UNIVERSITY OF CALIFORNIA

Los Angeles

Racial Bias in
Machine Learning Algorithms
in Secondary Mathematics Education

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Applied Statistics and Data Science

by

Suyeon Hwang

2024

ABSTRACT OF THE THESIS

Racial Bias in
Machine Learning Algorithms
in Secondary Mathematics Education

by

Suyeon Hwang
Master of Applied Statistics and Data Science
University of California, Los Angeles, 2024
Professor Ying Nian Wu, Chair

This paper examines racial bias and discriminations in machine learning algorithms using America's longitudinal high school students dataset. This study reveals machine learning algorithms may present a seemingly fair accuracy for both White and Asian student group and Black and Hispanic student group, but underneath the surface, the machine learning algorithms consistently produce a higher false positive rate for the White/Asian student groups while it consistently underestimates Black/Hispanic student group's 12th grade math performance. This paper provides a comprehensive analysis and comparison of seven commonly used machine learning algorithms' performances in terms of biased results towards the White and Asian student groups versus Black and Hispanic student groups.

The thesis of Suyeon Hwang is approved.

George Michailidis

Nicolas Christou

Ying Nian Wu, Committee Chair

University of California, Los Angeles

2024

To my former boyfriend and now fiancé, . . .

who—supported me from the beginning

of this academic journey when it only was a dream—

fervently encouraged and cheered me on throughout this program.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

With the rapid development of machine learning algorithms and artificial intelligence, society has been relying more and more on these tools to make and implement these predictions in the decision making process across different sectors and industries ranging from criminal justice, marketing, medical field, to education to just name a few. However, the fairness and bias of these seemingly objective mathematical models have been questioned and proved to be not so objective and unbiased in the past few years[1].

For instance, in 2016 the COMPAS algorithm used in the U.S. for predicting criminal recidivism was found to be biased against African American individuals where it falsely labeled them as higher risk than white individuals [2].

Researches have shown that there are reasons why machine learning makes biased results. In fact, the challenges of promoting fairness and overcoming biases of data can be categorized into two big factors: first, considering data used as inputs to an algorithm and second, the inner workings of the algorithm itself [3]. First, the sources of bias on input data could be due to multiple factors such as poorly selected data, incomplete, incorrect, or outdated data, selection bias, and last but not least unintentional perpetuation and promotion of historical biases. In each case, it is imperative to keep transparency and accountability during the entire data analysis cycle. Secondly, algorithms can be biased when an algorithm systematically favors one outcome over another [4].

Adaptations and implementations of machine learning and artificial intelligence in the educational sector has been a rising topic. In the classroom setting, current applications of AI are tutoring, personalized learning, testing, and automating tasks [5]. However, policymakers and educators are trying to adopt machine learning algorithms beyond the classroom and implement the ML algorithms in a bigger decision making context. In 2020, the United Kingdom used an algorithm to estimate exam results, however, the calculations favored elites and white students [6].

This paper is an extension and a comparison of an already existing paper, 'Who Gets the Benefits of the Doubt?', [7] which examines the fairness of various machine learning algorithms in secondary mathematics education context using a longitudinal study dataset from the National Center of Education Statistics to predict students' mathematical performance based on 60 features. All machine learning algorithm models that are used in this paper consistently produce a higher probability to White and Asian students of being in the top 50% in their math scores over Black and Hispanic students. Similarly, all machine learning algorithms consistently predict Black and Hispanic students are more likely to be in the bottom 50% of their math scores than reality.

The original paper suggests when it comes to determining whether a model is fair or not, accuracy alone could be a deceptive metric as it can overlook false-positive rates and false-negative rates, which matter greatly in terms of giving a benefit of doubt or falsely misclassifying students. Hence, in this paper, an additional metric F1-score is introduced and used to closely monitor the fairness of each model as F1 score is the harmonic mean of both precision and recall and a better metric especially with the imbalanced data.

Furthermore, this paper discusses results from seven different machine learning models and suggests future recommendations of adapting machine learning and artificial intelligence into the education system while minimizing racial bias.

CHAPTER 2

Data

2.1 Background of the Data

The original data is from the High School Longitudinal Study of 2009 (HSLS:09) from 2009 to 2016, which is one of the series of National Center of Education Statistics (NCES). This study monitors the transition and changes in national samples of students from their high school years through their postsecondary years. In the base-year, 2009, of this study, students were sampled randomly through a two-stage process, where the first part was a stratified random sampling and school recruitment. This gave 1,889 eligible schools, and a total of 944 of these schools participated in the study. During the second stage of sampling, 25,206 ninth-grade students were randomly sampled from these schools. At the end, 23,000 ninth-grade students from 944 schools were in this study. First follow-up happened in 2012, when the students were in their senior years of high school and the second follow-up happened in 2016, four years after they graduated from high school.

The original HSLS:09 data is made of multiple parts: student questionnaire, parent questionnaire, mathematics and science teacher questionnaire, school administrator questionnaire, counselor questionnaire, and mathematics assessment in algebraic reasoning.

However, in this paper the data that was used has only 60 features from 10,000 variables, which were chosen including students' 9th and 12th grade mathematics performance, 30 Parent information from parents' surveys (variables that start with P1), 14 student features (variables that start with S1), 13 math class information (variables that start with X1).

Data related to math scores is from the math assessment that uses Item Response Theory technique (multiple-choice) questions and it is a criterion-referenced measure of achievement at the time of the base-year assessment. This criterion is represented by the 188 items that determined the score for the X2X1TXMSCR variable. The assessment encompasses algebraic skills, reasoning, problem solving for 9th and 11th graders. Specifically, algebra proficiency scores were composite of algebraic expressions, multiplicative and proportional thinking, algebraic equivalents, systems of equations, and linear functions.

As this paper is an extension of the existing paper [7], the data that was used in this thesis is from the author of the existing paper, which has already been cleaned and processed. The data that was used in this paper has 16,633 observations with 60 features. 25 student variables including student's race, gender, their math experience, their first language, their math course grade, and their perception of math class. 35 items from the parent questionnaire were selected.

2.2 Data Cleaning

Originally, the data had negative values for all non-response or sensitive data, so these values were computed to NA. In this way, imputation could work. For imputation, the K-Nearest Neighbors method was used to impute missing values. Rows with missing 12th grade math score values (X1TXMSCR variable) were dropped as this is the target variable. The data then was transformed by standardizing and any categorical columns such as student gender and student race got converted to numerical values.

For all non-responses or sensitive blocked data got marked as "None", so the imputation could work. Missing or sensitively blocked data values were preprocessed by K-Nearest Neighbor method.

For the 9th grade and 12th grade math scores, students who were above 50th percentile were marked as 1 and bottom 50% were marked as 0.

Historically underrepresented students - Black and Hispanic- are grouped as BH or Black and Hispanic while White and Asian students are marked as WA throughout the data analysis. For binary classification purpose, the BH group is marked as 0 and the WA group is marked as 1.

CHAPTER 3

EDA

A few important observations have come up during the Exploring Dataset Analysis part. One of the most prominent factors to consider from EDA is the distribution of students' races. More than half of the students are White and there is a very low percentage of Hispanic and African American students.

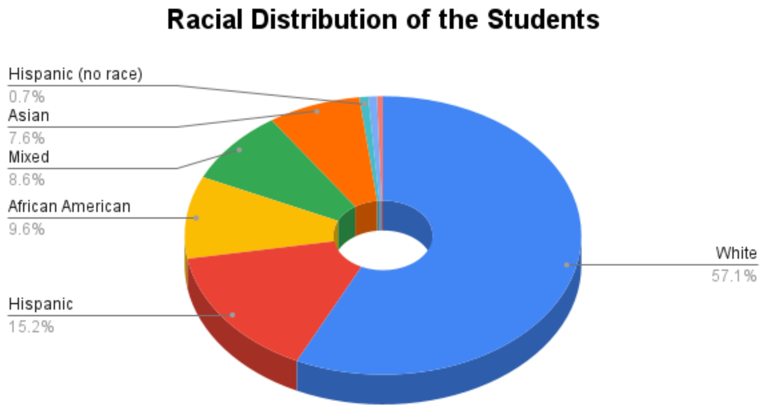


Figure 3.1: Piechart of the Racial Distribution

Once Black and Hispanic students make up the BH group and White and Asian American students make up the WA group, WA group almost three times bigger than the BH group.

Another observation is that out of students who are in the bottom 50% in their math performance, a majority of the bottom 50% group students are in the BH group. For black students, almost 70% of the students performed in the bottom 50; this is an important piece of information, as the 9th grade math performance variable is the most highly correlated

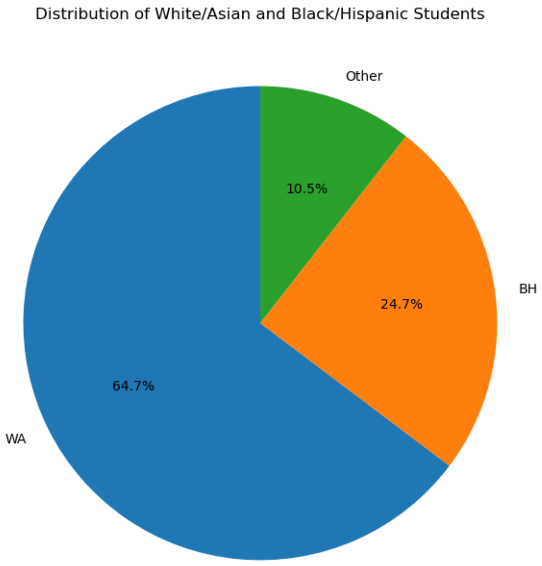


Figure 3.2: Distribution of the WA and the BH Group

with the target variable – 12th grade math performance. On the contrary, even though Asian American students only make up 7.5% of the data, a majority of the Asian Americans have performed in the Top 50 in their 9th grade math performance. Given that this is the data that is being fed into the models, one must keep in mind that one of the sources that ML creates biased results is data with an underrepresented population; one needs to keep in mind that the data already has a disproportionate amount of black and Hispanic students are in the low 50% group whereas a majority of the Asian American students in the WA group are in the top 50% performing group.

Through correlation coefficients, 9th grade math performance and 8th grade math performance as well as parent's education levels and family income are the most correlated variables with the target variable - 12th grade math performance. Generally, students' educational success is often most highly correlated with a student's socioeconomic status. In fact, one's race is often highly correlated with socioeconomic status. In fact, one of the most critical factors behind schooling disparities is socioeconomic status [8]

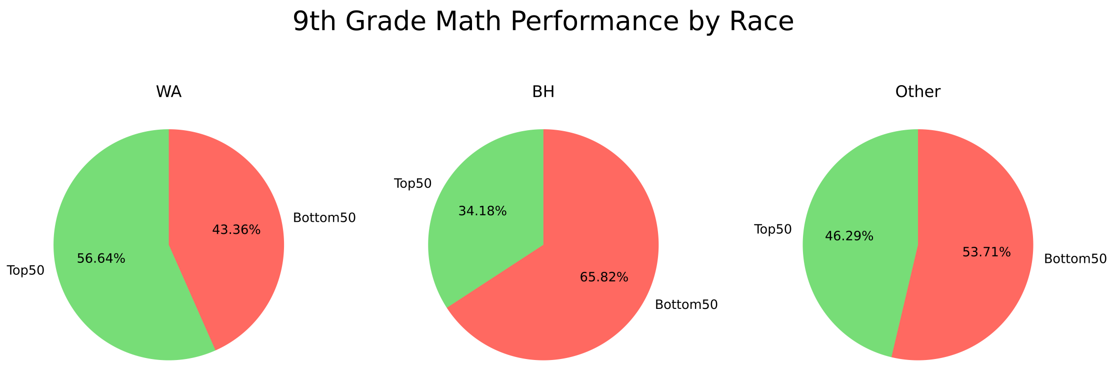


Figure 3.3: Distribution of 9th Grade Math Performance by Race.

Socioeconomic status has a high level of effect on student achievement [9]. For instance, students from a more affluent socioeconomic background can afford to have tutors or more academically suitable home environment such as students having their own room, desk, and computers whereas students from low-income households do not have a quiet environment at home to do their homework or focus on their academics. Overall, it is well documented across multiple academic articles that there is a strong association between students' socioeconomic measures and their academic outcomes [10].

Top 10 features that are most relevant to student race are Parent 1 and Parent 2 Race, whether parent 1 is born in the U.S., family income, parent 1's birth year, student's math interest, parent 2 occupation status, student's first language, and parent 2's highest education degree. In other words, these features are implicit racial features. Based on these most relevant top ten features to race, a subset of data is formed: a subset of data without racially implicit features In order to see how much race and implicit racial features are affecting predictions on student's 12th grade math performance, each model are trained and tested on these subset data as well as all data. At the end, these subset datasets are trained in machine learning algorithms and their results are compared between results with the full datasets.

