part of the AR interface. There are two potential types of signals which could be provided to the student. The first would be a simple message which appears on the screen telling the student to return their attention to the virtual instructor. The other signal could be an arrow which appears on the screen which guides the student back to the baseline coordinates. In order to determine the direction of the arrow, the difference between the student and baseline coordinates could be used. Since positive x-coordinate values correspond to the right side of the virtual environment and negative x-coordinate values correspond to the left, the sign of the x-coordinate difference would indicate which direction the student needs to turn to find the target coordinates. For example, if the student's x-coordinate is 3 and the baseline x-coordinate is -1, the difference between the student and baseline coordinates would be (3-(-1)) = 4. Since this value is positive, it indicates that the student is currently looking to the right of the target coordinates and that a signal should be provided which directs the student's attention to the left. If these simple attention guidance signals could be implemented in real-time, students would be much less likely to miss out on important information during the lecture. Figure 5-2 provides the framework diagram for the attention guidance system.

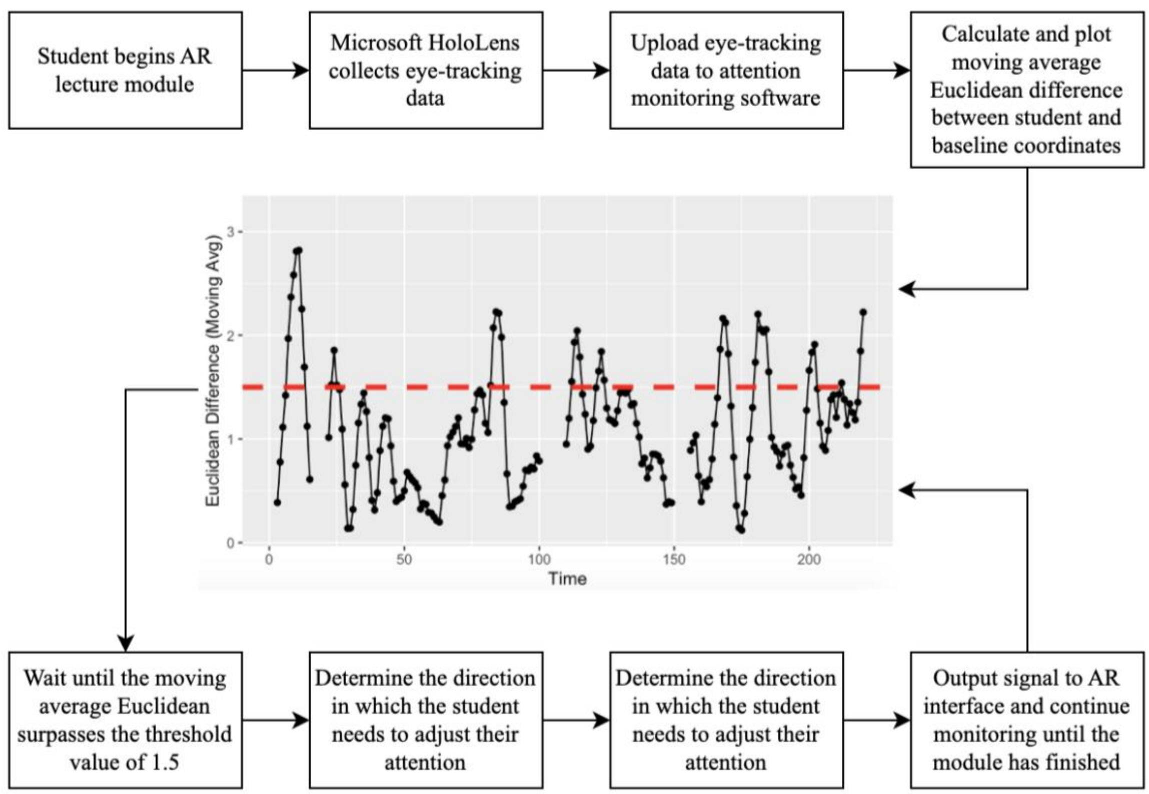

Figure 5-2: Attention guidance system diagram.

Chapter 6 - Conclusions

6.1 Conclusions

Based on the results of this research, student eye-tracking data appears to have a significant correlation with student answer correctness in AR learning environments. The use of a baseline dataset has proven to be an excellent foundation for eye-tracking data analysis methods. The baseline dataset is also an essential part of the proposed attention guidance system. The average difference method has the potential to be an effective predictor of student performance, although it hardly compares to the effectiveness of the distraction rate method. The simplicity of the average difference method does make it a useful comparison tool for assessing the effectiveness of other eye-tracking data analysis methods. For this reason, even if it is not used as the primary eye-tracking data analysis method, average difference should be considered in future experiments.

The distraction rate method has established itself as the most effective predictor of student answer correctness in AR learning environments. It outperformed average difference in every statistical test and predictive model. It can also be used to monitor student attention levels in real-time as a part of the proposed attention guidance system. The optimal distraction rate parameters in this application are a distance threshold of 1.5 and a moving average window of 5 seconds. These parameters may not be optimal in other AR learning environments but would still be an adequate starting point.

