Team Members: Vishnu Banna and Aakash Saravanan

Purdue Usernames: vbanna and saravan1

GitHub Usernames: vishnubanna and saravan1

GitHub Team Name: Vishnu's team

Project: Path 2: Student performance related to video-watching behavior

Dataset:

To determine the effects of student video-watching behavior on their performance on in-video quizzes, data collected from students watching videos from an online course, in 'behavior-performance.txt', will be parsed and analyzed. This data is collected from a study performed to predict whether a user will be Correct on First Attempt (CFA) in answering a question from student performance in Massive Open Online Courses (MOOC). In this dataset, the features that were examined for a student-video pair were userID, videoID, fracSpent, fracComp, fracPaused, numPauses, avgPBR, numRWs, numFFs, and s. userID corresponds to the anonymized ID of the student, where each student appears many times in the dataset and the whole dataset contains a total of 3976 students. video-id is the ID of the video ranging from 0 and 92 as there are 93 total videos. In this dataset, there are a total of 29304 student-video pairs. fracSpent is the total time, either playing or paused, a student spends watching a video divided by the length of the video. fracComp is the fraction of the video the student watched, ranging between 0 (none of the video watched) and 1 (video completely watched). fracPaused is the time the student paused on the video divided by the length of the video. numPauses is the number of times the student paused the video. avgPBR is the average playback rate that the student used while watching the video, ranging from 0.5x to 2.0x. numRWs is the number of times the student rewinds(skips backwards) in the video. numFFs is the number of times the student fast forwards(skips forwards) in the video. s is whether the student answered the question given directly after the video on their first attempt, either 0 (incorrect) or 1 (correct).

Methods:

1) How well can the students be naturally grouped or clustered by their video-watching behavior ('fracSpent', 'fracComp', 'fracPaused', 'numPauses', 'avgPBR', 'numRWs',

and ‘numFFs’)? In this analysis, all students that complete at least five of the videos were used.

The individual student’s data is combined, and another column for total videos watched is created to check how many videos each student watched. Thus, students that did not complete at least five of the videos were filtered out of the dataset. Before determining whether the students can be naturally grouped or clustered by their video-watching behavior, the TSNE (t-distributed Stochastic Neighbor Embedding, a module in sklearn, is used. This model reduces the number of dimensions of the dataset as this dataset offers several features (many dimensions). This tool is used to visualize high-dimensional data by converting similarities between data points to joint probabilities and minimizing the divergence between the data and the joint probabilities. The TSNE model was used primarily to reduce the number of dimensions and make it easier to visualize each student on the graph (summarizes each student’s information to represent them on a lower dimension graph). Using this method, the dataset can be visualized to determine if there are clusters that are formed by reducing the number of dimensions to 3. It allows us to visualize all the students as a datapoint on a graph without having to graph each feature independently. The features that were used for the TSNE model were all the student’s video-watching behavior features: ‘fracSpent’, ‘fracComp’, ‘fracPaused’, ‘numPauses’, ‘avgPBR’, ‘numRWs’, and ‘numFFs’. This method can be verified by using a K-means to cluster the output of the TSNE model, as the K-means clustering approach finds simple structures in the data (points that are close together). Using K-means clustering allows us to see how the datapoints are clustering together without having to plot all the axis. It also allows to find an optimal value for k for K-means clustering in a visual manner. If the dataset can be clustered together, then there are clusters within the features of the video-watching behavior that can be grouped. Since the TSNE model shows that clusters can be formed from the video-watching behavior features, K-means clustering is used to determine which features can be naturally grouped or clustered. K-means was chosen as there is no model required and the model clusters based on the closeness of the datapoints. Different features of the students’ video watching behavior were plotted against each other to determine whether the two features can be naturally grouped or clustered. The number of clusters, k, is the value determined from the TSNE model. This model is verified by collecting pairs of video-watching behavior features that can be naturally clustered or grouped.

2) Can student's video-watching behavior be used to predict a student's performance (i.e., average score 's' across all quizzes)? This analysis could ultimately save significant time by avoiding the need for tests. In this analysis, all students that complete at least half of the quizzes were used.

A student's video-watching behavior can be used to predict a student's performance if there is a relationship between a student's video-watching behavior and their performance on video quizzes. A polynomial ridge regression model with degree two will be used to train the given dataset and predict future student's performance given video-watching behavior parameters. Using this model allows for mitigating overfitting by penalizing non-zero model coefficients. This model is chosen to begin with because the model itself is easy to train and if the output is valid with a consistently low MSE, we can extract information about the features easily. It was chosen to be a polynomial model of degree two because this model offers the least MSE on the validation dataset with this degree. Before fitting the training dataset to the model, the dataset is filtered such only the students that complete at least half of the quizzes remain. This is determined by filtering out the students who did not completely watch at least half of the total videos. The regularization parameter, lamda, is determined from varying lamda and seeing the effect on MSE. The best model has the lambda value corresponding to the lowest MSE from the validation data. The regression model was trained repeatedly, such that the dataset given to the model was reshuffled each time the regression was trained. This was done to ensure that the model was truly learning about the data input (video-watching behavior features) and the MSE was consistently low. These two factors helped verify the accuracy of the model and how well the model can predict a students' performance given a student's video watching behavior. Next, we consider the implementation of a simple neural network. A neural network is a sequence of layered neurons in which each neuron is a linear function with a non-linear activation function. The key feature of a neural network is that the neurons can learn from a process called back propagation which uses an error function to guide the network in its learning process. This allows the model to fit to the non-linear features of the dataset. This method is chosen after the polynomial ridge regression because the polynomial regression proved that there is a non-linear relationship within the data. The network was trained with 1 epoch and a batch size of 2. These values were chosen as increasing it would cause the network to overfit to the training dataset. Considering the MSE on the validation and the training dataset was less than 0.1, it can be

concluded that there is a relationship between a students' video-watching behavior and their overall performance on video quizzes. To prove that there is a strong correlation between a students' video-watching behavior and a students' performance in the online course, we re-filtered the dataset to include all students who watched more than ten videos in the course. We used all the video-watching behavior features and the total number of videos that each student watched to develop both the polynomial ridge regression and the neural network.