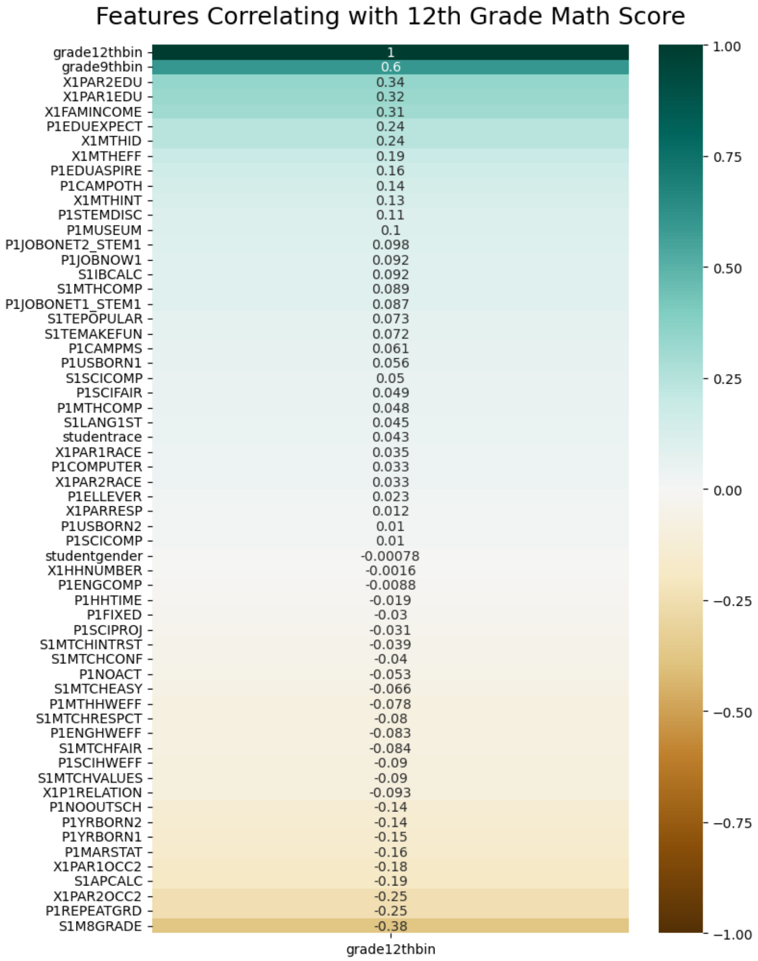


Figure 3.4: Correlation Coefficients of the Features with the Target Variable - 12th Grade Math Performance

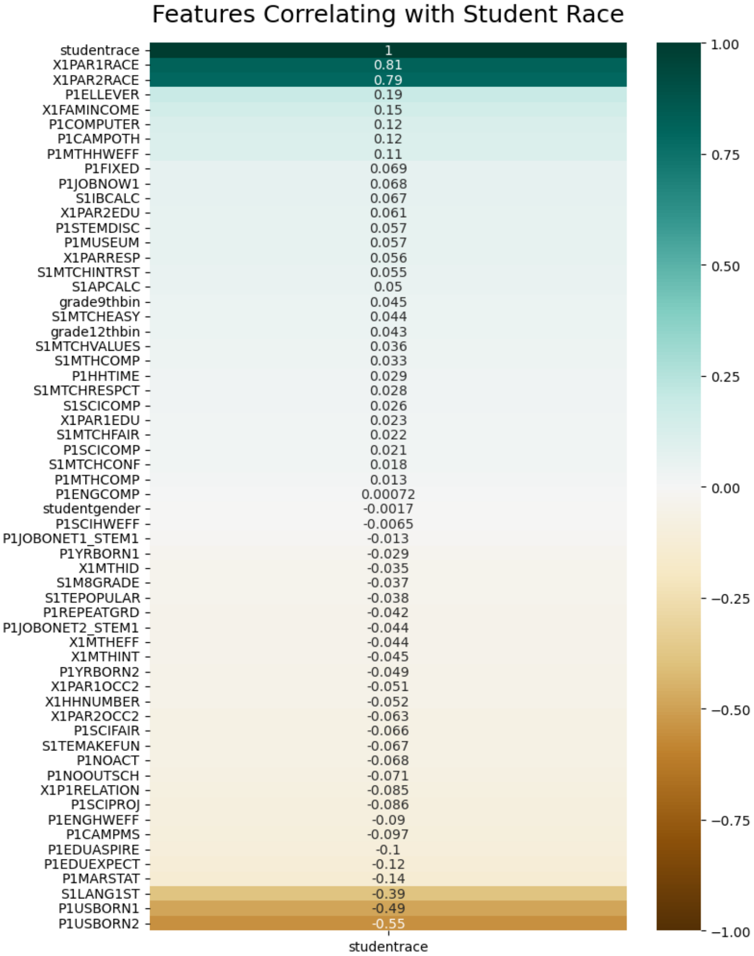


Figure 3.5: Correlation Coefficients of the Features with Race

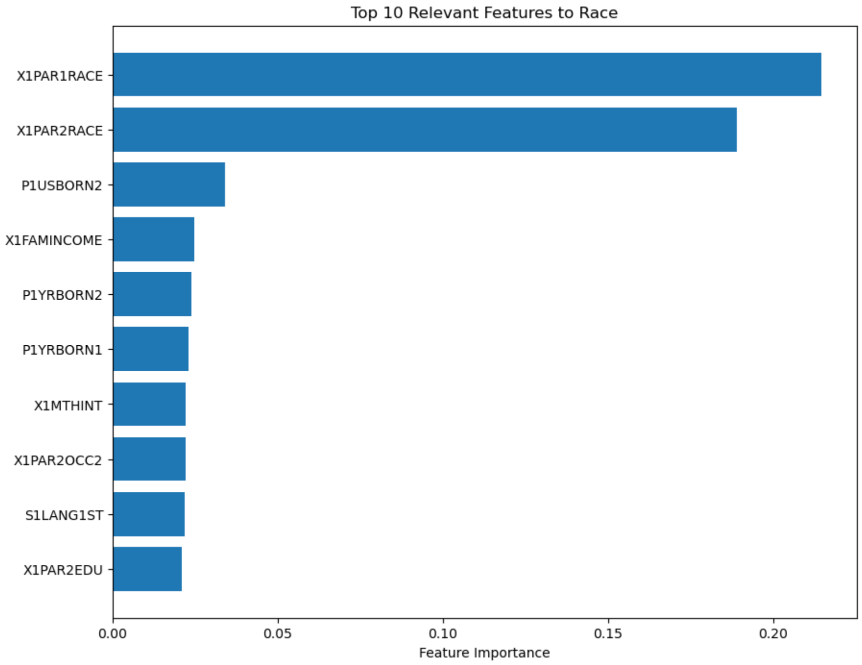


Figure 3.6: Top 10 relevant features to race through feature importance scores through random forest

CHAPTER 4

Methodology

Seven commonly used machine learning algorithms were used. For each model,
(1) models are trained and tested by the entire data once,
(2) models are trained and tested by the entire data 30 times by cross-validation, and
(3) models are trained by the WA group and BH group separately 30 times and tested by the WA and the BH group 30 times.

When models are trained by the data 30 times, the results are appended and the average of each metrics were calculated with the standard error. This was repeated with the subset of data, which is a data without the racial implicit features. Lastly, this method was repeated with the balanced data, which is consisted of the same number of the WA and the BH students to consider the imbalanced input data.

4.1 K-Nearest Neighbor

K-Nearest Neighbor (KNN) is a non-parametric method which means it does not make any assumption about the underlying distribution or structure of the data. KNN often is used for classification. The algorithm calculates the Euclidean distance between each point and all the data points in the training set then it selects k closest data points – neighbors– and assigns the class label through a major voting process which means it is most common among the k nearest neighbor.

In this analysis, the KNN method was first used with default hyperparameters as well as with the best hyperparameter after tuning the parameters through comparing the accuracy and error rate with different numbers of neighbors.

There are a few hyperparameters that can be tuned in order to improve the model's performance. One is the number of the neighbors that will be used to determine the prediction. Based on the accuracy of the testing and the training data, the best number of neighbors to be used is 11. The small number of neighbors means high complexity, high variance, and low bias while the large number of neighbors signifies low complexity, low variance, and high bias. In practice, values of k between 3 to 15 are reasonable choices [11]/

It was also done the second time through checking the error rate. Throughout the analysis, 11 neighbors were used.

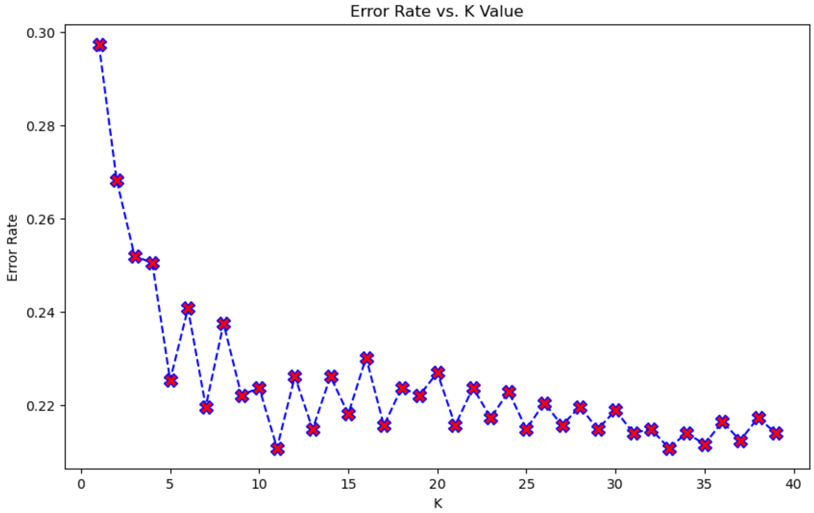


Figure 4.1: Error Rate vs. Number of Neighbors (K Value) The lowest error rate is obtained at k = 11.

4.2 Logistic Regression

Logistic regression is one of the most important analytics tools in the social sciences [12] and it is another widely used statistical model for classification problems where it fits a s-shaped curve to the training data. Specifically, we fit the training data to this function:

$$P(X) = \frac{1}{1 + e^{-\beta_0 - \beta_1 X_1}}$$

And this probability function outputs a value between 0 and 1. For a given x value, if the probability

$$P(y = 1 \mid x)$$

is more than 0.5, we consider that as a yes if not then no. This 0.5 is called either a decision boundary or a threshold value.

$$\text{decision}(x) = \begin{cases} 1 & \text{if } P(y = 1 \mid x) > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

At the end, based on the probabilities and a threshold value, the final classification is made. One note to make is that logistic regression is considered a generalized linear model because the outcome always depends on the sum of the inputs and the parameters.

When using logistic regression, there are a few parameters that could be tuned to enhance the model's performance. First is a regularization parameter (c), which is a value that determines the strength of regularization. Regularization is giving a penalty when the model overfits; larger values of c reduce the regularization, allowing the model to fit the training data more closely, which could increase the risk of overfitting. Smaller regularization values imply stronger regularization, which means the model gets penalized more heavily when overfitting, but this could lead to under-fitting if set too low. The cost function for logistic regression is the following:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(h_\theta(x^{(i)})\right) + \left(1-y^{(i)}\right)\log\left(1-h_\theta(x^{(i)})\right)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

Another parameter for logistic regression is a type of regularization. There are a total three: L2 regularization (Ridge regression), L1 regularization (lasso regression), and a combination of the two – elastic-net. Last parameter that will be described is a solver, which is an algorithm that is used for fitting the model: liblinear, saga, and lgfgs.

Some of the default hyperparameters that are used in the SciKit logistic regression function is the L2 Regularization and lbfgs.

The parameters were tuned through a grid search method. The training data for the entire data was used to find the optimal parameters. Saga, ridge regression, c-value of 0.1 and 300 max iterations were chosen for the analysis.

4.3 Support Vector Machine

SVM is a generalization of a simple and intuitive classifier called the maximal margin classifier, which is a hyperplane that maximizes the distance between the line and the closest observation points in each class of linearly separable data.

However, the maximal margin hyperplane is extremely sensitive to a change in a single observation point that may result in overfitting the training data. Instead of perfectly aiming for separating the data points at the cost of overfitting, a support vector classifier is used, which prioritizes greater robustness to each observation as well as better classification for the overall data. In case, the observations are not separable, then a soft margin, or a soft margin classifier, which is also called as the support vector classifier. Support vectors are the data points that are closest to the decision boundary or hyperplane, and these support vectors help define the position of the hyperplane. Generally, SVMs are intended for the binary classification problem.

4.4 Random Forest

Random forest is a classifier consisting of multiple tree structured classifiers where the trees are independent identically distributed random vectors where each tree votes for the most popular class at input x.[14] Random forest can be used for both classifications and regressions as it can average the prediction of each tree. Due to the fact that this model averages the results from many trees, random forests are less likely to overfit compared to a single decision tree. Additionally, random forests are less sensitive to noise and outliers.

4.5 Ensemble Method

Ensemble method in machine learning is a technique that combines the predictions of multiple models to produce a more accurate and robust prediction than what a single model could achieve on its own. Through combining multiple models, the strengths of each model can be leveraged while minimizing their individual weaknesses, leading to better overall performance; this model ensures improved accuracy by reducing the variance and bias of the model. As a result, ensemble methods could be more robust to overfitting and noise in the data, however, the model is a lot more complex and computationally expensive.

In order to choose what classifiers to ensemble for this classifier, nine different classifiers were tested with cross-validation; each classifier trained on all data and cross-validation scores were compared with the means and standard deviations.