6.2 Limitations

One of the most significant limitations of this experiment has to do with the student answer data. The split between correct and incorrect answers was heavily skewed with an overall accuracy of 87.4%. This was the main reason why machine learning algorithms were not considered as a method for predicting student performance. If possible, the difficult of the lecture material should be increased in order to result in an even split between student answer correctness. Additionally, free response questions should be used rather than multiple choice questions so that partial credit may be given. This would provide more information regarding the students' learning comprehension than only knowing if the answer was correct or incorrect.

Another problem encountered during this experiment was that students were able to view questions during the module. The first problem this causes is that it will simply distract the students. They will look away from the virtual instructor to read the question and then look around to search for the answer rather than focusing on the lecture. This leads to more sporadic eye movements which reduces level of insight provided by the data. The other problem with students looking at the question during the module is that the Microsoft HoloLens does not record eye-tracking data when the students are not looking at one of the five content panels. Therefore, any time a student looks down at their desk to read the question, it causes a gap in the data. These gaps are unaccounted for in the analysis even though they could be used indicate that a student is distracted.

In any future experiments involving AR learning environments, the lab design is undoubtedly the most important element to consider. The limitations of the AR system and the eye-tracking data collection system must be accounted for when designing the layout of the room as well as the virtual learning content. In the case of the Microsoft HoloLens, there must not be anything within the lab which would remove the students' attention from the virtual learning content. The questions being available to the student in this experiment were an example of this which led to missing data and most likely lowered the significance in the resulting prediction models.

6.3 Future Studies

The average difference and distraction rate methods are not the only two ways to compare student eye-tracking data to a baseline dataset. Further studies should be conducted which investigate alternative methods and assess their ability to predict student performance. Additionally, the use of machine learning modules would most likely be effective in this type of application. Many previous studies have utilized machine learning algorithms to analyze eye-tracking data and found success (Dzsotjan et al., 2021; Vortmann, 2019). Unfortunately, this experiment didn't yield enough data or a large enough split between student answer correctness to warrant the application of machine learning algorithms. Future experiments should be conducted which compare the effectiveness of the average difference and distraction rate methods to machine learning algorithms.

Once the most effective attention monitoring method has been identified, it should be implemented as a part of a real-time feedback system. The module-based feedback system and the attention guidance system should prevent students from missing out on important information during AR learning modules. Previous studies have found success with similar attention monitoring methods but have not incorporated the use of a baseline dataset to help redirect student attention (Biocca et al., 2006; Vortmann, 2019). These systems should be tested to determine if they can have a positive effect on learning outcomes. Students should also be surveyed to find out whether they prefer a continuous attention monitoring system or a module-based feedback system.

REFERENCES

Akçayır, M., & Akçayır, G. (2017). Advantages and challenges associated with augmented reality for education: A systematic review of the literature. *Educational Research Review, 20*, 1–11. https://doi.org/10.1016/j.edurev.2016.11.002

Alemdag, E., & Cagiltay, K. (2018). A systematic review of eye tracking research on multimedia learning. *Computers & Education*, *125*, 413–428. https://doi.org/10.1016/j.compedu.2018.06.023

Alzahrani, N. M. (2020). Augmented Reality: A Systematic Review of Its Benefits and Challenges in E-learning Contexts. *Applied Sciences*, *10*(16). https://doi.org/10.3390/app10165660

Asish, S. M., Kulshreshth, A. K., & Borst, C. W. (2022). Detecting distracted students in educational VR environments using machine learning on eye gaze data. *Computers & Graphics, 109*, 75–87. https://doi.org/https://doi.org/10.1016/j.cag.2022.10.007

Ban Hassan Majeed, & ALRikabi, H. TH. S. (2022). Effect of Augmented Reality Technology on Spatial Intelligence among High School Students. *International Journal of Emerging Technologies in Learning (IJET)*, *17*(24), 131–143. https://doi.org/10.3991/ijet.v17i24.35977

Billinghurst, M. (2002). Augmented reality in education. *New Horizons for Learning, 12*(5), 1–5.

Biocca, F., Tang, A., Owen, C., & Xiao, F. (2006). Attention Funnel: Omnidirectional 3D Cursor for Mobile Augmented Reality Platforms. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1115–1122. https://doi.org/10.1145/1124772.1124939

Bolker, B. M., Brooks, M. E., Clark, C. J., Geange, S. W., Poulsen, J. R., Stevens, M. H. H., & White, J.-S. S. (2009). Generalized linear mixed models: a practical guide for ecology and evolution. *Trends in Ecology & Evolution, 24*(3), 127–135. https://doi.org/https://doi.org/10.1016/j.tree.2008.10.008

Bos, A. S., Herpich, F., Kuhn, I., Guarese, R. L. M., Tarouco, L. M. R., Zaro, M. A., Pizzato, M., & Wives, L. (2019). Educational Technology and Its Contributions in Students' Focus and Attention Regarding Augmented Reality Environments and the Use of Sensors. *Journal of Educational Computing Research*, *57*(7), 1832–1848. https://doi.org/10.1177/0735633119854033

Bower, M., Howe, C., McCredie, N., Robinson, A., & Grover, D. (2014). Augmented Reality in education – cases, places and potentials. *Educational Media International*, *51*(1), 1–15. https://doi.org/10.1080/09523987.2014.889400

Buettner, R., Sauer, S., Maier, C., & Eckhardt, A. (2018). Real-time Prediction of User Performance based on Pupillary Assessment via Eye-Tracking. *AIS Transactions on Human-Computer Interaction*, 26–60. https://doi.org/10.17705/1thci.00103