3) Taking step (2) a step further, how well can you predict a student's performance on a *particular* in-video quiz question (i.e., whether they will be correct or incorrect) based on their video-watching behaviors while watching the corresponding video? In this analysis, all student-video pairs are used.

For all the models, we used the features associated with the students' watching behavior, then we calculated the time-series data for the accumulation of the number of videos watched and the number of points accumulated at each repeated occurrence of a userID. Because the videos are chronological, from 0 to 92, a student at video 14 could potentially have watched 13 videos up to this point. We included this feature hoping this data could help the model predict how well the student performs on the current video. Initially, we use a linear ridge regression to fit a model to predict a student's performance on an in-video quiz question. This model is the fastest to fit the data and shows if individual features are related to the labels. To verify this method, check the MSE of the model, and then graphed the model's output to the actual output. Due to low MSE and the model only outputting 0.5, we decided to try classification methods instead. We tried a Logistic regression because it uses similar methodology to a linear regression, but it limits its outputs such that it could be used for classification. It uses a class boundary to limit its values to 0 to 1. Because these are classification models, we start to look at accuracy instead of MSE. To verify these classification models, we started to consider the accuracy of the models. In terms of classification methods, we tried a Gaussian Bayes model, kNN, multinomial naïve Bayes, and we tried to apply an unsupervised GMM to potentially predict students' performance using clusters of video-watching behavior. Due to low accuracy in these classification methods, we tried using a Support Vector Machine and a Random Forest Model where the Random Forest Model offered better results. The Random Forest model takes the training data and models it by building a set of decision trees. These can be used to classify data. As a method to confirm all

these classifications, we implemented a neural network to determine there is truly no simple correlations between the student's video-watching behavior and a student's performance.

Results:

1) How well can the students be naturally grouped or clustered by their video-watching behavior ('fracSpent', 'fracComp', 'fracPaused', 'numPauses', 'avgPBR', 'numRWs', and 'numFFs')? In this analysis, all students that complete at least five of the videos were used.

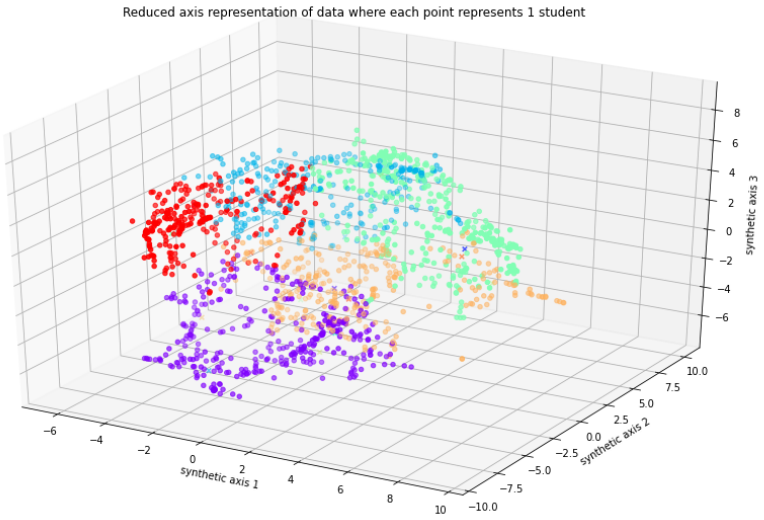

Figure 1: TSNE representing video-watching behavior features reduced to 3 dimensions

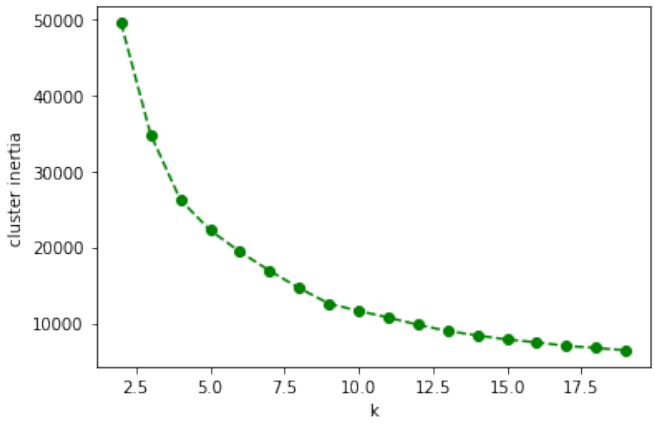

Figure 2: cluster inertia vs k (number of clusters) from TSNE model

Figure 1 allows us to visualize all the student's video watching behavior features reduced to just three dimensions to see whether all the features corresponding to a student can be naturally clustered. Figure 2 shows the effect of the number of clusters chosen, thus allowing us to deduce the elbow point which is n=6. Hence, this gives the number of clusters to use for K means clustering.