At the end, five classifiers were chosen based on the best cross-validation means: Gradient Boosting, Logistic Regression, Extra Trees, Random Forest, and Support Vector Classifier. Afterwards, hyperparameters were tuned by the search grid, where it once again trained on the features and looked for the best estimators. Finally, the voting classifier, which is made of the five classifiers that were chosen earlier with the tuned parameters, was trained by all data and it used soft voting to make the final prediction. In soft voting, the output class is

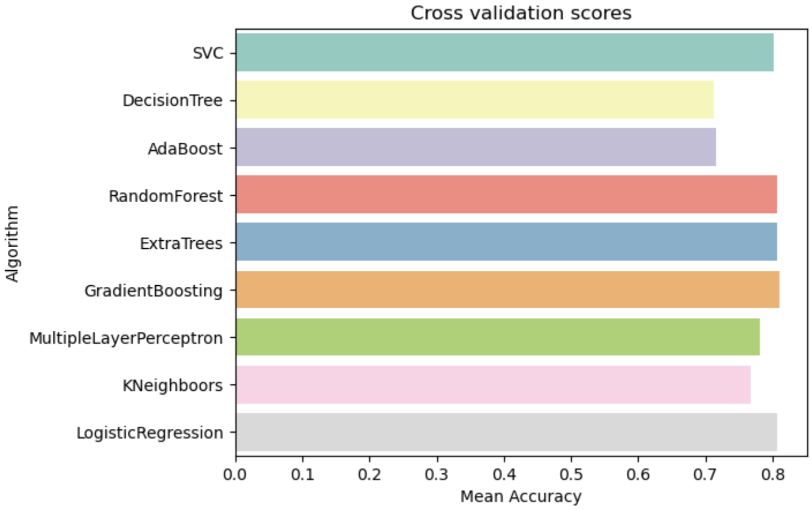


Figure 4.2: Cross-validated mean scores for choosing different algorithms for ensemble method.

Rank	CrossValMeans	CrossValErrors	Algorithm
1	0.810433	0.0067	GradientBoosting
2	0.807275	0.007706	LogisticRegression
3	0.806599	0.007568	ExtraTrees
4	0.806374	0.008376	RandomForest
5	0.800961	0.007225	SVC
6	0.780442	0.005507	MultipleLayerPerceptron
7	0.767663	0.006514	KNeighbors
8	0.716326	0.014001	AdaBoost
9	0.712494	0.013037	DecisionTree

Table 4.1: Cross-Validation Means and Errors by Algorithm

calculated by the average of the probability given to that class. Each base model classifier independently assigns the probability of likeliness of each type and the final predictor is the class having the highest average probability.

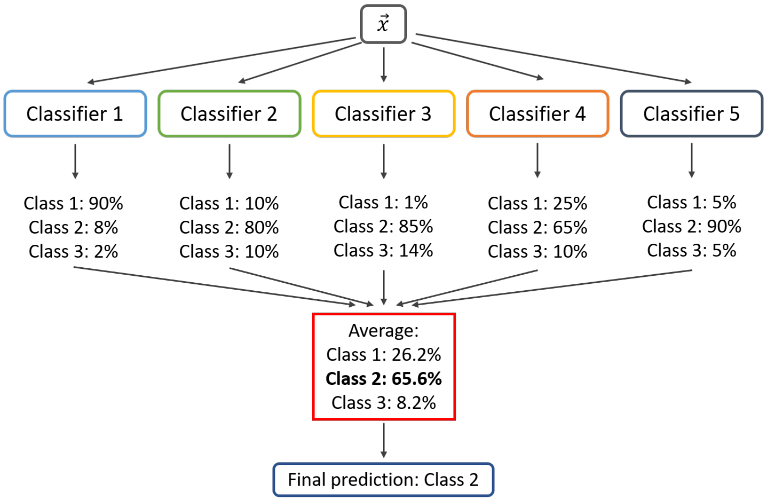


Figure 4.3: Illustration of how soft voting in ensemble method works

Next, hyperparameters were tuned for each model. For the random forest model, the parameter grid was used with 9-fold cross-validation and based on the accuracy, 54 combinations of parameters were evaluated. In other words, 54 parameter sets went through 9 training and testing procedures, leading to 486 total fits. 10 max features and minimum 3 leaves were selected.

4.6 Neural Network

"Neural network is a type of artificial intelligence that attempts to imitate the way a human brain works" [13]. The feed-forward neural network with one hidden layer is one of the most commonly used type for regression-like modeling applications [14].

$$x_o = f_o\left(\sum_{\text{inputs } i} w_i x_i\right)$$

“f_0 is called the activation function and some standard choices include, identity, logistic, and indicator functions. The w_i are weights, which are usually uninterpretable. The Neural Network learns the weights from data. Neural networks are effective for large complex datasets but they can often overfit without careful control,” [14]/

In other words, each output of the neural network model is a sum of some weights and input feature. In more rigorous form, it takes the form:

$$y_o = \phi_o \left(\sum_h w_{ho} \, \phi_h \left(\sum_i w_{ih} x_i \right) \right)$$

ϕ_h notates the activation functions for the hidden layer, which is often logistic function.

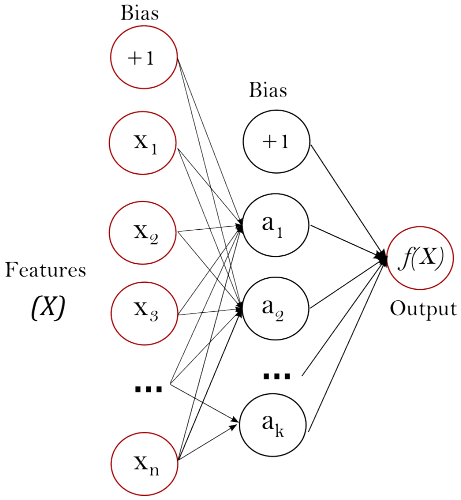
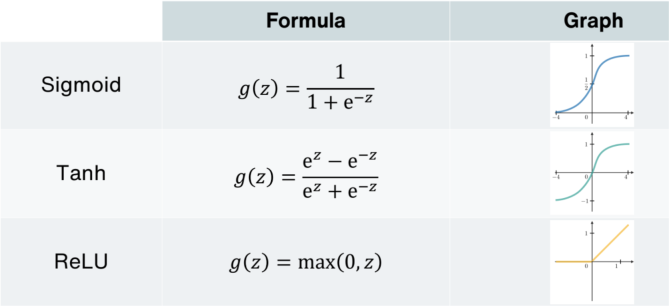


Figure 4.4: Illustration of how neural network works with one hidden layer (Scikit Learn 2007)

In order for the neural network to work, data needs to be standardized. At first, the model was performed with default parameters: 32 hidden layers and 16 second hidden layers with ReLu and sigmoid function for the final output layer. The batch size of 10 was used over 50 epochs. There are a few choices of common activation functions, but the most commonly used one for the final output is the ReLU and sigmoid functions for hidden layers.



	Formula	Graph
Sigmoid	$g(z) = \dfrac{1}{1 + \mathrm{e}^{-z}}$	
Tanh	$g(z) = \dfrac{\mathrm{e}^{z} - \mathrm{e}^{-z}}{\mathrm{e}^{z} + \mathrm{e}^{-z}}$	
ReLU	$g(z) = \max(0, z)$	

Figure 4.5: Commonly used activation functions for neural network

Some common hyperparameter tuning techniques are grid search, random search, and bayesian optimization. In this case, grid search was used, since it's one of the most simple strategies. Models were trained now with tuned hyperparameters for all data, WA data only, and BH data only.

Through the grid search process, for all data and BH data, it recommended batch size of 64, epochs of 10, and optimizer of adam. For WA data, it recommended batch size of 64, epochs of 10, and optimizer of rmsprop.

4.7 XGBoost

Extreme gradient boosting, which is also called as XGBoost, is another type of ensemble supervised machine learning algorithm that can be used for both classification and regression problems. XGBoost is a type of gradient boosting method, but it is different in a few ways. XGBoost uses both L1 and L2 regularization that can reduce overfitting and is usually faster than gradient boosting due to the parallelization of tree construction. It is also known for its ability to handle missing values within a data set. [7a]

XGBoost model performance could be enhanced by tuning the hyperparameters. Once again, grid search method was used to find the best learning rate, max depth, number of estimators, and subsample. Once the best hyperparameters were found – learning rate: 0.1, max depth: 4, n estimators : 100, sub sample: 0.5 –, the models were trained in three different ways like the other models.

CHAPTER 5

Results

5.1 KNN

The first-round model was trained on all the data. It was tested in three different ways:

1. Trained on the full dataset once and tested with the BH and WA subsets,
2. Trained on the full dataset 30 times using shuffle-split and tested with the BH and the WA subset data 30 times using shuffle-split, and
3. Trained and tested separately on the WA and BH subsets, each 30 times.

The metrics did not differ substantially in terms of the accuracy and the F1 score.

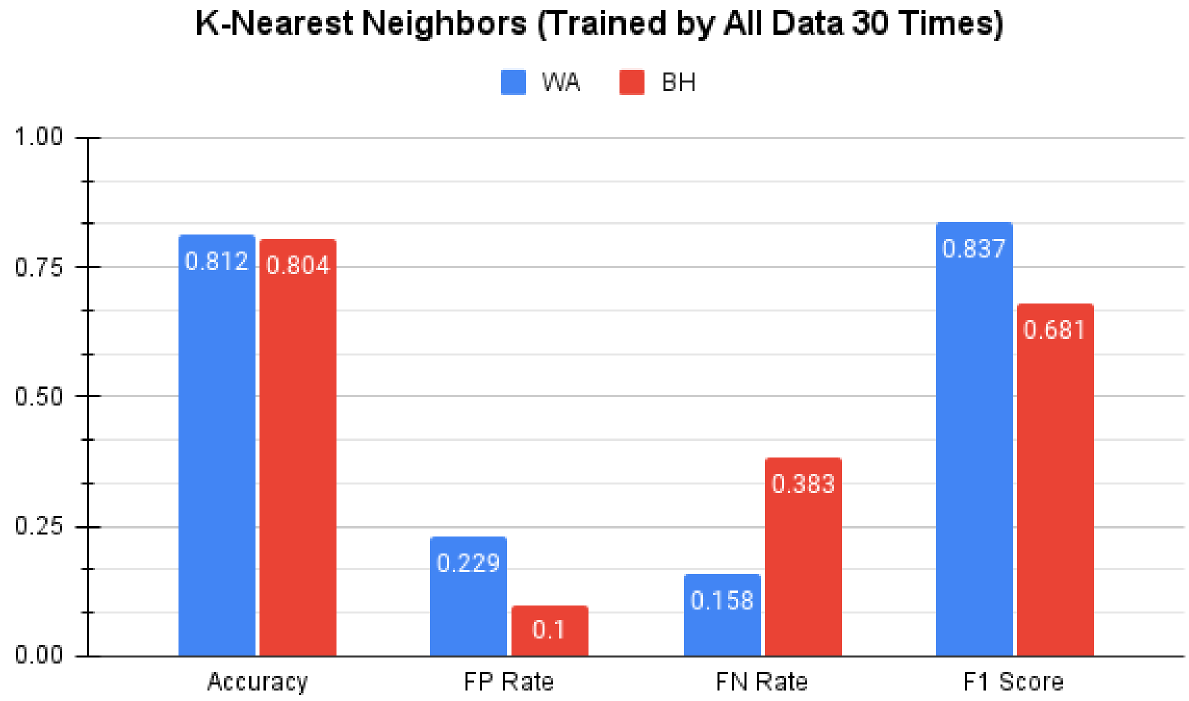
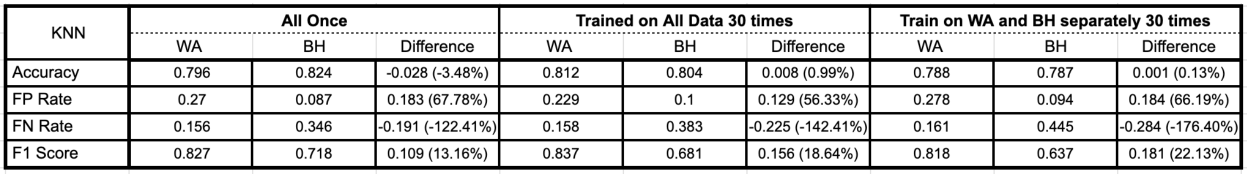


Figure 5.1: Results for KNN trained by all data 30 times

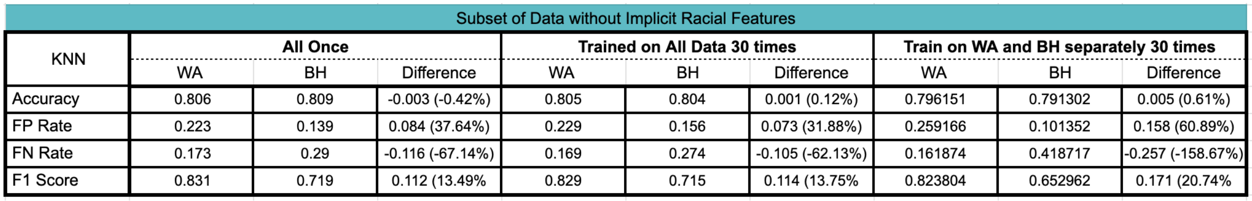
As seen in the bar chart above, although the accuracy for both WA and BH may appear almost the same, the false positive and false negative rates differ by more than double.



KNN	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.796	0.824	-0.028 (-3.48%)	0.812	0.804	0.008 (0.99%)	0.788	0.787	0.001 (0.13%)
FP Rate	0.27	0.087	0.183 (67.78%)	0.229	0.1	0.129 (56.33%)	0.278	0.094	0.184 (66.19%)
FN Rate	0.156	0.346	-0.191 (-122.41%)	0.158	0.383	-0.225 (-142.41%)	0.161	0.445	-0.284 (-176.40%)
F1 Score	0.827	0.718	0.109 (13.16%)	0.837	0.681	0.156 (18.64%)	0.818	0.637	0.181 (22.13%)

Table 5.1: Metrics Comparison for KNN

Similarly, during the second round the model was performed by the subset of data in three different ways. The hyperparameter was tuned the same way. Both by the accuracy of the training and testing data as well as the error rate. The best number of neighbors was 23.