Cai, S., Liu, E., Shen, Y., Liu, C., Li, S., & Shen, Y. (2020). Probability learning in mathematics using augmented reality: impact on student's learning gains and attitudes. *Interactive Learning Environments*, *28*(5), 560–573. https://doi.org/10.1080/10494820.2019.1696839

Chen, C. H., Ho, C.-H., & Lin, J.-B. (2015). The Development of an Augmented Reality Game-based Learning Environment. *Procedia - Social and Behavioral Sciences*, *174*, 216–220. https://doi.org/10.1016/j.sbspro.2015.01.649

Chettaoui, N., Atia, A., & Bouhlel, M. S. (2023). Student Performance Prediction with Eye-Gaze Data in Embodied Educational Context. *Education and Information Technologies*, *28*(1), 833–855. https://doi.org/10.1007/s10639-022-11163-9

Dhar, P., Rocks, T., Samarasinghe, R. M., Stephenson, G., & Smith, C. (2021). Augmented reality in medical education: students' experiences and learning outcomes. *Medical Education Online*, *26*(1), 1953953. https://doi.org/10.1080/10872981.2021.1953953

Dunleavy, M., Dede, C., & Mitchell, R. (2009). Affordances and Limitations of Immersive Participatory Augmented Reality Simulations for Teaching and Learning. *Journal of Science Education and Technology*, *18*(1), 7–22. https://doi.org/10.1007/s10956-008-9119-1

Dzsotjan, D., Ludwig-Petsch, K., Mukhametov, S., Ishimaru, S., Kuechemann, S., & Kuhn, J. (2021). The Predictive Power of Eye-Tracking Data in an Interactive AR Learning Environment. *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*, 467–471. https://doi.org/10.1145/3460418.3479358

Gurevych, R., Silveistr, A., Mokliuk, M., Shaposhnikova, I., Gordiichuk, G., & Saiapina, S. (2021). Using Augmented Reality Technology in Higher Education Institutions. *Postmodern Openings*, *12*(2). https://doi.org/10.18662/po/12.2/299

Heinz, M., Büttner, S., & Röcker, C. (2019). Exploring Training Modes for Industrial Augmented Reality Learning. *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, 398–401. https://doi.org/10.1145/3316782.3322753

Kaur, D. P., Mantri, A., & Horan, B. (2022). Design implications for adaptive augmented reality based interactive learning environment for improved concept comprehension in engineering paradigms. *Interactive Learning Environments*, *30*(4), 589–607. https://doi.org/10.1080/10494820.2019.1674885

Khedher, A., Jraidi, I., & Frasson, C. (2019, April). *Predicting Learners' Performance Using EEG and Eye Tracking Features*.

Lee, K. (2012). Augmented Reality in Education and Training. *TechTrends*, *56*(2), 13–21. https://doi.org/10.1007/s11528-012-0559-3

Lee, T., Wen, Y., Chan, M. Y., Azam, A. B., Looi, C. K., Taib, S., Ooi, C. H., Huang, L. H., Xie, Y., & Cai, Y. (2022). Investigation of virtual & augmented reality classroom learning environments in university STEM education. *Interactive Learning Environments*, *0*(0), 1–16. https://doi.org/10.1080/10494820.2022.2155838

Li, X., Younes, R., Bairaktarova, D., & Guo, Q. (2020). Predicting Spatial Visualization Problems' Difficulty Level from Eye-Tracking Data. *Sensors*, *20*(7). https://doi.org/10.3390/s20071949

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. https://doi.org/10.1016/j.compedu.2016.01.001

Martin-Gutierrez, J., & Meneses Fernández, M. (2014). Applying Augmented Reality in Engineering Education to Improve Academic Performance & Student Motivation. *International Journal of Engineering Education*, *30*, 625–635.

Mayer, R. E. (2010). Unique contributions of eye-tracking research to the study of learning with graphics. *Learning and Instruction*, *20*(2), 167–171. https://doi.org/10.1016/j.learninstruc.2009.02.012

Peterson Joshua and Pardos, Z. and R. M. and S. A. and G. C. and M. J. (2015). Understanding Student Success in Chemistry Using Gaze Tracking and Pupillometry. In N. and M. A. and V. M. F. Conati Cristina and Heffernan (Ed.), *Artificial Intelligence in Education* (pp. 358–366). Springer International Publishing.

Radosavljevic, S., Radosavljevic, V., & Grgurovic, B. (2020). The potential of implementing augmented reality into vocational higher education through mobile learning. *Interactive Learning Environments*, *28*(4), 404–418. https://doi.org/10.1080/10494820.2018.1528286

Radu, I. (2012). Why should my students use AR? A comparative review of the educational impacts of augmented-reality. *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 313–314. https://doi.org/10.1109/ISMAR.2012.6402590

Rahman, Y., Asish, S. M., Fisher, N. P., Bruce, E. C., Kulshreshth, A. K., & Borst, C. W. (2020). Exploring Eye Gaze Visualization Techniques for Identifying Distracted Students in Educational VR. *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, 868–877. https://doi.org/10.1109/VR46266.2020.00009

Renner, P., & Pfeiffer, T. (2017). Attention guiding techniques using peripheral vision and eye tracking for feedback in augmented-reality-based assistance systems. *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, 186–194. https://doi.org/10.1109/3DUI.2017.7893338

Rodrigues, P., & Rosa, P. J. (2017). *Eye-Tracking as a Research Methodology in Educational Context* (pp. 1–26). https://doi.org/10.4018/978-1-5225-1005-5.ch001

Saidin, N. F., Abd Halim, N. D., & Yahaya, N. (2015). A Review of Research on Augmented Reality in Education: Advantages and Applications. *International Education Studies*, *8*(13). https://doi.org/10.5539/ies.v8n13p1

Sharma, K., Caballero, D., Verma, H., Jermann, P., & Dillenbourg, P. (2015). Looking AT versus Looking THROUGH: A Dual Eye-Tracking Study in MOOC Context. *International Conference on Computer Supported Collaborative Learning*.