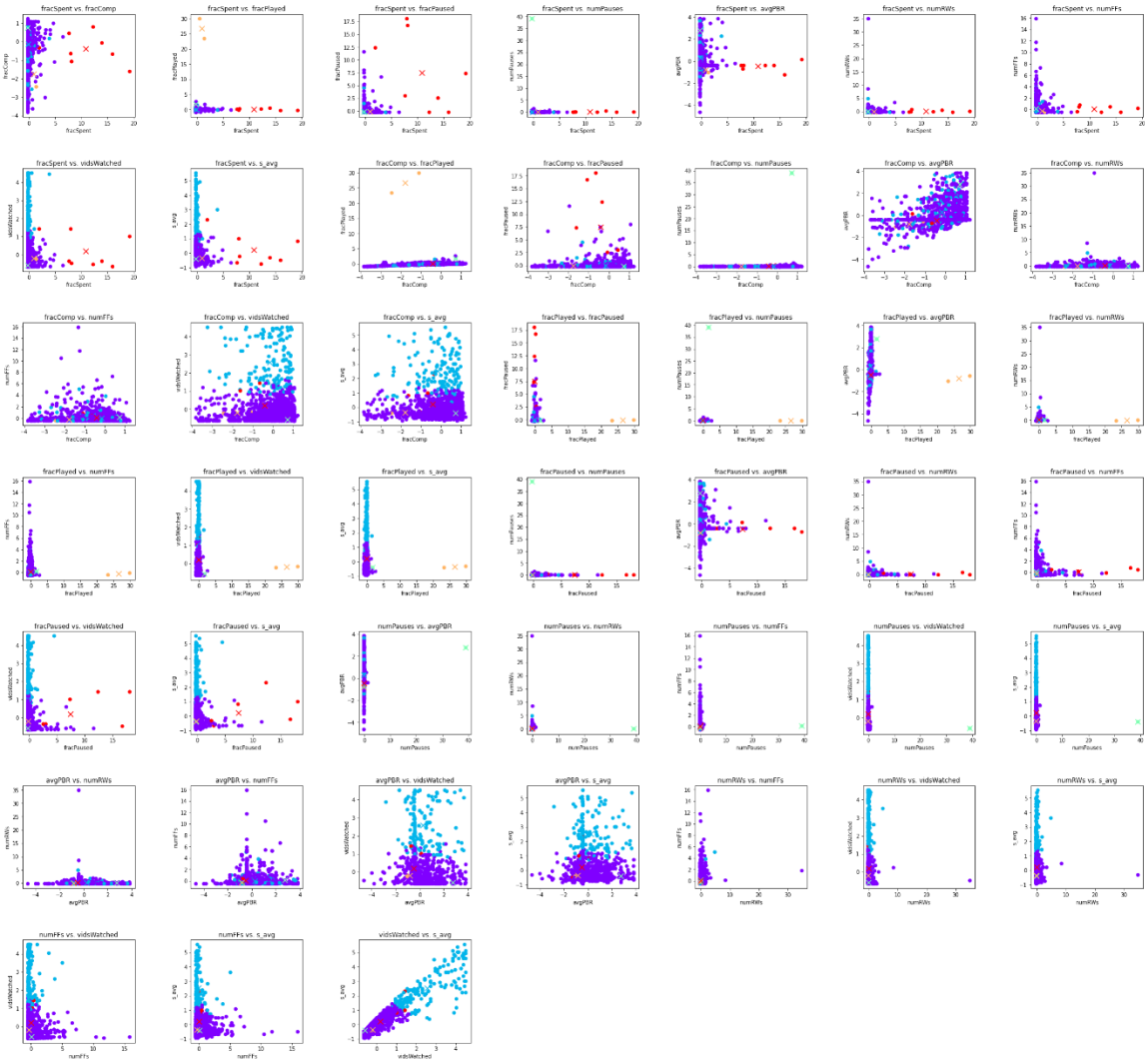

Figure 3: K means output plotting video-watching behavior features against each other

Considering the output from Figure 3, it is deduced that the students can be naturally grouped or clustered by their video-watching behavior. Specifically, 'fracComp' vs 'vidsWatched', 'avgPBR' vs 'vidsWatched', 'numFFs' vs 'vidsWatched', 'fracSpent' vs

‘vidsWatched’ can be clustered well together as distinct clusters are formed from Figure 3. ‘vidsWatched’ is the total number of videos that a student has watched from the online course.

2) Can student’s video-watching behavior be used to predict a student’s performance (i.e., average score ‘s’ across all quizzes)? This analysis could ultimately save significant time by avoiding the need for tests. In this analysis, all students that complete at least half of the quizzes were used.

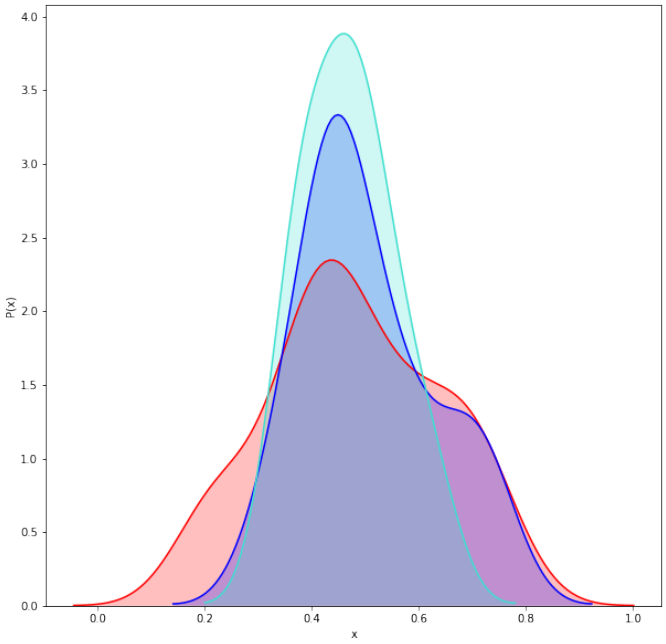

Figure 4: Polynomial Ridge Regression Probability Distribution

(Red: Training, Blue: Testing, Turquoise: Predicted)

Figure 4 shows how the output is distributed but does not contain information about the accuracy of the model. The distribution is to check whether there is bias in the prediction output. Since the probability distribution is not skewed in one direction, there is no bias in the model.