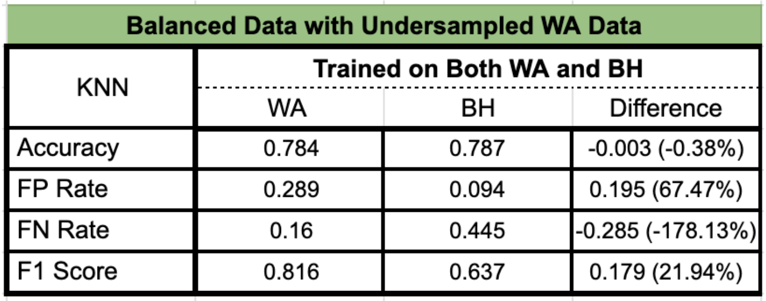


Subset of Data without Implicit Racial Features									
KNN	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.806	0.809	-0.003 (-0.42%)	0.805	0.804	0.001 (0.12%)	0.796151	0.791302	0.005 (0.61%)
FP Rate	0.223	0.139	0.084 (37.64%)	0.229	0.156	0.073 (31.88%)	0.259166	0.101352	0.158 (60.89%)
FN Rate	0.173	0.29	-0.116 (-67.14%)	0.169	0.274	-0.105 (-62.13%)	0.161874	0.418717	-0.257 (-158.67%)
F1 Score	0.831	0.719	0.112 (13.49%	0.829	0.715	0.114 (13.75%	0.823804	0.652962	0.171 (20.74%

Table 5.2: Metrics Comparison for KNN that was trained by subset of data without implicit racial features

Accuracy does not seem to capture these significant differences between false negative and false positive rates.

Lastly, the model was trained by the balanced data, which has the same number of the WA and BH students. Interestingly, the difference for each metric between the WA and the BH group seems to have increased.



Balanced Data with Undersampled WA Data			
KNN	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.784	0.787	-0.003 (-0.38%)
FP Rate	0.289	0.094	0.195 (67.47%)
FN Rate	0.16	0.445	-0.285 (-178.13%)
F1 Score	0.816	0.637	0.179 (21.94%)

Table 5.3: Metrics Comparison for KNN that was trained by balanced data

5.2 Logistic Regression

With tuned parameters, the model was trained by all data once, all data 30 times, and by separate WA and BH data 30 times. All three times, there is not much difference in accuracy- all three differences for accuracy are below 2%. However, the F1 score has a significantly bigger difference for all three models. Based on the higher false positive rates, one can assume logistic regression consistently predicts WA students to be performing higher than they actually are while underestimating BH group's performance than they actually are. WA's false positive rates are higher for all three times and BH's false negative rates are higher all three times.

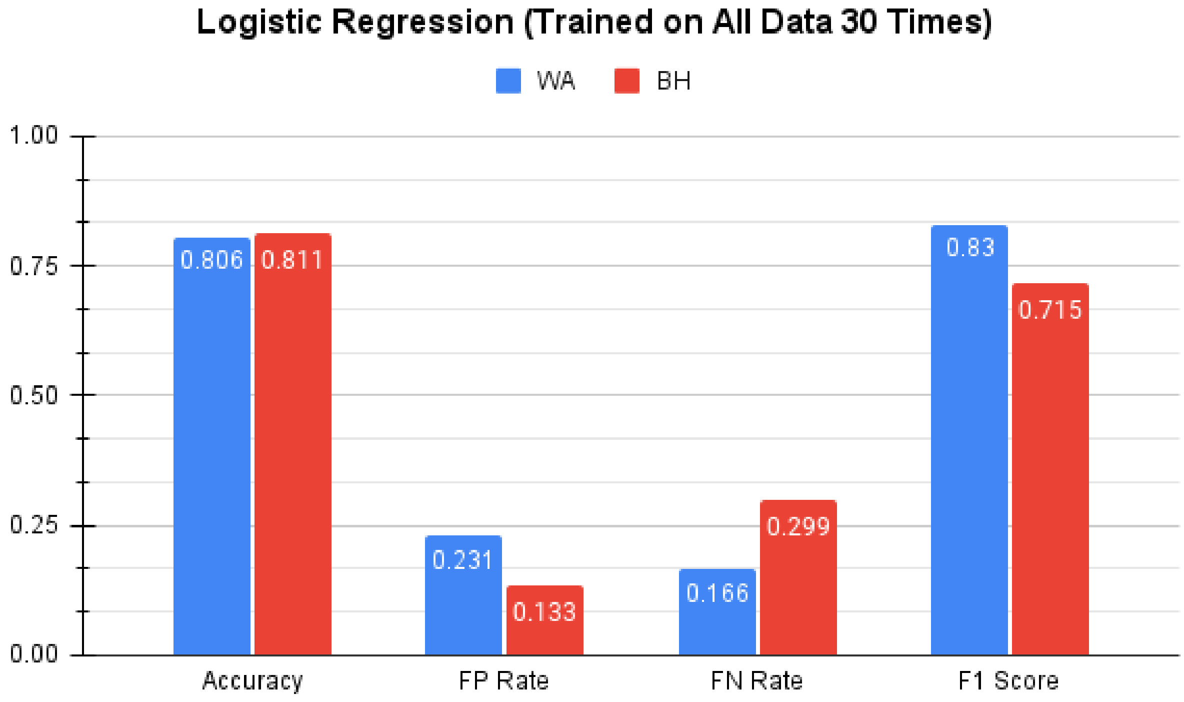
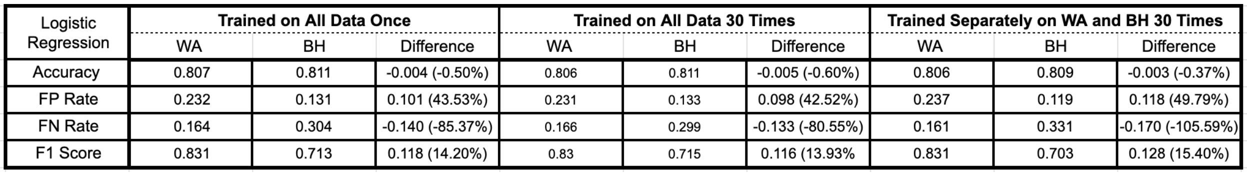


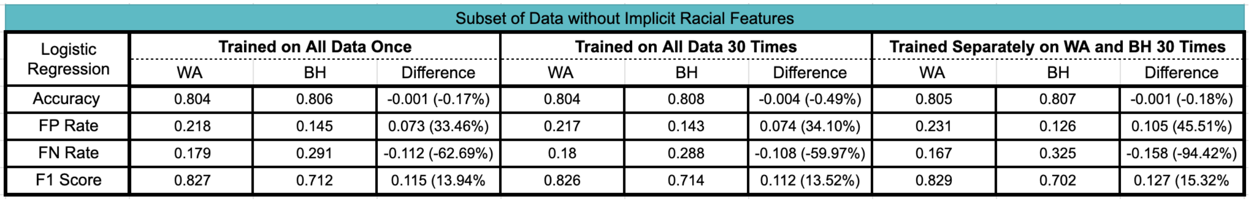
Figure 5.2: Results for logistic regression trained by all data 30 times



Logistic Regression	Trained on All Data Once			Trained on All Data 30 Times			Trained Separately on WA and BH 30 Times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.807	0.811	-0.004 (-0.50%)	0.806	0.811	-0.005 (-0.60%)	0.806	0.809	-0.003 (-0.37%)
FP Rate	0.232	0.131	0.101 (43.53%)	0.231	0.133	0.098 (42.52%)	0.237	0.119	0.118 (49.79%)
FN Rate	0.164	0.304	-0.140 (-85.37%)	0.166	0.299	-0.133 (-80.55%)	0.161	0.331	-0.170 (-105.59%)
F1 Score	0.831	0.713	0.118 (14.20%)	0.83	0.715	0.116 (13.93%	0.831	0.703	0.128 (15.40%)

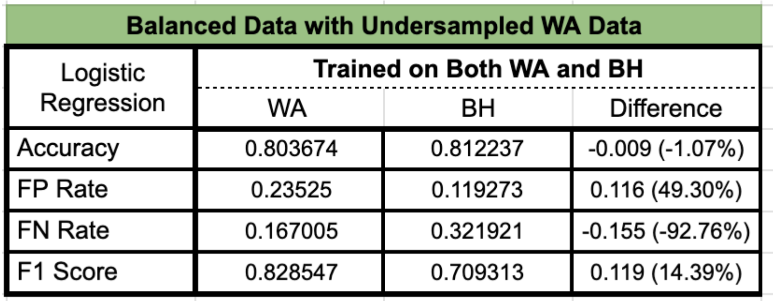
Table 5.4: Metrics Comparison for logistic regression model that was trained by all data

Next, logistic regression was trained by a subset of data without implicit racial features. Between the model that was trained by all data and subset data, there is almost no difference in accuracy, F1 score, False Positive Rate, and False Negative Rates.



Subset of Data without Implicit Racial Features									
Logistic Regression	Trained on All Data Once			Trained on All Data 30 Times			Trained Separately on WA and BH 30 Times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.804	0.806	-0.001 (-0.17%)	0.804	0.808	-0.004 (-0.49%)	0.805	0.807	-0.001 (-0.18%)
FP Rate	0.218	0.145	0.073 (33.46%)	0.217	0.143	0.074 (34.10%)	0.231	0.126	0.105 (45.51%)
FN Rate	0.179	0.291	-0.112 (-62.69%)	0.18	0.288	-0.108 (-59.97%)	0.167	0.325	-0.158 (-94.42%)
F1 Score	0.827	0.712	0.115 (13.94%	0.826	0.714	0.112 (13.52%)	0.829	0.702	0.127 (15.32%

Table 5.5: Metrics Comparison for logistic regression model that was trained by subset of data without implicit racial features



Balanced Data with Undersampled WA Data			
Logistic Regression	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.803674	0.812237	-0.009 (-1.07%)
FP Rate	0.23525	0.119273	0.116 (49.30%)
FN Rate	0.167005	0.321921	-0.155 (-92.76%)
F1 Score	0.828547	0.709313	0.119 (14.39%)

Table 5.6: Metrics Comparison for logistic regression model that was trained by balanced data

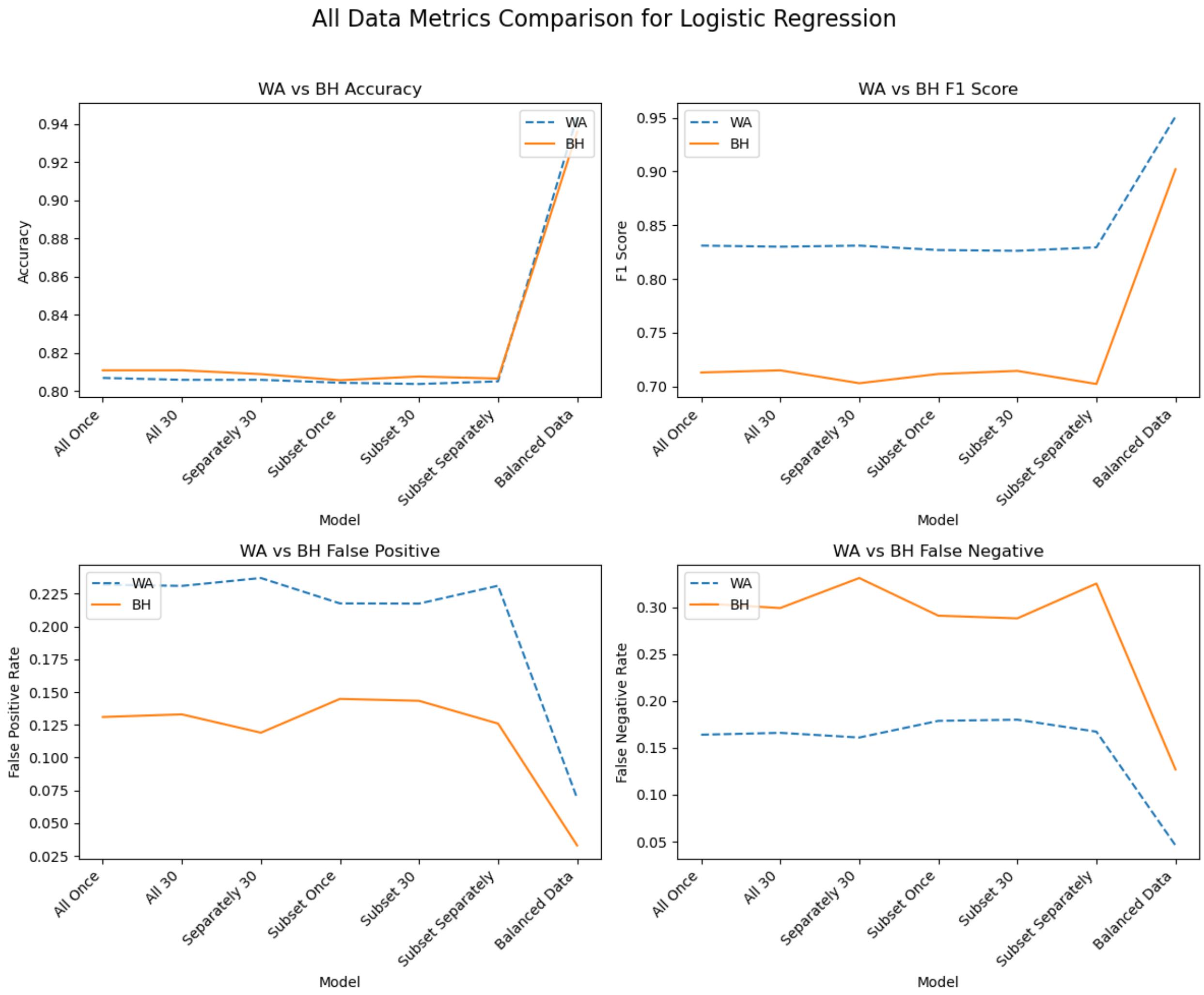
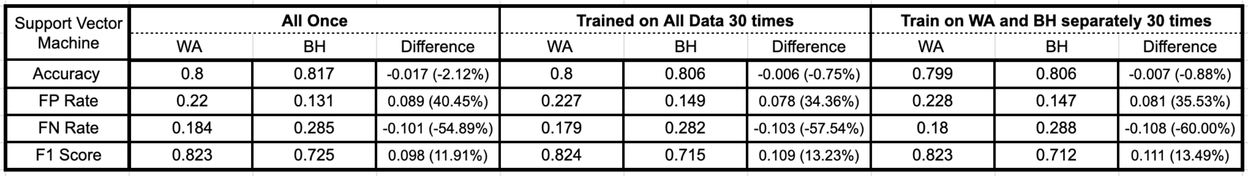


Figure 5.3: Logistic Regression Model Results Comparison. The summary of the results of the model depending on how it was trained

For logistic regression, the WA group's accuracy is consistently lower than the accuracy for the BH group, whereas the F1 score is higher for the WA group. Although the accuracies for may seem to differ a lot on the chart, the differences are all under 1% while the F1 scores differ by more than 10% each time.