Sharma, K., Giannakos, M., & Dillenbourg, P. (2020). Eye-tracking and artificial intelligence to enhance motivation and learning. *Smart Learning Environments*, *7*(1), 13. https://doi.org/10.1186/s40561-020-00122-x

Soltis, N. A., McNeal, K. S., Atkins, R. M., & Maudlin, L. C. (2020). A novel approach to measuring student engagement while using an augmented reality sandbox. *Journal of Geography in Higher Education*, *44*(4), 512–531. https://doi.org/10.1080/03098265.2020.1771547

Stigall, J., Bodempudi, S. T., Sharma, S., Scribner, D., Grynovicki, J., & Grazaitis, P. (2018). Building Evacuation using Microsoft HoloLens. *27th International Conference on Software Engineering and Data Engineering*, 8–10.

Vortmann, L.-M. (2019). Attention-driven Interaction Systems for Augmented Reality. *2019 International Conference on Multimodal Interaction*, 482–486. https://doi.org/10.1145/3340555.3356088

Wang, C.-Y., Tsai, M.-J., & Tsai, C.-C. (2016). Multimedia recipe reading: Predicting learning outcomes and diagnosing cooking interest using eye-tracking measures. *Computers in Human Behavior*, *62*, 9–18. https://doi.org/https://doi.org/10.1016/j.chb.2016.03.064

Wu, H.-K., Lee, S. W.-Y., Chang, H.-Y., & Liang, J.-C. (2013). Current status, opportunities and challenges of augmented reality in education. *Computers & Education*, *62*, 41–49. https://doi.org/10.1016/j.compedu.2012.10.024

APPENDIX

Average X Difference Statistical Test Output

T-test comparing the average x-coordinate difference of modules which students answered correctly with modules which students answered incorrectly:

```
        Two Sample t-test

data:  data$Avg.x.Diff[which(data$Answer == "Incorrect")] and data$Avg.x.Diff[which(data$Answer == "Correct")]
t = 2.0151, df = 432, p-value = 0.02226
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.0252044       Inf
sample estimates:
mean of x mean of y
1.0937309 0.9552346
```

Simple logistic regression model output for average x-coordinate difference:

```
Call:
glm(formula = Answer ~ Avg.x.Diff, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2264   0.4426   0.4824   0.5261   0.8526

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   2.5593     0.3422   7.480 7.43e-14 ***
Avg.x.Diff   -0.5745     0.2883  -1.993   0.0463 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 322.14  on 433  degrees of freedom
Residual deviance: 318.34  on 432  degrees of freedom
  (28 observations deleted due to missingness)
AIC: 322.34

Number of Fisher Scoring iterations: 4
```

glmmPQL mixed-effects logistic regression model output for average x-coordinate difference:

```
> summary(modpql)
Linear mixed-effects model fit by maximum likelihood
  Data: data
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | Student
        (Intercept)  Residual
StdDev:   0.9278026 0.7444562

Variance function:
 Structure: fixed weights
 Formula: ~invwt
Fixed effects:  Answer ~ Avg.x.Diff + Mod
                 Value Std.Error  DF    t-value p-value
(Intercept)  29.995043 198856.65 389  0.0001508  0.9999
Avg.x.Diff   -0.897278      0.43 389 -2.0675924  0.0393
Mod1-2      -25.718360 198856.65 389 -0.0001293  0.9999
Mod1-3        0.763270 280142.29 389  0.0000027  1.0000
Mod1-4      -25.854504 198856.65 389 -0.0001300  0.9999
Mod1-5      -27.990233 198856.65 389 -0.0001408  0.9999
Mod1-6      -26.727546 198856.65 389 -0.0001344  0.9999
Mod1-7        1.260389 283121.47 389  0.0000045  1.0000
Mod2-1        0.638825 281209.89 389  0.0000023  1.0000
Mod2-2      -27.122005 198856.65 389 -0.0001364  0.9999
Mod2-3      -25.461456 198856.65 389 -0.0001280  0.9999
Mod2-4      -27.812447 198856.65 389 -0.0001399  0.9999
Mod2-5      -26.996120 198856.65 389 -0.0001358  0.9999
Mod2-6      -27.656021 198856.65 389 -0.0001391  0.9999
Mod2-7      -29.067135 198856.65 389 -0.0001462  0.9999
```