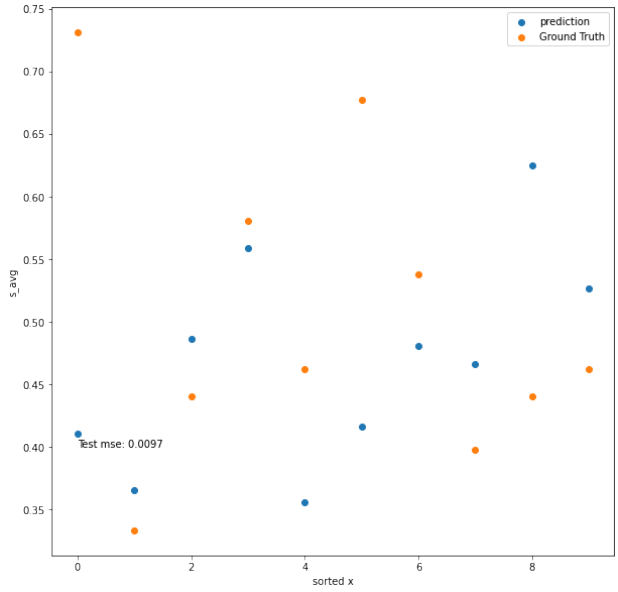

Figure 5: Polynomial Ridge Regression (sorted_x vs s_avg)

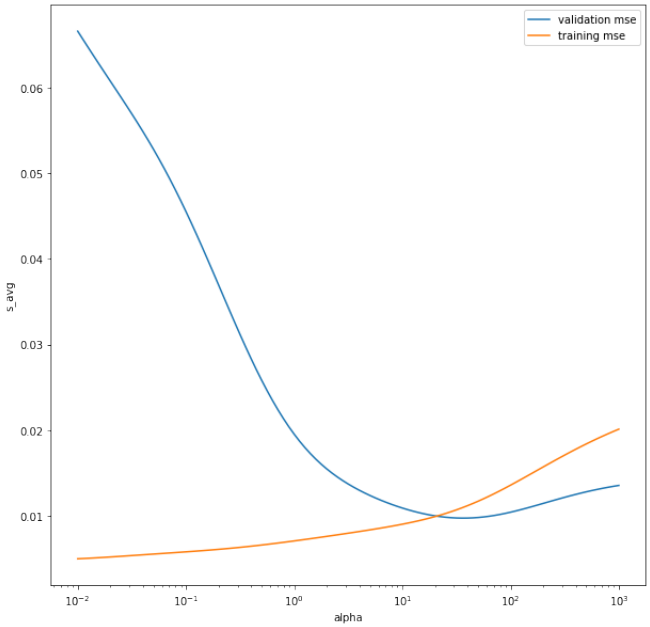

Figure 6: MSE for different values of lambda for Polynomial Ridge Regression

sorted_x is the values of the s column sorted, but the input features, are not sorted. s-avg is the average s score for each student. This sorted order is used train the model, so it is clear to see what the model has learned. Figure 6 shows the optimal point for lambda where MSE for both the validation and training data is minimal. Using this model, Figure 5 shows that the

validation MSE is low. It is concluded that student's video-watching behavior can be used to predict a student's performance.

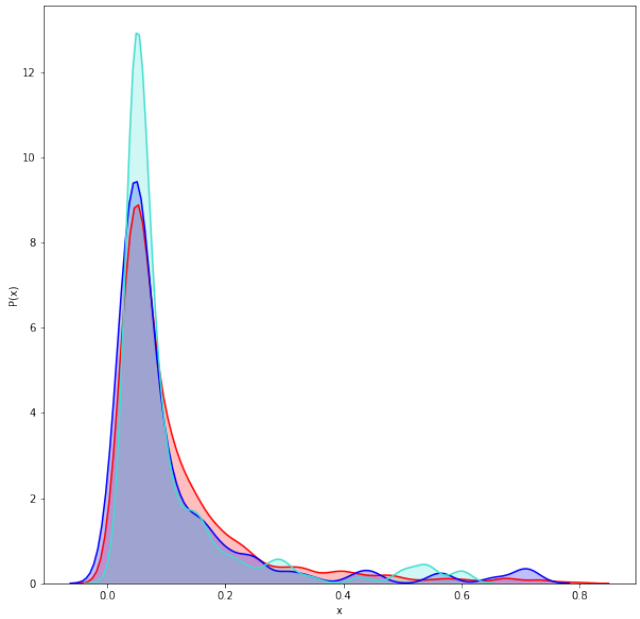

Figure 7: Neural Network Probability distribution

(Red: Training, Blue: Testing, Turquoise: Predicted)