5.3 Support Vector Machine

The support vector machine was trained by all data once, trained by all data 30 times, and trained 30 times separately by WA and BH data. This process was repeated with the subset of data without implicit racial features.



Support Vector Machine	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.8	0.817	-0.017 (-2.12%)	0.8	0.806	-0.006 (-0.75%)	0.799	0.806	-0.007 (-0.88%)
FP Rate	0.22	0.131	0.089 (40.45%)	0.227	0.149	0.078 (34.36%)	0.228	0.147	0.081 (35.53%)
FN Rate	0.184	0.285	-0.101 (-54.89%)	0.179	0.282	-0.103 (-57.54%)	0.18	0.288	-0.108 (-60.00%)
F1 Score	0.823	0.725	0.098 (11.91%)	0.824	0.715	0.109 (13.23%)	0.823	0.712	0.111 (13.49%)

Table 5.7: Metrics Comparison for SVM model that was trained by all data

Whether the model was trained by the data once or 30 times, the results did not differ greatly. SVM still had a higher false negative rate for the BH group and a higher false positive rate for the WA group.

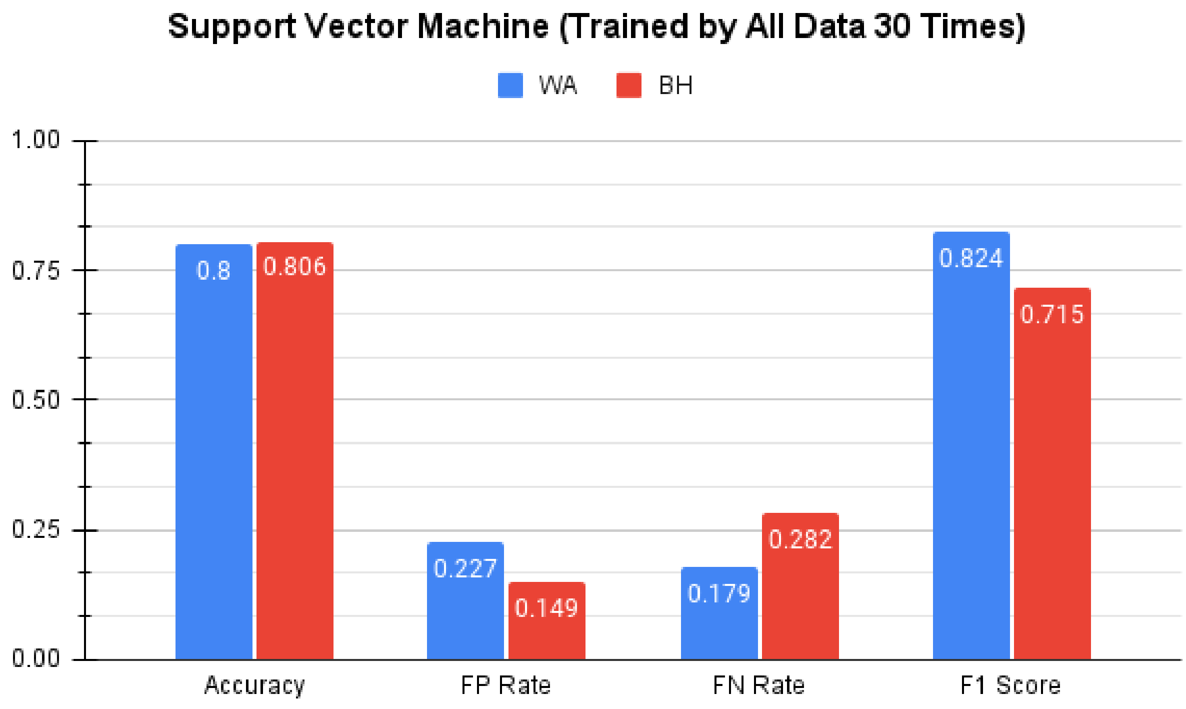
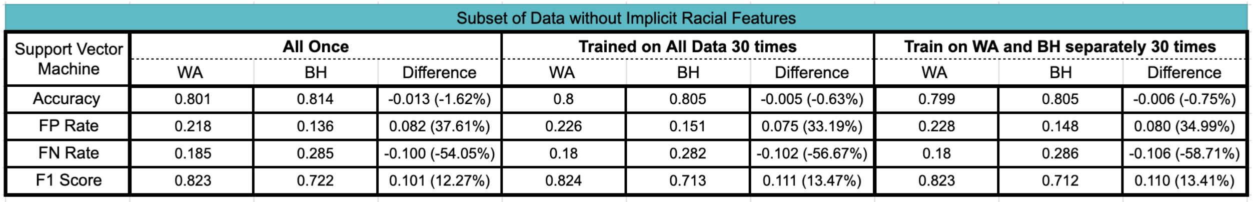
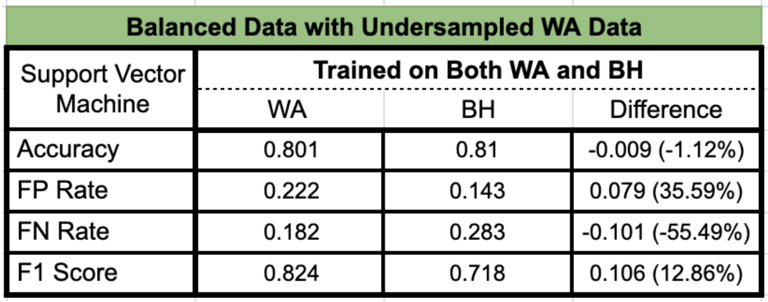


Figure 5.4: Results for support vector machine trained by all data 30 times



Subset of Data without Implicit Racial Features									
Support Vector Machine	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.801	0.814	-0.013 (-1.62%)	0.8	0.805	-0.005 (-0.63%)	0.799	0.805	-0.006 (-0.75%)
FP Rate	0.218	0.136	0.082 (37.61%)	0.226	0.151	0.075 (33.19%)	0.228	0.148	0.080 (34.99%)
FN Rate	0.185	0.285	-0.100 (-54.05%)	0.18	0.282	-0.102 (-56.67%)	0.18	0.286	-0.106 (-58.71%)
F1 Score	0.823	0.722	0.101 (12.27%)	0.824	0.713	0.111 (13.47%)	0.823	0.712	0.110 (13.41%)

Table 5.8: Metrics Comparison for SVM model that was trained by subset of data without implicit racial features



Balanced Data with Undersampled WA Data

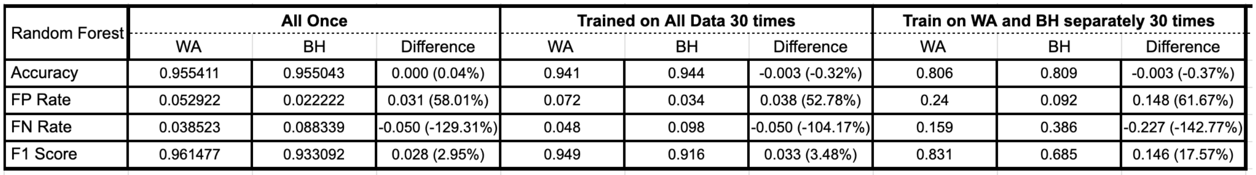
Support Vector Machine	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.801	0.81	-0.009 (-1.12%)
FP Rate	0.222	0.143	0.079 (35.59%)
FN Rate	0.182	0.283	-0.101 (-55.49%)
F1 Score	0.824	0.718	0.106 (12.86%)

Table 5.9: Metrics Comparison for SVM model that was trained by balanced data

5.4 Random Forest

To compare the results, random forest was trained by all data once and tested with the WA and the BH test set. Next, random forest was trained by all data 30 times through shuffle-split and tested 30 times with the WA and the BH test dataset. Lastly, random forest was trained and tested by the WA and the BH data 30 times separately.

Surprisingly, the model that was trained only once had a higher accuracy and a barely any difference in accuracy. It had a lower difference in F1 score compared to the model that was trained 30 times.



Random Forest	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.955411	0.955043	0.000 (0.04%)	0.941	0.944	-0.003 (-0.32%)	0.806	0.809	-0.003 (-0.37%)
FP Rate	0.052922	0.022222	0.031 (58.01%)	0.072	0.034	0.038 (52.78%)	0.24	0.092	0.148 (61.67%)
FN Rate	0.038523	0.088339	-0.050 (-129.31%)	0.048	0.098	-0.050 (-104.17%)	0.159	0.386	-0.227 (-142.77%)
F1 Score	0.961477	0.933092	0.028 (2.95%)	0.949	0.916	0.033 (3.48%)	0.831	0.685	0.146 (17.57%)

Table 5.10: Metrics Comparison for random forest model that was trained by all data

For the model that was trained by all subset data 30 times has achieved a higher accuracy. However, the difference between the false positive and false negative rate was almost double the difference from the model that was trained by the subset data only once. When the model was trained by subset of data separately by the WA and the BH group had a higher difference in both false positive and false negative rates. This was captured by the F1 score between the WA and the BH group unlike the accuracy.

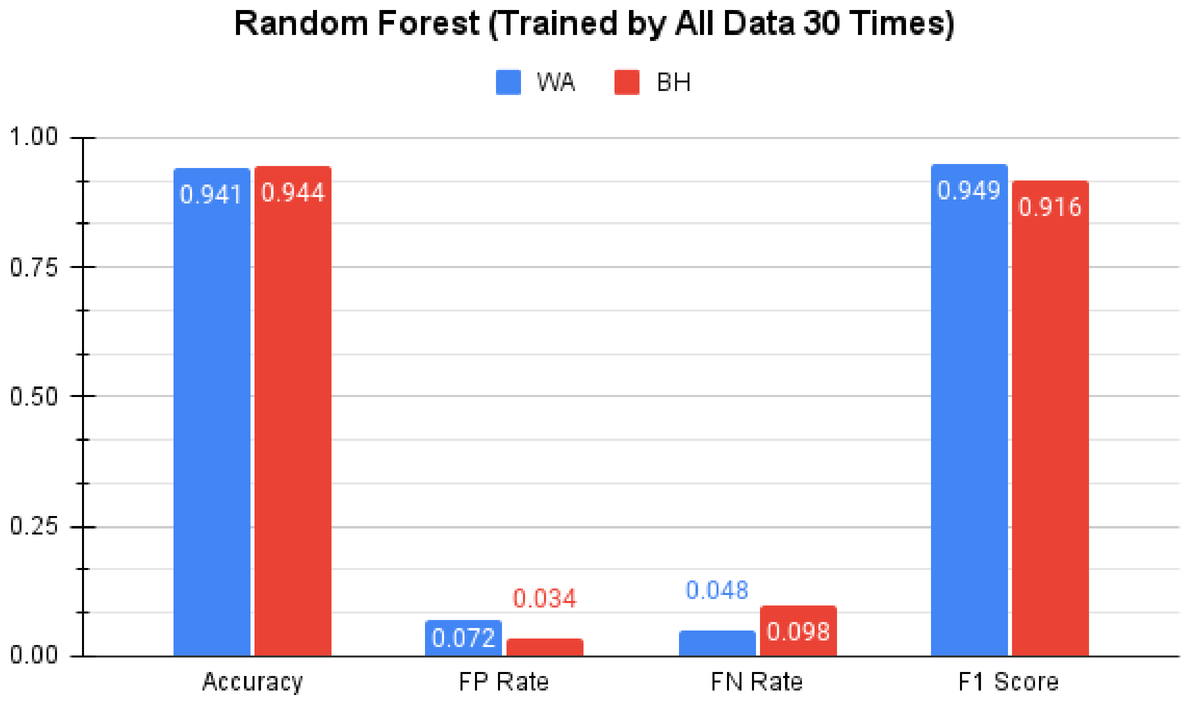
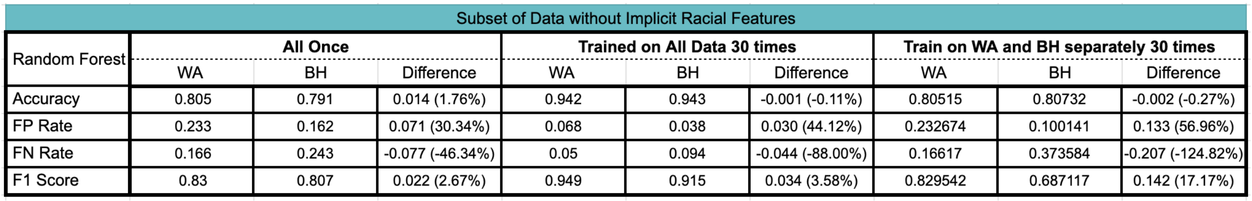
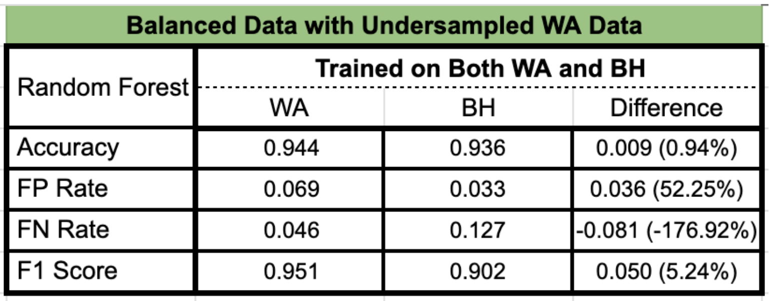