glmer mixed-effects logistic regression model output for average x-coordinate difference:

```
> summary(model)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: Answer ~ Avg.x.Diff + Mod + (1 | Student)
   Data: data

     AIC      BIC   logLik deviance df.resid
   267.9    333.0   -117.9    235.9      418

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.3328  0.0000  0.1635  0.3680  2.1326

Random effects:
 Groups  Name        Variance Std.Dev.
 Student (Intercept) 0.3252   0.5702
Number of obs: 434, groups:  Student, 31

Fixed effects:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   20.7423  4956.9169   0.004    0.997
Avg.x.Diff    -0.5906     0.4882  -1.210    0.226
Mod1-2       -16.7868  4956.9170  -0.003    0.997
Mod1-3         0.2851  6686.6718   0.000    1.000
Mod1-4       -16.8608  4956.9170  -0.003    0.997
Mod1-5       -18.9956  4956.9170  -0.004    0.997
Mod1-6       -17.8293  4956.9170  -0.004    0.997
Mod1-7         1.0124  7324.7867   0.000    1.000
Mod2-1        -0.1248  6235.1018   0.000    1.000
Mod2-2       -18.1918  4956.9170  -0.004    0.997
Mod2-3       -16.6019  4956.9171  -0.003    0.997
Mod2-4       -18.9425  4956.9170  -0.004    0.997
Mod2-5       -18.1180  4956.9170  -0.004    0.997
Mod2-6       -18.7857  4956.9170  -0.004    0.997
Mod2-7       -20.1968  4956.9170  -0.004    0.997
```

T-test comparing the average x-coordinate difference of modules which students answered correctly with modules which students answered incorrectly (excluding modules with 100% accuracy):

```
        Two Sample t-test

data:  data$Avg.x.Diff[which(data$Answer == "Incorrect")] and data$Avg.x.Diff[which(data$Answer == "Correct")]
t = 3.6144, df = 308, p-value = 0.0001757
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.1150672       Inf
sample estimates:
mean of x mean of y
1.0937309 0.8820346
```

Simple logistic regression model output for average x-coordinate difference (excluding modules with 100% accuracy):

```
Call:
glm(formula = Answer ~ Avg.x.Diff, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2367   0.4555   0.5335   0.6307   1.3717

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   2.7738     0.4020   6.899 5.22e-12 ***
Avg.x.Diff   -1.2209     0.3595  -3.396 0.000685 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 283.60  on 309  degrees of freedom
Residual deviance: 272.01  on 308  degrees of freedom
  (20 observations deleted due to missingness)
AIC: 276.01

Number of Fisher Scoring iterations: 4
```

glmmPQL mixed-effects logistic regression model output for average x-coordinate difference (excluding modules with 100% accuracy):

```
> summary(modpql)
Linear mixed-effects model fit by maximum likelihood
  Data: data
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | Student
        (Intercept)  Residual
StdDev:    0.700439 0.8912968

Variance function:
 Structure: fixed weights
 Formula: ~invwt
Fixed effects:  Answer ~ Avg.x.Diff + Mod
                Value Std.Error  DF   t-value p-value
(Intercept)  3.966331 0.9966708 269  3.979580  0.0001
Avg.x.Diff  -0.672823 0.4645643 269 -1.448288  0.1487
Mod1-4      -0.091222 1.3119340 269 -0.069532  0.9446
Mod1-5      -2.197986 1.0085438 269 -2.179366  0.0302
Mod1-6      -1.024621 1.0871636 269 -0.942472  0.3468
Mod2-2      -1.391185 1.0529590 269 -1.321215  0.1876
Mod2-3       0.204162 1.3175659 269  0.154954  0.8770
Mod2-4      -2.105598 1.0215669 269 -2.061145  0.0403
Mod2-5      -1.303233 1.0592580 269 -1.230326  0.2196
Mod2-6      -1.953646 1.0234298 269 -1.908921  0.0573
Mod2-7      -3.310928 1.0187664 269 -3.249939  0.0013
```

glmer mixed-effects logistic regression model output for average x-coordinate difference (excluding modules with 100% accuracy):

```
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod
]
 Family: binomial  ( logit )
Formula: Answer ~ Avg.x.Diff + Mod + (1 | Student)
   Data: data

     AIC      BIC   logLik deviance df.resid
   259.9    304.7   -117.9    235.9      298

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.2065  0.1511  0.2834  0.4370  2.1432

Random effects:
 Groups  Name        Variance Std.Dev.
 Student (Intercept) 0.3302   0.5747
Number of obs: 310, groups:  Student, 31

Fixed effects:
             Estimate Std. Error  z value Pr(>|z|)
(Intercept)  3.911995   0.003642 1074.117  < 2e-16 ***
Avg.x.Diff  -0.594317   0.003641 -163.209  < 2e-16 ***
Mod1-4      -0.025263   0.003639   -6.942 3.88e-12 ***
Mod1-5      -2.160211   0.003639 -593.552  < 2e-16 ***
Mod1-6      -0.989464   0.003640 -271.856  < 2e-16 ***
Mod2-2      -1.352745   0.003640 -371.666  < 2e-16 ***
Mod2-3       0.238823   0.003639   65.621  < 2e-16 ***
Mod2-4      -2.105108   0.003639 -578.426  < 2e-16 ***
Mod2-5      -1.277632   0.003640 -351.028  < 2e-16 ***
Mod2-6      -1.947172   0.003639 -535.013  < 2e-16 ***
Mod2-7      -3.360885   0.003639 -923.552  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Average Euclidean Difference Statistical Test Output

T-test comparing the average Euclidean difference of modules which students answered correctly with modules which students answered incorrectly:

```
        Two Sample t-test

data:  data$Avg.Diff[which(data$Answer == "Incorrect")] and data$Avg.Diff[which(data$Answer == "Correct")]
t = 1.8534, df = 432, p-value = 0.03226
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.01375897        Inf
sample estimates:
mean of x mean of y
 1.182005  1.057602
```

Simple logistic regression model output for average Euclidean difference:

```
Call:
glm(formula = Answer ~ Avg.Diff, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2136   0.4483   0.4838   0.5254   0.8207

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   2.5793     0.3739   6.898 5.27e-12 ***
Avg.Diff     -0.5431     0.2958  -1.836   0.0663 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 322.14  on 433  degrees of freedom
Residual deviance: 318.91  on 432  degrees of freedom
  (28 observations deleted due to missingness)
AIC: 322.91

Number of Fisher Scoring iterations: 4
```

glmmPQL mixed-effects logistic regression model output for average Euclidean difference:

```
> summary(modpql)
Linear mixed-effects model fit by maximum likelihood
  Data: data
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | Student
        (Intercept)  Residual
StdDev:   0.9192697 0.7462389

Variance function:
 Structure: fixed weights
 Formula: ~invwt
Fixed effects:  Answer ~ Avg.Diff + Mod
                 Value Std.Error  DF    t-value p-value
(Intercept)  30.104320 199802.51 389  0.0001507  0.9999
Avg.Diff     -0.859395      0.44 389 -1.9507121  0.0518
Mod1-2      -25.767063 199802.51 389 -0.0001290  0.9999
Mod1-3        0.717509 281210.28 389  0.0000026  1.0000
Mod1-4      -25.920061 199802.51 389 -0.0001297  0.9999
Mod1-5      -28.022632 199802.51 389 -0.0001403  0.9999
Mod1-6      -26.806621 199802.51 389 -0.0001342  0.9999
Mod1-7        1.150510 284338.30 389  0.0000040  1.0000
Mod2-1        0.540700 282685.36 389  0.0000019  1.0000
Mod2-2      -27.174485 199802.51 389 -0.0001360  0.9999
Mod2-3      -25.533946 199802.51 389 -0.0001278  0.9999
Mod2-4      -27.906627 199802.51 389 -0.0001397  0.9999
Mod2-5      -27.063214 199802.51 389 -0.0001354  0.9999
Mod2-6      -27.729313 199802.51 389 -0.0001388  0.9999
Mod2-7      -29.154923 199802.51 389 -0.0001459  0.9999
```

glmer mixed-effects logistic regression model output for average Euclidean difference:

```
> summary(model)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: Answer ~ Avg.Diff + Mod + (1 | Student)
   Data: data

     AIC      BIC   logLik deviance df.resid
   268.0    333.2   -118.0    236.0      418

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.4333  0.0000  0.1640  0.3658  2.0788

Random effects:
 Groups  Name        Variance Std.Dev.
 Student (Intercept) 0.3242   0.5694
Number of obs: 434, groups:  Student, 31

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.079e+01  4.895e+03   0.004    0.997
Avg.Diff    -5.604e-01  4.951e-01  -1.132    0.258
Mod1-2      -1.678e+01  4.895e+03  -0.003    0.997
Mod1-3       6.216e-03  6.267e+03   0.000    1.000
Mod1-4      -1.687e+01  4.895e+03  -0.003    0.997
Mod1-5      -1.899e+01  4.895e+03  -0.004    0.997
Mod1-6      -1.785e+01  4.895e+03  -0.004    0.997
Mod1-7       8.579e-01  7.091e+03   0.000    1.000
Mod2-1      -2.169e-01  6.128e+03   0.000    1.000
Mod2-2      -1.820e+01  4.895e+03  -0.004    0.997
Mod2-3      -1.662e+01  4.895e+03  -0.003    0.997
Mod2-4      -1.898e+01  4.895e+03  -0.004    0.997
Mod2-5      -1.813e+01  4.895e+03  -0.004    0.997
Mod2-6      -1.881e+01  4.895e+03  -0.004    0.997
Mod2-7      -2.023e+01  4.895e+03  -0.004    0.997
```

T-test comparing the average Euclidean difference of modules which students answered correctly with modules which students answered incorrectly (excluding modules with 100% accuracy):

```
        Two Sample t-test

data:  data$Avg.Diff[which(data$Answer == "Incorrect")] and data$Avg.Diff[which(data$Answer == "Correct")]
t = 3.491, df = 308, p-value = 0.0002757
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.1053819       Inf
sample estimates:
mean of x mean of y
1.1820047 0.9821957
```

Simple logistic regression model output for average Euclidean difference (excluding modules with 100% accuracy):

```
Call:
glm(formula = Answer ~ Avg.Diff, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2241   0.4646   0.5343   0.6269   1.3436

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   2.8721     0.4395   6.535 6.35e-11 ***
Avg.Diff     -1.2047     0.3659  -3.293 0.000993 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 283.60  on 309  degrees of freedom
Residual deviance: 272.78  on 308  degrees of freedom
  (20 observations deleted due to missingness)
AIC: 276.78

Number of Fisher Scoring iterations: 4
```

glmmPQL mixed-effects logistic regression model output for average Euclidean difference (excluding modules with 100% accuracy):