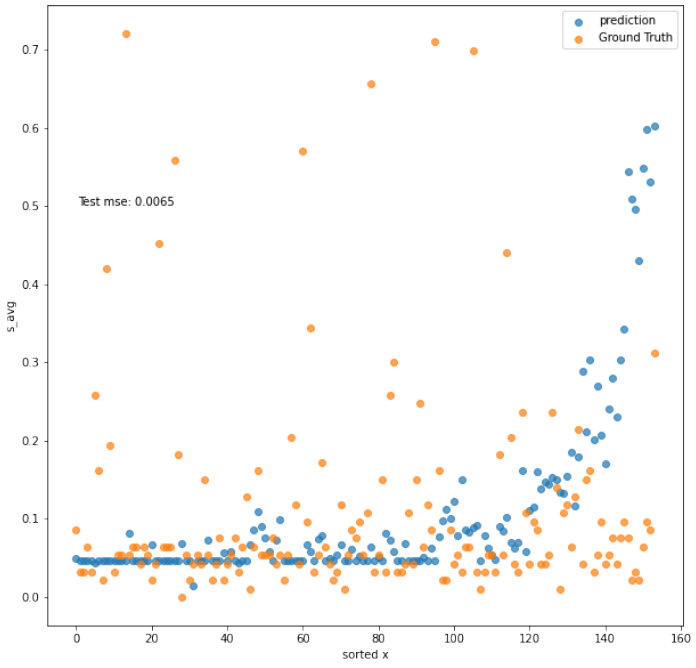

Figure 8: Neural network 'fracSpent' vs 's_avg' Predicted vs Actual

Considering Figure 7, it is seen that the probability distribution of the prediction output follows the probability distribution of the dataset. Figure 8 is used to determine the accuracy of the simple neural network. In both figures, the MSE was 0.0065, which further shows that the students' video watching behavior can be used to predict student's performance.

3) Taking step (2) a step further, how well can you predict a student's performance on a *particular* in-video quiz question (i.e., whether they will be correct or incorrect) based on their video-watching behaviors while watching the corresponding video? In this analysis, all student-video pairs are used.

Attempt 1: Linear Ridge Regression

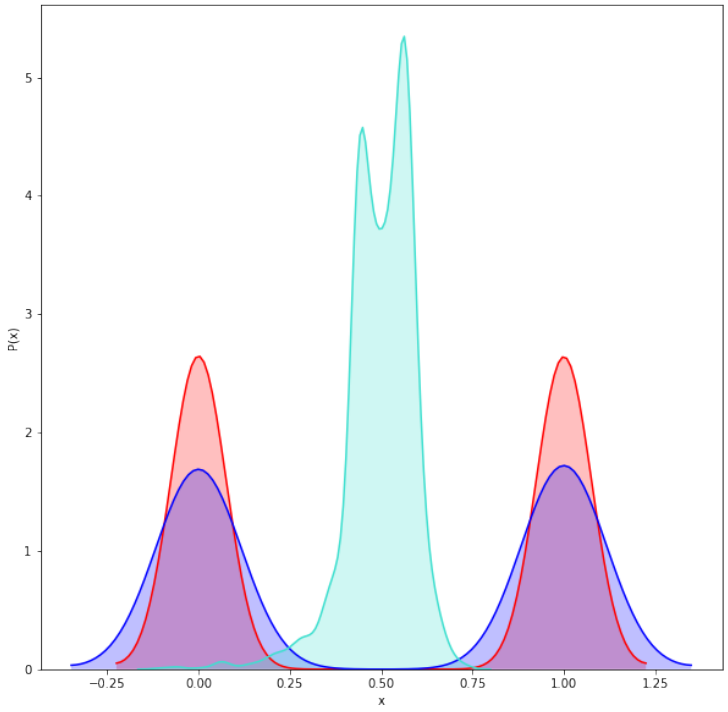

Figure 9: Probability distribution for Linear Ridge Regression

(Red: Training, Blue: Testing, Turquoise: Predicted)

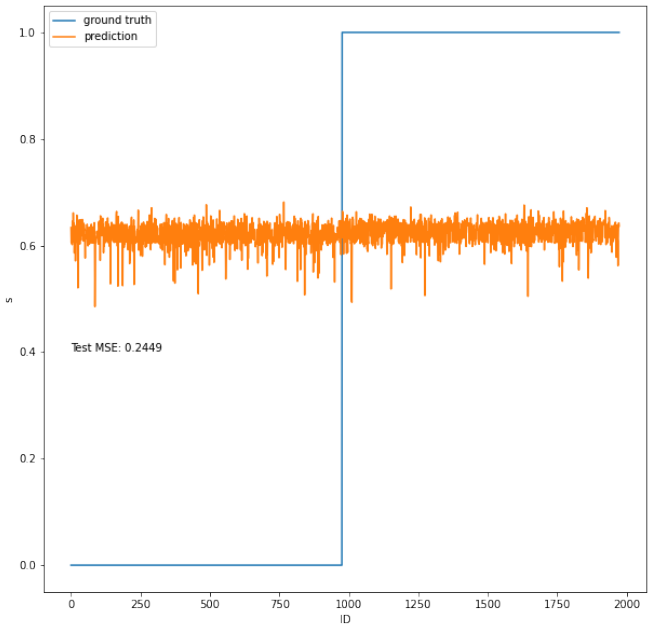

Figure 10: Linear Ridge Regression sorted_x vs s

From Figure 10, it is seen that the model did not learn anything about the dataset because the model treats the value for s as an arbitrary value (anywhere from 0 to 1). From Figure 10, all inputs predicted an output of 0.5 as this model offers the optimized MSE on testing dataset.