Figure 5.5: Results for random forest model that was trained by all data 30 times



Subset of Data without Implicit Racial Features									
Random Forest	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.805	0.791	0.014 (1.76%)	0.942	0.943	-0.001 (-0.11%)	0.80515	0.80732	-0.002 (-0.27%)
FP Rate	0.233	0.162	0.071 (30.34%)	0.068	0.038	0.030 (44.12%)	0.232674	0.100141	0.133 (56.96%)
FN Rate	0.166	0.243	-0.077 (-46.34%)	0.05	0.094	-0.044 (-88.00%)	0.16617	0.373584	-0.207 (-124.82%)
F1 Score	0.83	0.807	0.022 (2.67%)	0.949	0.915	0.034 (3.58%)	0.829542	0.687117	0.142 (17.17%)

Table 5.11: Metrics Comparison for random forest model that was trained by subset of data without implicit racial features



Balanced Data with Undersampled WA Data			
Random Forest	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.944	0.936	0.009 (0.94%)
FP Rate	0.069	0.033	0.036 (52.25%)
FN Rate	0.046	0.127	-0.081 (-176.92%)
F1 Score	0.951	0.902	0.050 (5.24%)

Table 5.12: Metrics Comparison for random forest model that was trained by balanced data

An interesting observation is that for the model that was trained once by the subset data, most of the metrics were similar except for the WA group. The accuracy and the F1 score significantly dropped while most of the other metrics for the model that was trained 30 times remained the same.

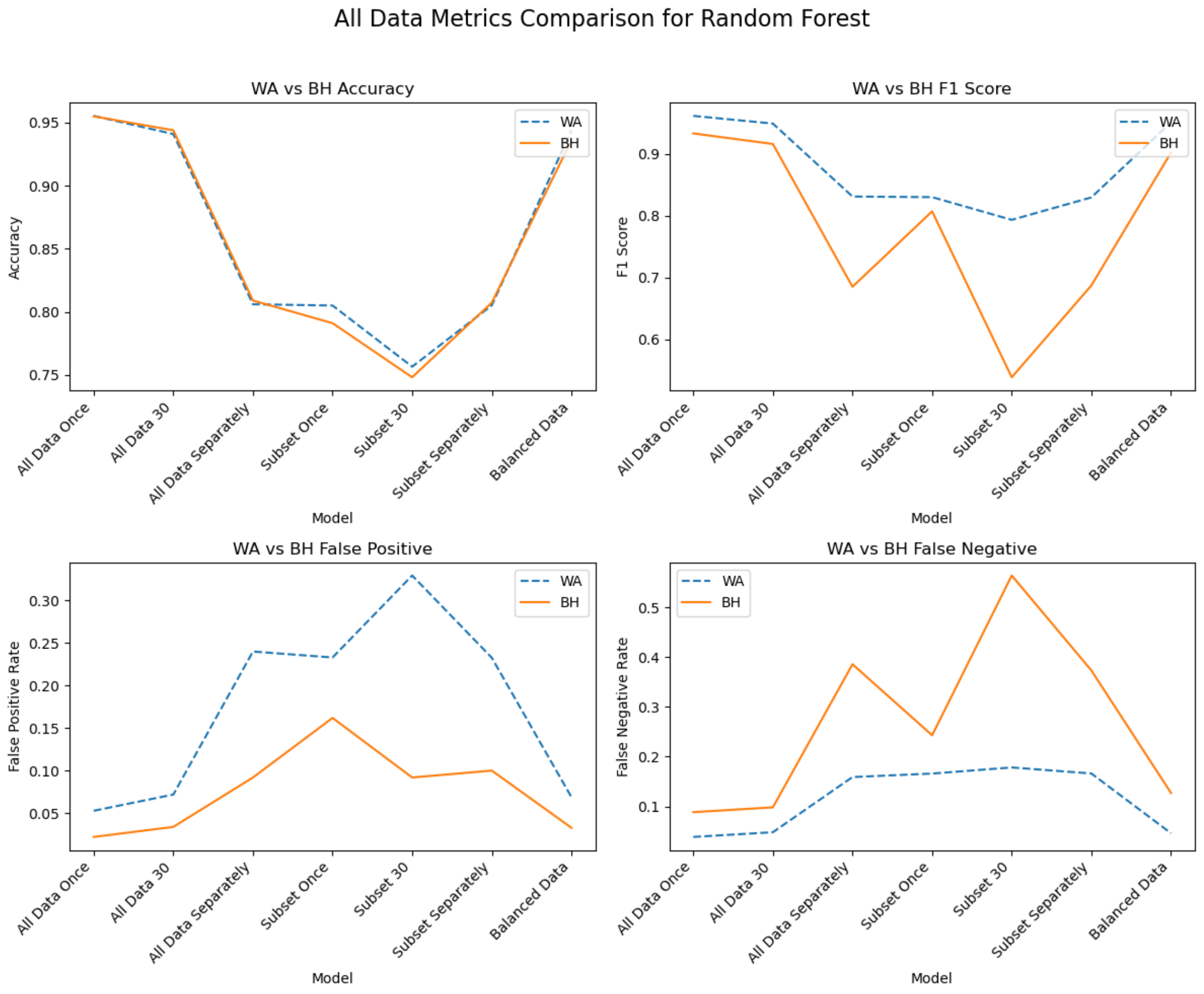


Figure 5.6: Random Forest Model Results Comparison. The summary of the results of the model depending how the model was trained.

As seen in Figure 5.5, the accuracy for both the WA and the BH group seems to be almost the same no matter how the model was trained. However, the F1 score does differ depending which dataset was used. For instance, when the model was trained by the subset data 30 times, it has the highest difference while the difference in F1 score is very small when the data is trained by all data either once or 30 times as well as the balanced data. We can infer how the subset data without implicit racial features do not help much in terms of improving the racial bias in these machine learning algorithms.

Random forest has a few interesting observations. While the accuracy for the WA and the BH group is consistently about the same, the accuracy decreases as it gets trained more by data. The WA's false negative rates do not really change throughout different models. The model has higher differences in terms of false positive and false negative rates when it gets trained by a subset of data without implicit racial features.

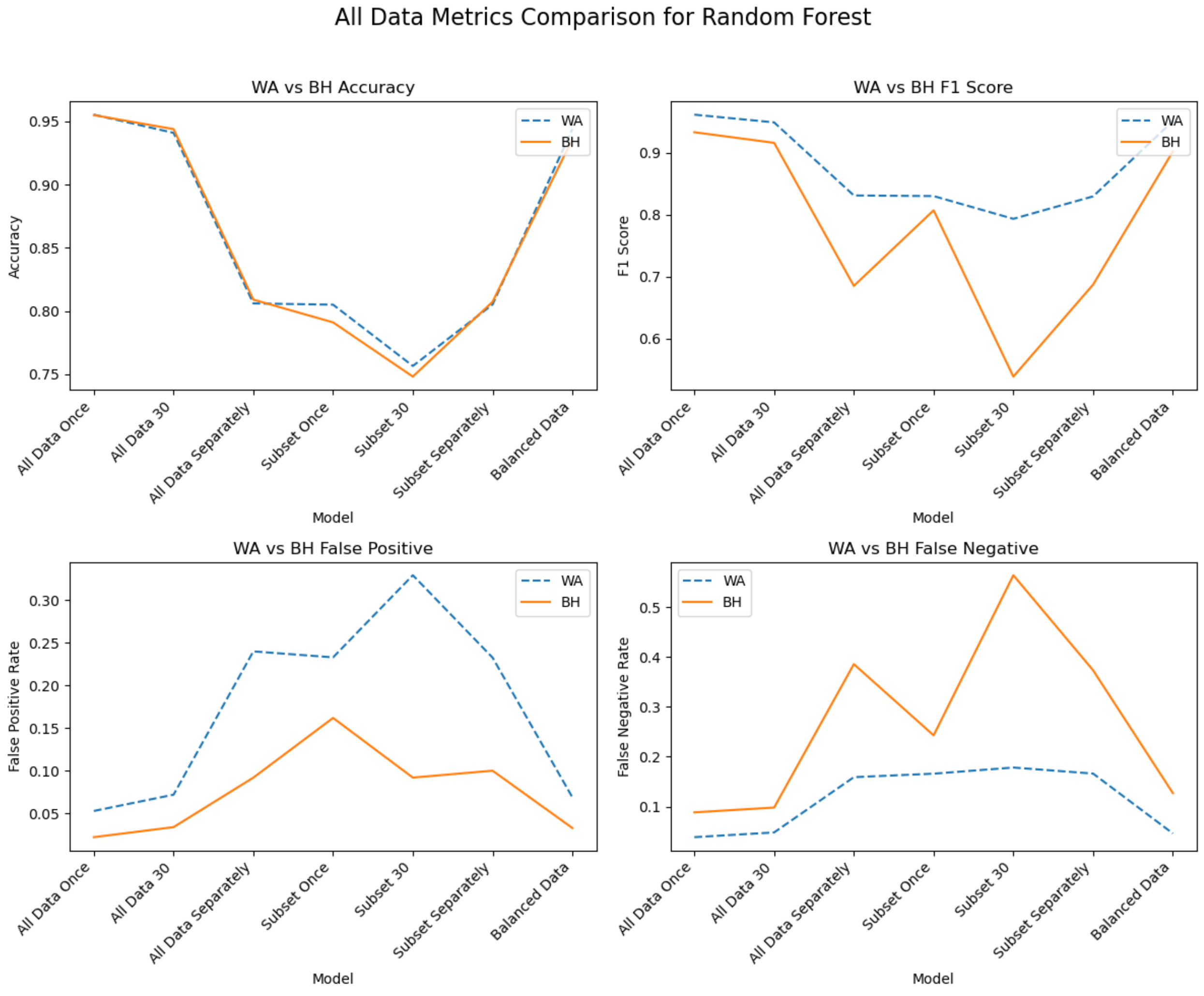


Figure 5.7: Random Forest Model Results Comparison. The summary of the results of the model depending how the model was trained.

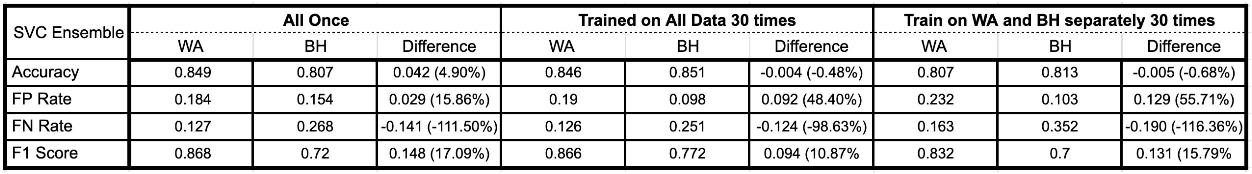
5.5 Ensemble Method

For ensemble method first round, the model was trained by

1) full data once and tested by the BH and the WA subset data,

2) Trained on the full dataset 30 times using shuffle-split and tested 30 times by the BH and the WA subset. Based on these 30 results, the average of accuracy, FP rate, FN rate, and F1 score was calculated.

3) Trained separately on the WA and BH subsets and tested by the WA and BH subset data, each 30 times.