```
> summary(modpql)
Linear mixed-effects model fit by maximum likelihood
  Data: data
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | Student
        (Intercept)  Residual
StdDev:   0.6910794 0.8946899

Variance function:
 Structure: fixed weights
 Formula: ~invwt
Fixed effects:  Answer ~ Avg.Diff + Mod
                Value Std.Error  DF   t-value p-value
(Intercept)  4.007657 1.0194993 269  3.931005  0.0001
Avg.Diff    -0.635523 0.4714638 269 -1.347979  0.1788
Mod1-4      -0.104477 1.3172214 269 -0.079316  0.9368
Mod1-5      -2.189053 1.0117361 269 -2.163660  0.0314
Mod1-6      -1.052189 1.0895482 269 -0.965711  0.3351
Mod2-2      -1.398952 1.0564212 269 -1.324237  0.1865
Mod2-3       0.181933 1.3214576 269  0.137676  0.8906
Mod2-4      -2.145931 1.0218165 269 -2.100114  0.0367
Mod2-5      -1.322637 1.0622412 269 -1.245138  0.2142
Mod2-6      -1.978786 1.0258707 269 -1.928885  0.0548
Mod2-7      -3.348184 1.0196343 269 -3.283711  0.0012
```

glmer mixed-effects logistic regression model output for average Euclidean difference (excluding modules with 100% accuracy):

```
> summary(model)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod
]
 Family: binomial  ( logit )
Formula: Answer ~ Avg.Diff + Mod + (1 | Student)
   Data: data

     AIC      BIC   logLik deviance df.resid
   260.0    304.9   -118.0    236.0      298

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.3348  0.1527  0.2860  0.4402  2.0814

Random effects:
 Groups  Name        Variance Std.Dev.
 Student (Intercept) 0.3256   0.5706
Number of obs: 310, groups:  Student, 31

Fixed effects:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.967541   0.003668 1081.78   <2e-16 ***
Avg.Diff    -0.561132   0.003667 -153.01   <2e-16 ***
Mod1-4      -0.046442   0.003665  -12.67   <2e-16 ***
Mod1-5      -2.169067   0.003665 -591.83   <2e-16 ***
Mod1-6      -1.032604   0.003665 -281.73   <2e-16 ***
Mod2-2      -1.377621   0.003785 -363.94   <2e-16 ***
Mod2-3       0.201222   0.003665   54.90   <2e-16 ***
Mod2-4      -2.160409   0.003665 -589.49   <2e-16 ***
Mod2-5      -1.314216   0.003785 -347.19   <2e-16 ***
Mod2-6      -1.989411   0.003665 -542.81   <2e-16 ***
Mod2-7      -3.414839   0.003665 -931.84   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Distraction Rate Statistical Test Output

T-test comparing the distraction rate of modules which students answered correctly with modules which students answered incorrect:

```
        Two Sample t-test

data:  data$Distraction.Rate[which(data$Answer == "Incorrect")] and data$Distraction.Rate[which
(data$Answer == "Correct")]
t = 2.7146, df = 431, p-value = 0.003451
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.02997144        Inf
sample estimates:
mean of x mean of y
0.3105006 0.2341917
```

Simple logistic regression model output for distraction rate:

```
Call:
glm(formula = Answer ~ Distraction.Rate, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2584   0.4171   0.4664   0.5334   0.8544

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)        2.4688     0.2542   9.713  < 2e-16 ***
Distraction.Rate  -1.8457     0.6953  -2.654  0.00794 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 321.88  on 432  degrees of freedom
Residual deviance: 315.13  on 431  degrees of freedom
  (29 observations deleted due to missingness)
AIC: 319.13

Number of Fisher Scoring iterations: 4
```

glmmPQL mixed-effects logistic regression model output for distraction rate:

```
> summary(modpql)
Linear mixed-effects model fit by maximum likelihood
  Data: data
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | Student
        (Intercept)  Residual
StdDev:    1.030578 0.7401089

Variance function:
 Structure: fixed weights
 Formula: ~invwt
Fixed effects:  Answer ~ Distraction.Rate + Mod
                     Value Std.Error  DF   t-value p-value
(Intercept)      29.816125 192240.91 388  0.000155  0.9999
Distraction.Rate  -3.482155      0.94 388 -3.723115  0.0002
Mod1-2          -25.601531 192240.91 388 -0.000133  0.9999
Mod1-3            1.012443 269512.94 388  0.000004  1.0000
Mod1-4          -25.913916 192240.91 388 -0.000135  0.9999
Mod1-5          -27.908893 192240.91 388 -0.000145  0.9999
Mod1-6          -26.493778 192240.91 388 -0.000138  0.9999
Mod1-7            1.816002 274346.28 388  0.000007  1.0000
Mod2-1            0.888505 271696.82 388  0.000003  1.0000
Mod2-2          -26.828496 192240.91 388 -0.000140  0.9999
Mod2-3          -25.262331 192240.91 388 -0.000131  0.9999
Mod2-4          -27.565333 192240.91 388 -0.000143  0.9999
Mod2-5          -26.789278 192240.91 388 -0.000139  0.9999
Mod2-6          -27.494464 192240.91 388 -0.000143  0.9999
Mod2-7          -28.798708 192240.91 388 -0.000150  0.9999
```