Attempt 2: Logistic Ridge Regression

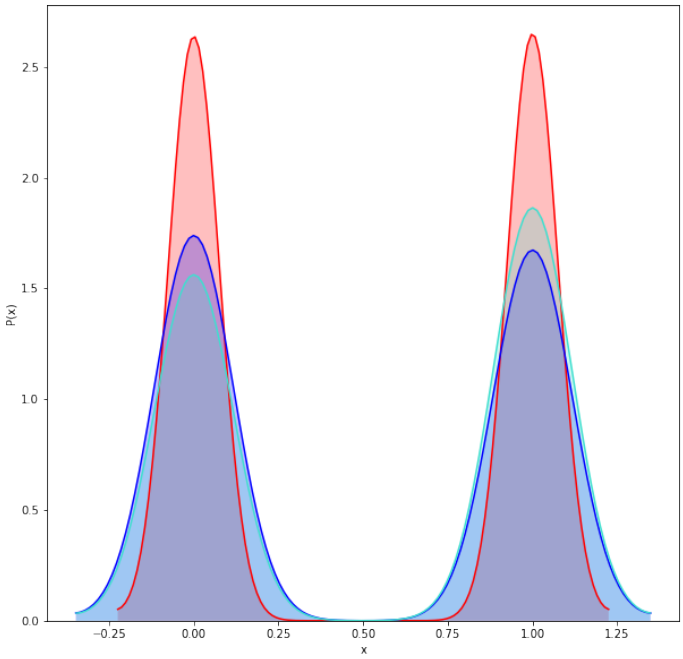

Figure 11: Logistic Ridge Regression Probability distribution.

(Red: Training, Blue: Testing, Turquoise: Predicted)

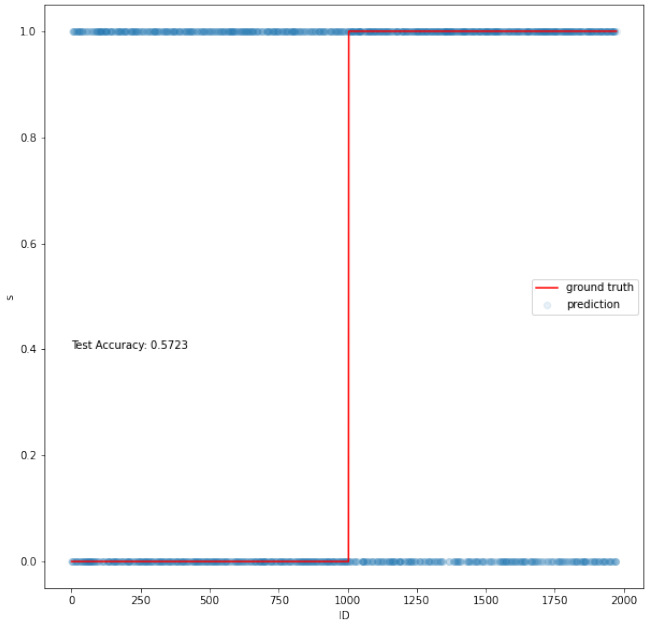

Figure 12: Logistic Ridge Regression sorted_x vs s

This was a better attempt as the model either predicts a 0 or 1 for s, but this model picks 0 and 1 half of the time as shown with the probability distribution in Figure 11. The problem arises when considering Figure 12, because it shows that the model arbitrarily chooses s and chooses not as a result of the input features. This model offers an accuracy of 57.2%.

Attempt 3: Gaussian Bayes

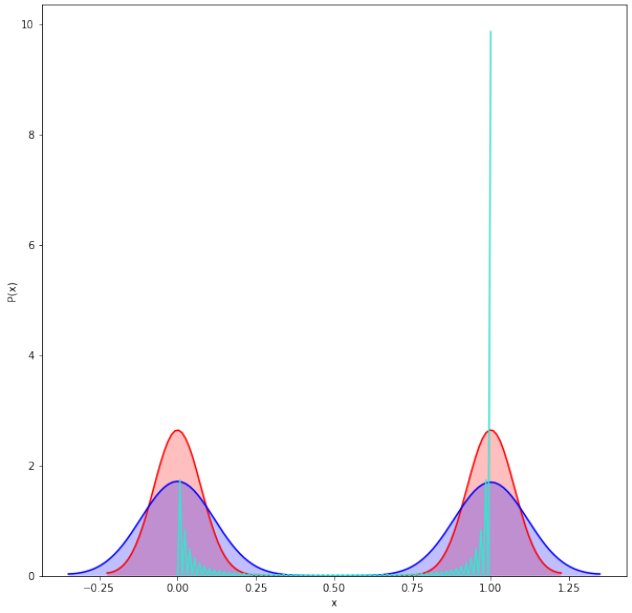

Figure 13: Gaussian Bayes Probability distribution.

(Red: Training, Blue: Testing, Turquoise: Predicted)

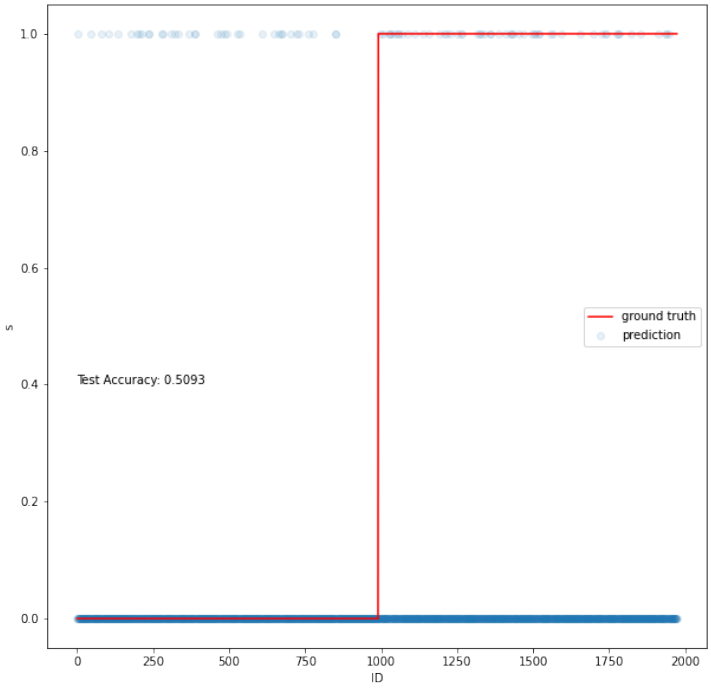

Figure 14: Gaussian Bayes sorted_x vs s

This model is less accurate than the Logistic Ridge Regression model as the accuracy is 50.93% compared to 57.2%. Figures 13 and 14 shows similar results to the Logistic Ridge Regression model for how both models predict value for s.