SVC Ensemble	All Once			Trained on All Data 30 times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.849	0.807	0.042 (4.90%)	0.846	0.851	-0.004 (-0.48%)	0.807	0.813	-0.005 (-0.68%)
FP Rate	0.184	0.154	0.029 (15.86%)	0.19	0.098	0.092 (48.40%)	0.232	0.103	0.129 (55.71%)
FN Rate	0.127	0.268	-0.141 (-111.50%)	0.126	0.251	-0.124 (-98.63%)	0.163	0.352	-0.190 (-116.36%)
F1 Score	0.868	0.72	0.148 (17.09%)	0.866	0.772	0.094 (10.87%	0.832	0.7	0.131 (15.79%

Table 5.13: Metrics Comparison for ensemble method that was trained by all data

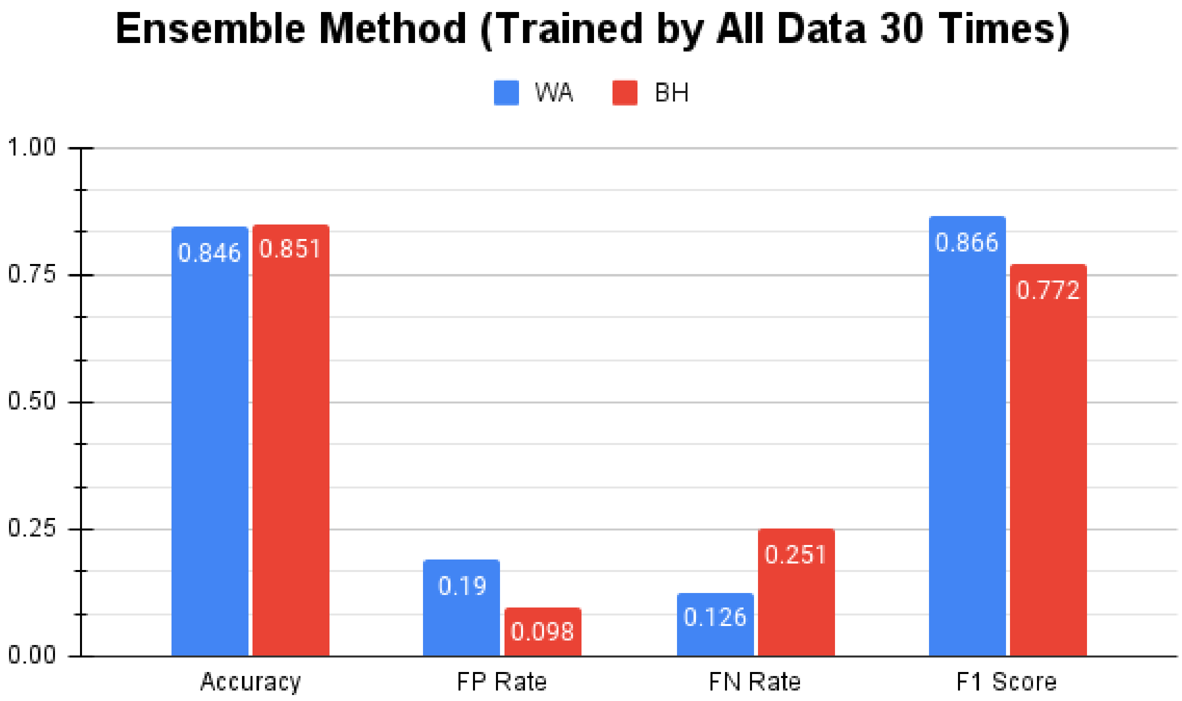
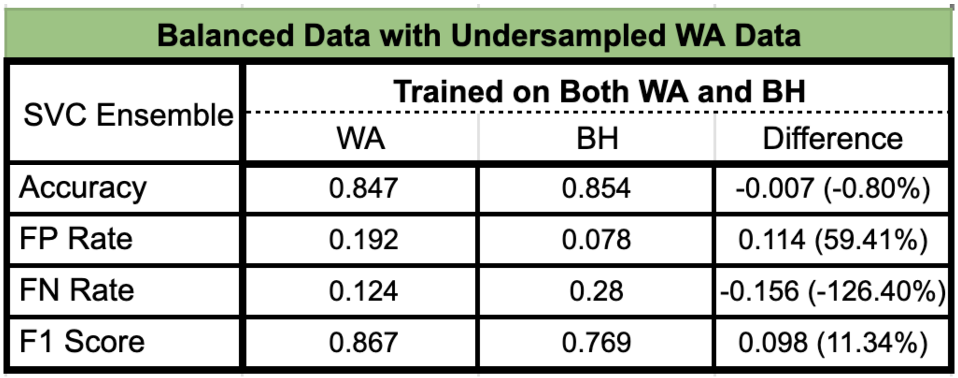


Figure 5.8: Results for ensemble method that was trained by all data 30 times



Balanced Data with Undersampled WA Data

SVC Ensemble	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.847	0.854	-0.007 (-0.80%)
FP Rate	0.192	0.078	0.114 (59.41%)
FN Rate	0.124	0.28	-0.156 (-126.40%)
F1 Score	0.867	0.769	0.098 (11.34%)

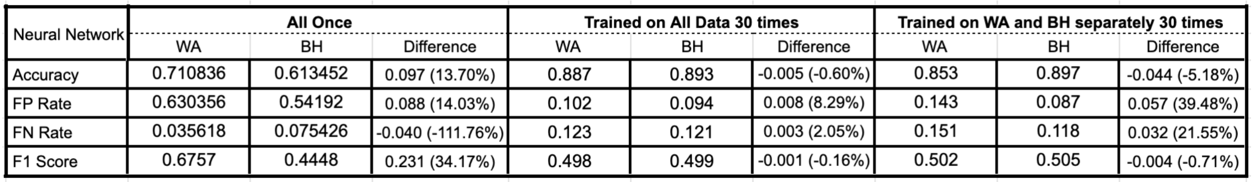
Table 5.14: Metrics Comparison for ensemble method that was trained by balanced data

5.6 Neural Network

Model was trained by all data once and tested on WA data and BH data separately. Second, the model was trained by all data 30 times with shuffle split, where it created 30 training and testing data sets. With these 30 training and testing data, the model improved its performance over time; the more iterations the model went over, the accuracy increased. Thirdly, in a similar manner, the model was trained and tested by WA data separately 30 times and the model was trained and tested by BH data only. And the metrics were appended and compared.

At first, the results of the neural network were mediocre despite its fame for being more of an advanced model. To improve the model's performance, tuning the hyperparameter as well as modifying the network architecture such as increasing more hidden layers.

Below is a result for the neural network model when it was trained with one hidden layer and tuned hyperparameters.

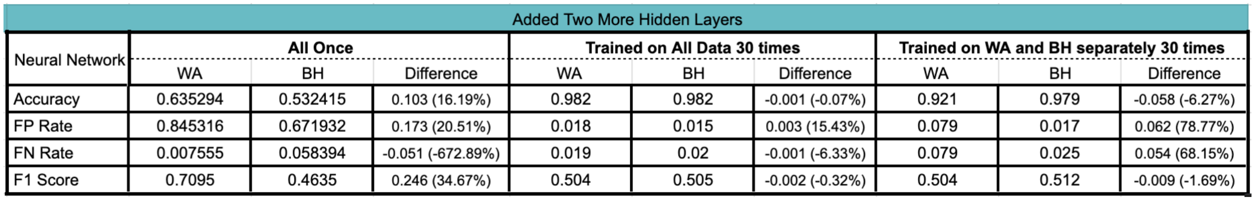


Neural Network	All Once			Trained on All Data 30 times			Trained on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.710836	0.613452	0.097 (13.70%)	0.887	0.893	-0.005 (-0.60%)	0.853	0.897	-0.044 (-5.18%)
FP Rate	0.630356	0.54192	0.088 (14.03%)	0.102	0.094	0.008 (8.29%)	0.143	0.087	0.057 (39.48%)
FN Rate	0.035618	0.075426	-0.040 (-111.76%)	0.123	0.121	0.003 (2.05%)	0.151	0.118	0.032 (21.55%)
F1 Score	0.6757	0.4448	0.231 (34.17%)	0.498	0.499	-0.001 (-0.16%)	0.502	0.505	-0.004 (-0.71%)

Table 5.15: Metrics Comparison for neural network that was trained by all data with just one hidden layer

With the tuned hyperparameters, the neural network model was done three times with all data. First, it was trained by all data just once and predicted the results with the WA and BH data separately. The results for WA and BH were appended and it had the highest difference for False Negative Rates as well as the highest F1 score and accuracy. When the model was trained by all data 30 times and tested with the WA and BH Data separately, it had the lowest differences for all metrics: accuracy, FP rate, FN rate, and F1 score.

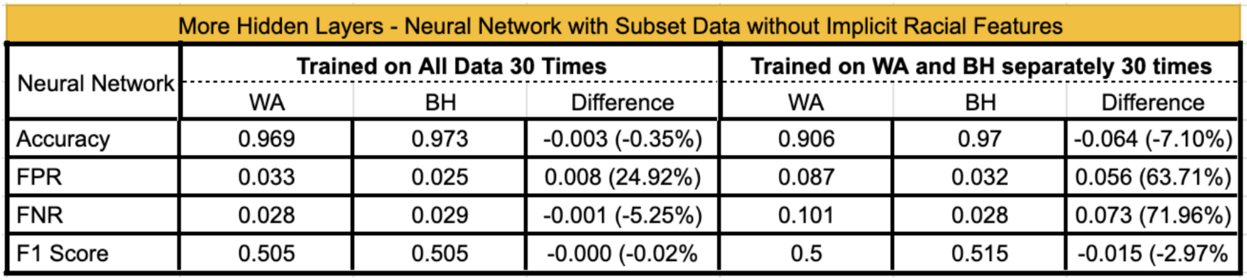
Next, to improve the performance, two more hidden layers with ReLu activation function were added. The rest of the hyperparameters remained the same: the batch size of 64, epoch of 10, and the final output layer with the sigmoid function. Although the model's accuracy improved overall except for the first case where the model was trained by the model only once, the difference between WA and BH for False Positive Rates, False Negative Rates, and F1 ratio rather increased.



Added Two More Hidden Layers									
Neural Network	All Once			Trained on All Data 30 times			Trained on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.635294	0.532415	0.103 (16.19%)	0.982	0.982	-0.001 (-0.07%)	0.921	0.979	-0.058 (-6.27%)
FP Rate	0.845316	0.671932	0.173 (20.51%)	0.018	0.015	0.003 (15.43%)	0.079	0.017	0.062 (78.77%)
FN Rate	0.007555	0.058394	-0.051 (-672.89%)	0.019	0.02	-0.001 (-6.33%)	0.079	0.025	0.054 (68.15%)
F1 Score	0.7095	0.4635	0.246 (34.67%)	0.504	0.505	-0.002 (-0.32%)	0.504	0.512	-0.009 (-1.69%)

Table 5.16: Metrics Comparison for neural network that was trained by all data with just two more hidden layer

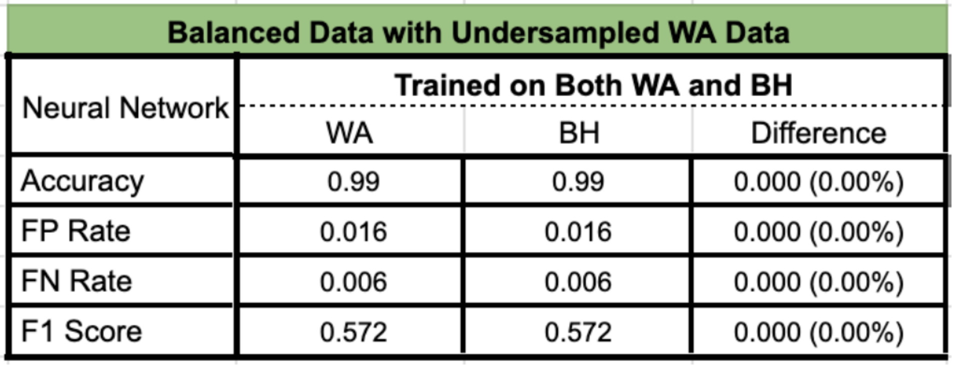
Next, a subset of data without implicit racial features were trained by the tuned neural network model. Due to the consistent low accuracy value when the model was trained only once by all data, the last round was only done twice: trained by all data 30 times and trained by WA and BH data 30 times separately. This resulted in the lowest differences in FPR and F1 ratio, which implies overall the model predicted pretty similarly for both WA and BH groups.



More Hidden Layers - Neural Network with Subset Data without Implicit Racial Features						
Neural Network	Trained on All Data 30 Times			Trained on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference
Accuracy	0.969	0.973	-0.003 (-0.35%)	0.906	0.97	-0.064 (-7.10%)
FPR	0.033	0.025	0.008 (24.92%)	0.087	0.032	0.056 (63.71%)
FNR	0.028	0.029	-0.001 (-5.25%)	0.101	0.028	0.073 (71.96%)
F1 Score	0.505	0.505	-0.000 (-0.02%	0.5	0.515	-0.015 (-2.97%

Table 5.17: Metrics Comparison for neural network that was trained by all data with just two more hidden layer

Lastly, a balanced data that has the same number of WA students and BH students were used to train the neural network model. Neural network has four hidden layers of ReLu function and the last activation function is sigmoid function.



Balanced Data with Undersampled WA Data			
Neural Network	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.99	0.99	0.000 (0.00%)
FP Rate	0.016	0.016	0.000 (0.00%)
FN Rate	0.006	0.006	0.000 (0.00%)
F1 Score	0.572	0.572	0.000 (0.00%)

Table 5.18: Metrics Comparison for neural network that was trained by balanced data 30 times

The result is strikingly good here. For the model that was trained by the balanced data 30 times, there is no difference for all metrics between the WA and the BH group, which is a desired result since this indicates the model made the prediction equally no matter what the race of the student was. The result seems too good to be true, so there will be a separate discussion on this result.