glmer mixed-effects logistic regression model output for distraction rate:

```
> summary(model)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod
]
 Family: binomial  ( logit )
Formula: Answer ~ Distraction.Rate + Mod + (1 | Student)
   Data: data

     AIC      BIC   logLik deviance df.resid
   263.4    328.5   -115.7    231.4      417

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.3301  0.0000  0.1552  0.3447  2.6773

Random effects:
 Groups  Name        Variance Std.Dev.
 Student (Intercept) 0.4635   0.6808
Number of obs: 433, groups:  Student, 31

Fixed effects:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        20.70653 5007.70350   0.004   0.9967
Distraction.Rate   -2.67672    1.07375  -2.493   0.0127 *
Mod1-2            -16.71416 5007.70362  -0.003   0.9973
Mod1-3              0.01495 6071.94327   0.000   1.0000
Mod1-4            -16.94579 5007.70361  -0.003   0.9973
Mod1-5            -18.94958 5007.70352  -0.004   0.9970
Mod1-6            -17.63273 5007.70355  -0.004   0.9972
Mod1-7              1.33208 6933.05615   0.000   0.9998
Mod2-1              0.47446 6726.51681   0.000   0.9999
Mod2-2            -17.96378 5007.70354  -0.004   0.9971
Mod2-3            -16.42920 5007.70362  -0.003   0.9974
Mod2-4            -18.71980 5007.70353  -0.004   0.9970
Mod2-5            -17.95032 5007.70354  -0.004   0.9971
Mod2-6            -18.63914 5007.70352  -0.004   0.9970
Mod2-7            -19.97315 5007.70352  -0.004   0.9968
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

T-test comparing the distraction rate of modules which students answered correctly with modules which students answered incorrectly (excluding modules with 100% accuracy):

```
        Two Sample t-test

data:  data$Distraction.Rate[which(data$Answer == "Incorrect")] and data$Distraction.Rate[which
(data$Answer == "Correct")]
t = 4.3601, df = 307, p-value = 8.881e-06
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.06901996        Inf
sample estimates:
mean of x mean of y
0.3105006 0.1994662
```

Simple logistic regression model output for distraction rate (excluding modules with 100% accuracy):

```
Call:
glm(formula = Answer ~ Distraction.Rate, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2288   0.4174   0.5117   0.6274   1.2436

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)        2.3967     0.2762   8.679  < 2e-16 ***
Distraction.Rate  -3.2943     0.8259  -3.989 6.64e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 283.22  on 308  degrees of freedom
Residual deviance: 266.87  on 307  degrees of freedom
  (21 observations deleted due to missingness)
AIC: 270.87

Number of Fisher Scoring iterations: 4
```

glmmPQL mixed-effects logistic regression model output for distraction rate (excluding modules with 100% accuracy):

```
> summary(modpql)
Linear mixed-effects model fit by maximum likelihood
  Data: data
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | Student
        (Intercept)  Residual
StdDev:    0.811683 0.8771739

Variance function:
 Structure: fixed weights
 Formula: ~invwt
Fixed effects:  Answer ~ Distraction.Rate + Mod
                     Value Std.Error  DF   t-value p-value
(Intercept)       3.965533 0.9468508 268  4.188129  0.0000
Distraction.Rate -2.895498 1.0056592 268 -2.879204  0.0043
Mod1-4           -0.253332 1.2976582 268 -0.195222  0.8454
Mod1-5           -2.224460 0.9987438 268 -2.227258  0.0268
Mod1-6           -0.903664 1.0764886 268 -0.839455  0.4020
Mod2-2           -1.229551 1.0444889 268 -1.177180  0.2402
Mod2-3            0.298265 1.2981361 268  0.229764  0.8185
Mod2-4           -1.955886 1.0093977 268 -1.937676  0.0537
Mod2-5           -1.207976 1.0445924 268 -1.156409  0.2485
Mod2-6           -1.883856 1.0046349 268 -1.875164  0.0619
Mod2-7           -3.146753 0.9956875 268 -3.160383  0.0018
```

glmer mixed-effects logistic regression model output for distraction rate (excluding modules with 100% accuracy):

```
> summary(model)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod
]
 Family: binomial  ( logit )
Formula: Answer ~ Distraction.Rate + Mod + (1 | Student)
   Data: data

     AIC      BIC   logLik deviance df.resid
   255.4    300.2   -115.7    231.4      297

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.2529  0.1421  0.2604  0.4244  2.6696

Random effects:
 Groups  Name        Variance Std.Dev.
 Student (Intercept) 0.463    0.6805
Number of obs: 309, groups:  Student, 31

Fixed effects:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)        3.9616     1.0665   3.715 0.000204 ***
Distraction.Rate  -2.6684     1.1889  -2.244 0.024807 *
Mod1-4            -0.1971     1.4465  -0.136 0.891603
Mod1-5            -2.2075     1.1111  -1.987 0.046949 *
Mod1-6            -0.8751     1.1977  -0.731 0.465007
Mod2-2            -1.2272     1.1592  -1.059 0.289741
Mod2-3             0.3138     1.4462   0.217 0.828194
Mod2-4            -1.9782     1.1196  -1.767 0.077243 .
Mod2-5            -1.2057     1.1596  -1.040 0.298476
Mod2-6            -1.8963     1.1147  -1.701 0.088913 .
Mod2-7            -3.2291     1.1073  -2.916 0.003543 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```