Attempt 4: Support Vector Machine (SVM)

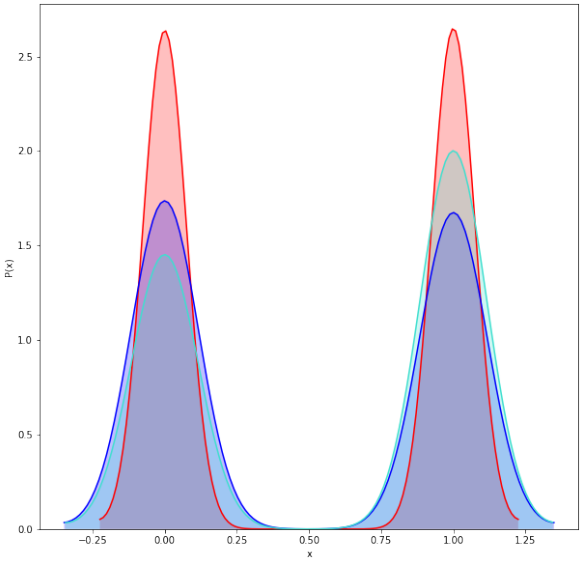

Figure 15: SVM probability distribution.

(Red: Training, Blue: Testing, Purple: Predicted)

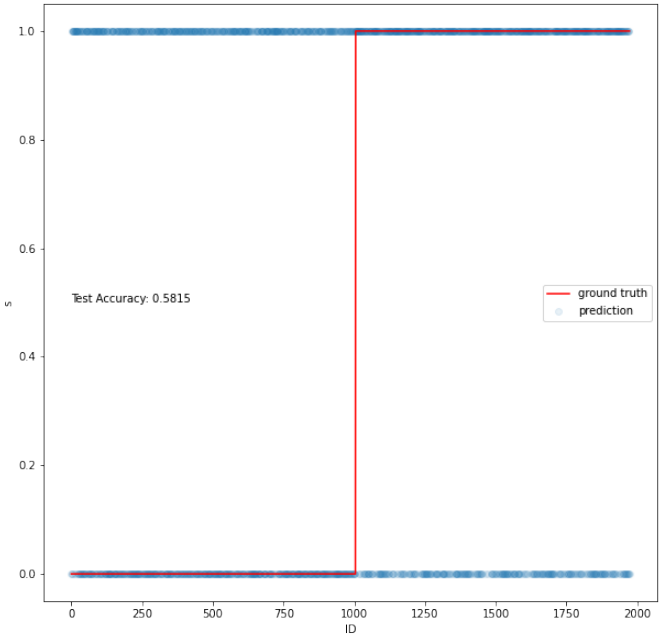

Figure 16: SVM sorted_x vs s

Attempt 5: Random Forest Model

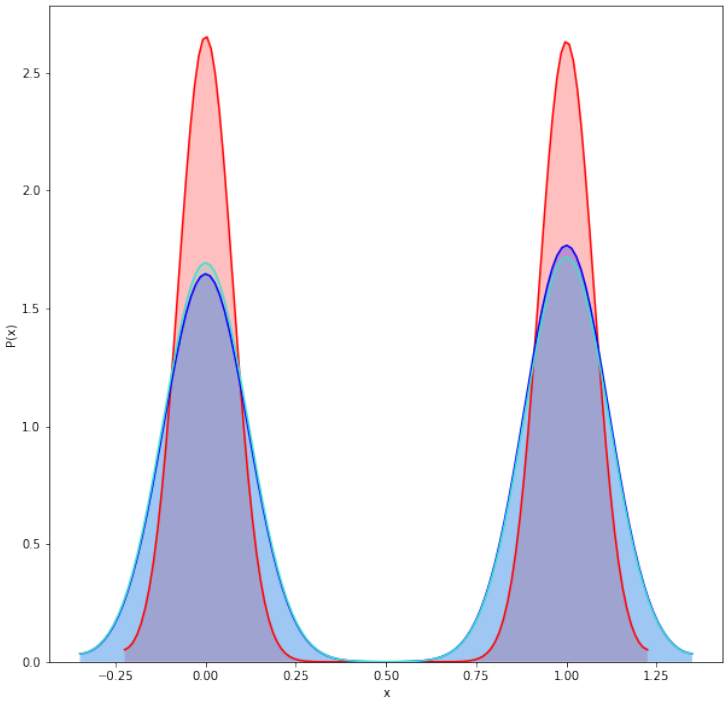

Figure 17: Random Forest model probability distribution.

(Red: Training, Blue: Testing, Purple: Predicted)

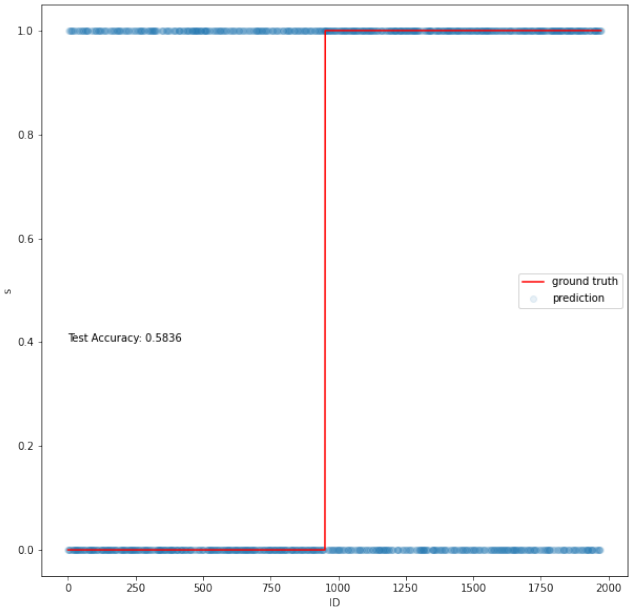

Figure 18: Random Forest model sorted_x vs s

Looking at both Attempt 4 and 5, both models were more accurate than the models used in Attempts 1-3. However, looking at Figure 17 and 18 shows that most the predicted values for s were incorrect as the accuracy was only 58.15% and 58.36%.

Attempt 6: Neural Network

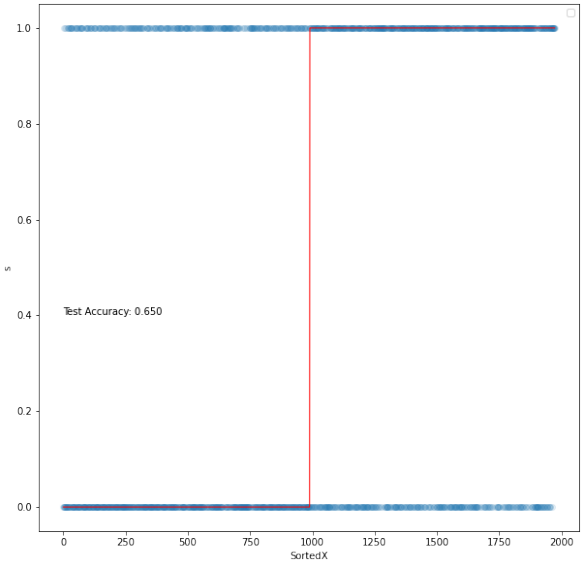

Figure 20: Neural Network model sorted_x vs s

As a final attempt, we tried comparing each testing data point to the training data one by one to see if a prediction could be made, and this yielded a 99.9% accuracy. Essentially, we viewed the training data as the model and we just used Euclidean distance to find the closest data point. The model is clearly over fit, as no generalizations were be made, but it clearly shows that a relationship exists in the data. This led us to try a last-ditch approach of building a Dense Forward Feeding Neural Network. As seen from Figure 20, the test accuracy is only 65% so there were many instances where the model predicted the s value incorrectly. This shows that with in the data set there exists no simple linear or semi-linear relationships in the data.

When viewing the results from all of the attempts (1-7), it is evident that we cannot predict a student's performance on a particular in-video quiz question (i.e., whether they will be correct or incorrect) based on their video-watching behaviors while watching the corresponding video. It can be deduced that looking at each student's video-watching behavior on each video,

there is no simple or general way to predict whether they will get the question at the end of the video correct. Based on the number of models tested, is clear that there is no correlation and no simple correlations can be extracted from the dataset. Now the question becomes why is there no correlation? Basic intuition would tell us that a student who watched a higher fraction of a video, or spent more time watching a video would be more likely to get a question correct. However, from the data we were able to extract and the original data we were given, it is clear that there are other more external factors that influence a person getting a question correctly. While video watching behavior can suggest that a student is invested in that videos, it does not have the ability to show whether a student is understanding the material. In other words, spending more time does not mean having more understanding. This also makes more the results of problem 2. Problem 2 suggests that a student's performance can be inferred from their overall or average watching behavior. This indicates that looking at a student's average scores, average watching behavior, and total number of videos watched can be used to make more general predictions. When looking at Problem 2 compared to Problem 3, this result makes sense, because consistently in the amount of effort a student puts forth can be linked to the ability to perform on quiz questions. Furthermore, a student who watches more video is more likely to get more questions correct merely because you answered more questions. Finally, using the average of students watching behavior can help filters out external features that may lead to variability in the dataset. For example, a student watches all 93 videos, and get 80 percent of the questions correct. Assuming that on the days this student missed a question, his watching behavior was wildly inconsistent, the variations in the watching behavior would have less bearing on the average scores because, for about 75 of the 93 videos, this student's behavior was consistent. Similarly, if the student's behavior is consistent on the days, they missed the questions, the external factors that may have caused the student to miss the question have less effect on the student's overall performance, once again because there is a sense consistency on 75 of the 93 videos. In problem 3, the classification is too arbitrary. There is no sense of a general trends in the performance that a model could pick up on because each individual video for each student is viewed as a singular independent data point. As a result the external factors that affect each data point have more effect on the model's ability to pick up on a trend. Now we are not suggesting that no trend exists at all, as not all possible models have been tested. We are simply suggesting that considering

each data point individually results in a very noisy data set that a simple model will not be able to learn from.

Potential Future Options:

Based on the results we have found above, there are a few possible directions that can be tried in the future. Given that viewing each data point individually (Problem 3) obscures trends, but viewing a student's aggregate performance tends to highlight trends (Problem 2), using a RNN or an LSTM neural network as the model, and inputting he current watching behavior and a sequence of the student watching behavior on prior videos, could be used to more accurately predict the probability that a student will get the end of video question correct. By using a sequence for prediction, you are potentially giving the model more information about the student to base its prediction on.