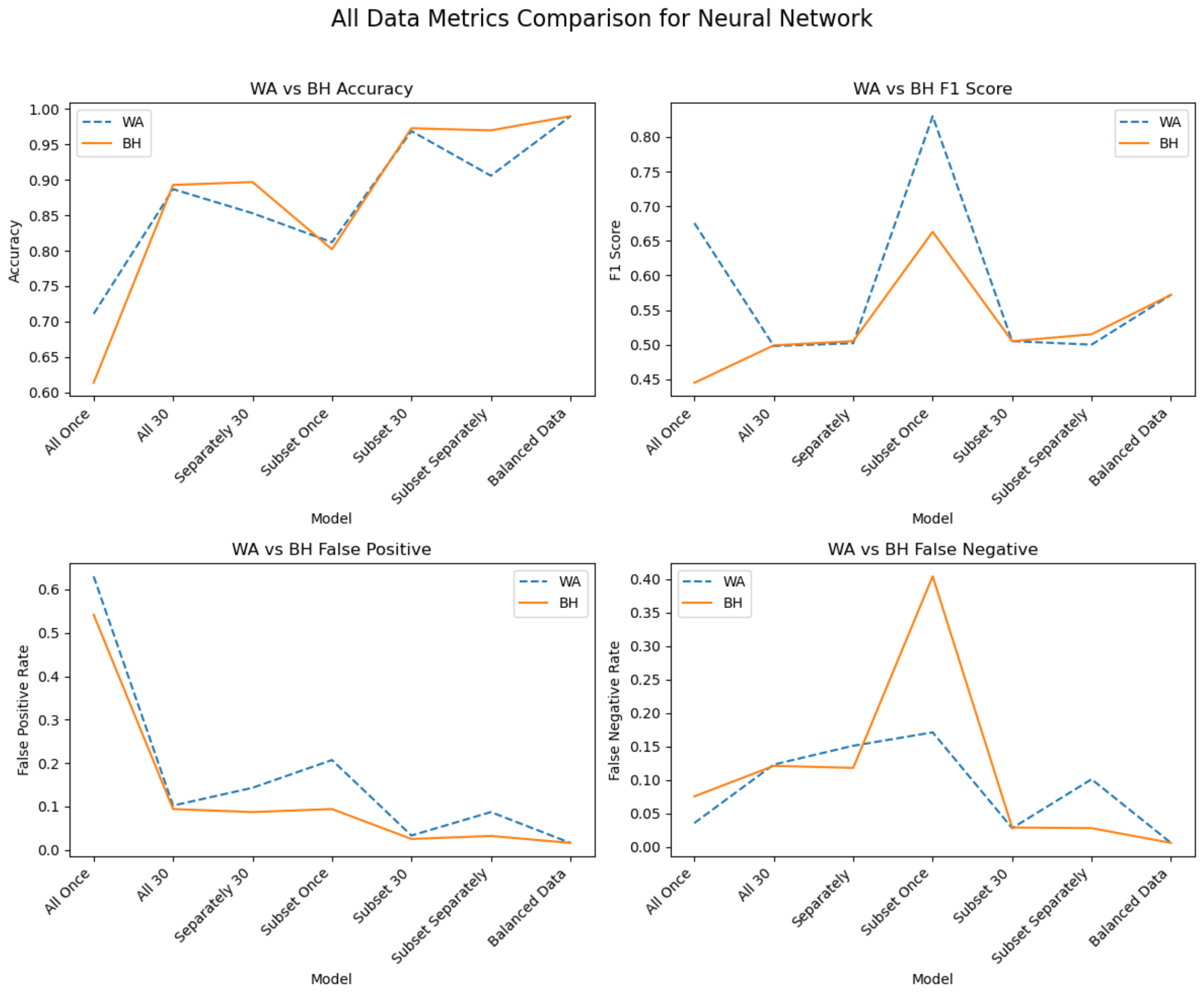


Figure 5.9: Neural Network Model Results Comparison. The summary of the results of the model depending how the model was trained.

For neural networks, the accuracies for the WA and the BH group had a very small difference. The model actually produced higher accuracy when the model was trained by the subset of data 30 times. The false positive rates and the false negative rates for the WA and the BH group are very close to one another except for the data that was trained by the subset data once. For both times that the model was trained by all data 30 times for the whole data and the subset of data without the implicit racial features, the false positive and the false negative rates differed by the smallest difference out of all machine learning algorithms.

Overall, neural network's result was consistent between the WA and the BH group as long as the model was trained by the data 30 times. For example, the F1 scores, false positive rates, and false negative rates differ the most for the model that was trained by all data once and subset data once.

5.7 XGBoost

The model here was trained by

1) full data once and tested by the BH and the WA subset data,

2) the full dataset 30 times using shuffle-split and tested 30 times by the BH and the WA subset. Based on these 30 results, the average of accuracy, FP rate, FN rate, and F1 score was calculated.

3) the WA and BH subsets separately and tested by the WA and BH subset data, each 30 times.

There was not any significant difference in terms of accuracy and the F1 score when the model was trained by all data once or all data 30 times. The false positive and the false negatives' differences between the WA and the BH group. In fact, the the accuracy and the F1 scores were higher for the model that was trained by the data only once than the those of the model that was trained by the data 30 times. Additionally, the model that was trained by the WA and BH model separately performed worse in almost all ways. The accuracy and the F1 score decreased whereas the false negative rates difference between the WA and BH increased, signifying that the model predicted more BH group to be performing worse.

Although the hyperparameters were tuned as mentioned in the methodology section, the results of the model with tuned parameters were far worse, so the results of the model with the tuned hyperparameters are skipped in this paper.

Below is a result of the model that was trained by all data.

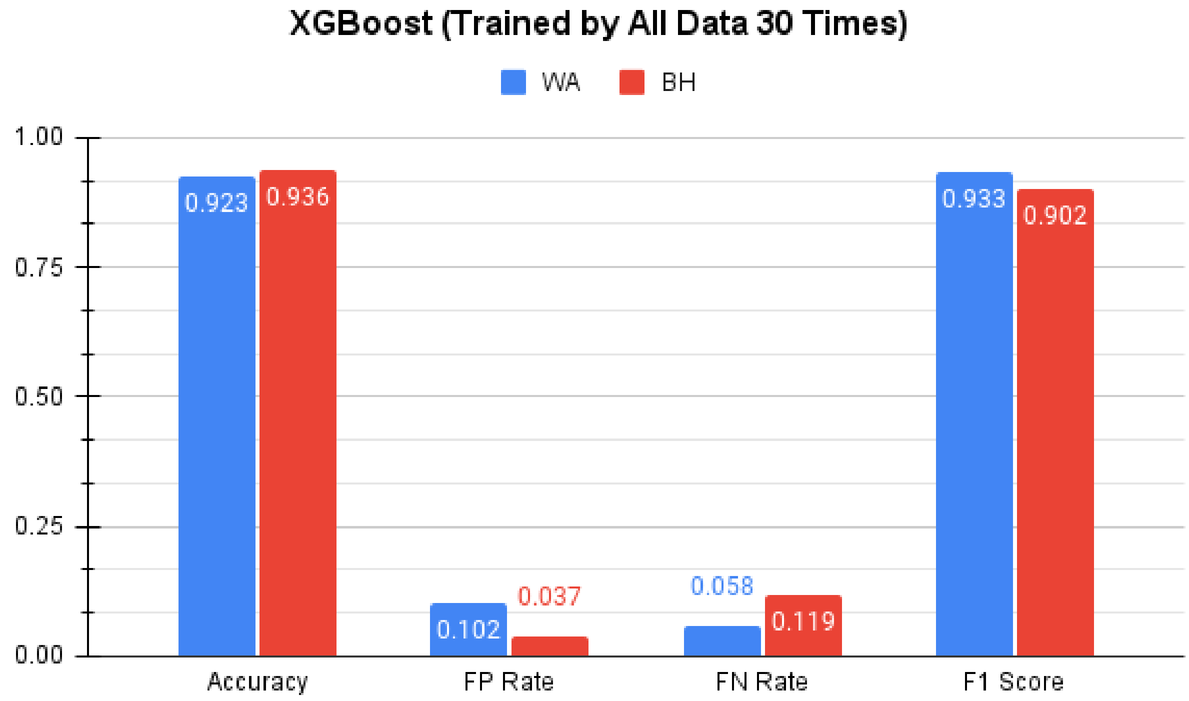
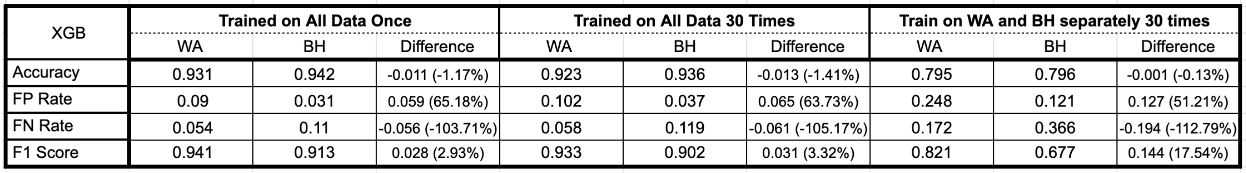


Figure 5.10: XGB results that was trained by all data 30 times

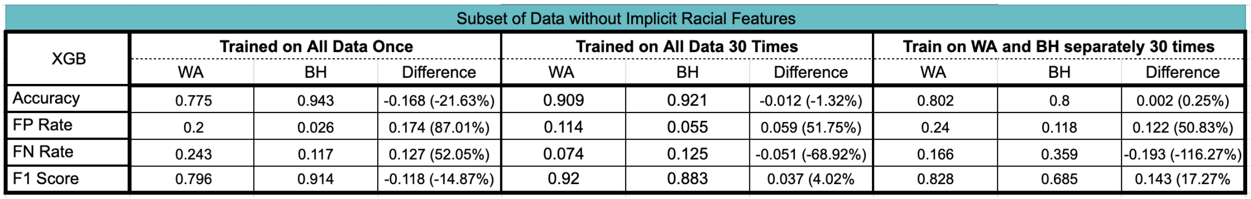


XGB	Trained on All Data Once			Trained on All Data 30 Times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.931	0.942	-0.011 (-1.17%)	0.923	0.936	-0.013 (-1.41%)	0.795	0.796	-0.001 (-0.13%)
FP Rate	0.09	0.031	0.059 (65.18%)	0.102	0.037	0.065 (63.73%)	0.248	0.121	0.127 (51.21%)
FN Rate	0.054	0.11	-0.056 (-103.71%)	0.058	0.119	-0.061 (-105.17%)	0.172	0.366	-0.194 (-112.79%)
F1 Score	0.941	0.913	0.028 (2.93%)	0.933	0.902	0.031 (3.32%)	0.821	0.677	0.144 (17.54%)

Table 5.19: Metrics Comparison for XGB

While the accuracy for both WA and BH groups is very close to another, it fails to capture the difference between false positive and false negative rates. For instance, the false negative rate for the BH group is more than double the false negative rate for the WA group.

Next, the model was trained by a subset of data without implicit racial features.

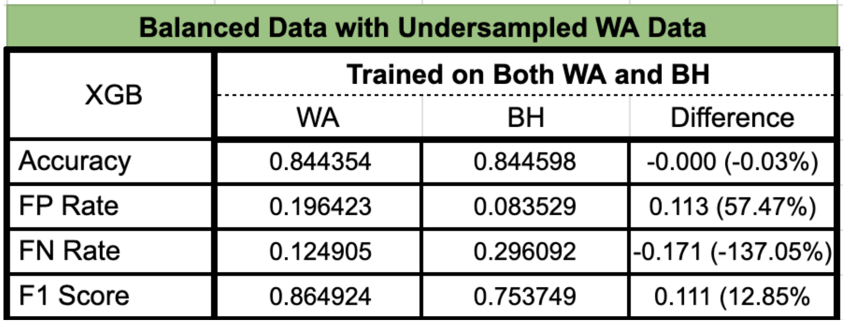


Subset of Data without Implicit Racial Features									
XGB	Trained on All Data Once			Trained on All Data 30 Times			Train on WA and BH separately 30 times		
	WA	BH	Difference	WA	BH	Difference	WA	BH	Difference
Accuracy	0.775	0.943	-0.168 (-21.63%)	0.909	0.921	-0.012 (-1.32%)	0.802	0.8	0.002 (0.25%)
FP Rate	0.2	0.026	0.174 (87.01%)	0.114	0.055	0.059 (51.75%)	0.24	0.118	0.122 (50.83%)
FN Rate	0.243	0.117	0.127 (52.05%)	0.074	0.125	-0.051 (-68.92%)	0.166	0.359	-0.193 (-116.27%)
F1 Score	0.796	0.914	-0.118 (-14.87%)	0.92	0.883	0.037 (4.02%	0.828	0.685	0.143 (17.27%

Table 5.20: Metrics Comparison for XGB with subset of data without implicit racial features

For all three models, there are noticeable differences between the overall accuracy and F1 scores for the model that was trained by the data once, 30 times together, and 30 times separately. The reason why XGBoost models perform worse without implicit racial features needs to be further investigated.

Lastly, a balanced data that has the same number of WA students and BH students were used to train the XGB model.



Balanced Data with Undersampled WA Data			
XGB	Trained on Both WA and BH		
	WA	BH	Difference
Accuracy	0.844354	0.844598	-0.000 (-0.03%)
FP Rate	0.196423	0.083529	0.113 (57.47%)
FN Rate	0.124905	0.296092	-0.171 (-137.05%)
F1 Score	0.864924	0.753749	0.111 (12.85%

Table 5.21: Metrics Comparison for XGB with balanced data

Overall, when all seven models' results are compared, there is one unexplainable detail that needs to be further investigated. The XGB model that was trained by the subset of data that does not have any implicit racial features somehow had a higher false negative rate for the WA group, which means the model predicts more WA students to be underperforming than they actually are. This is a rare result.

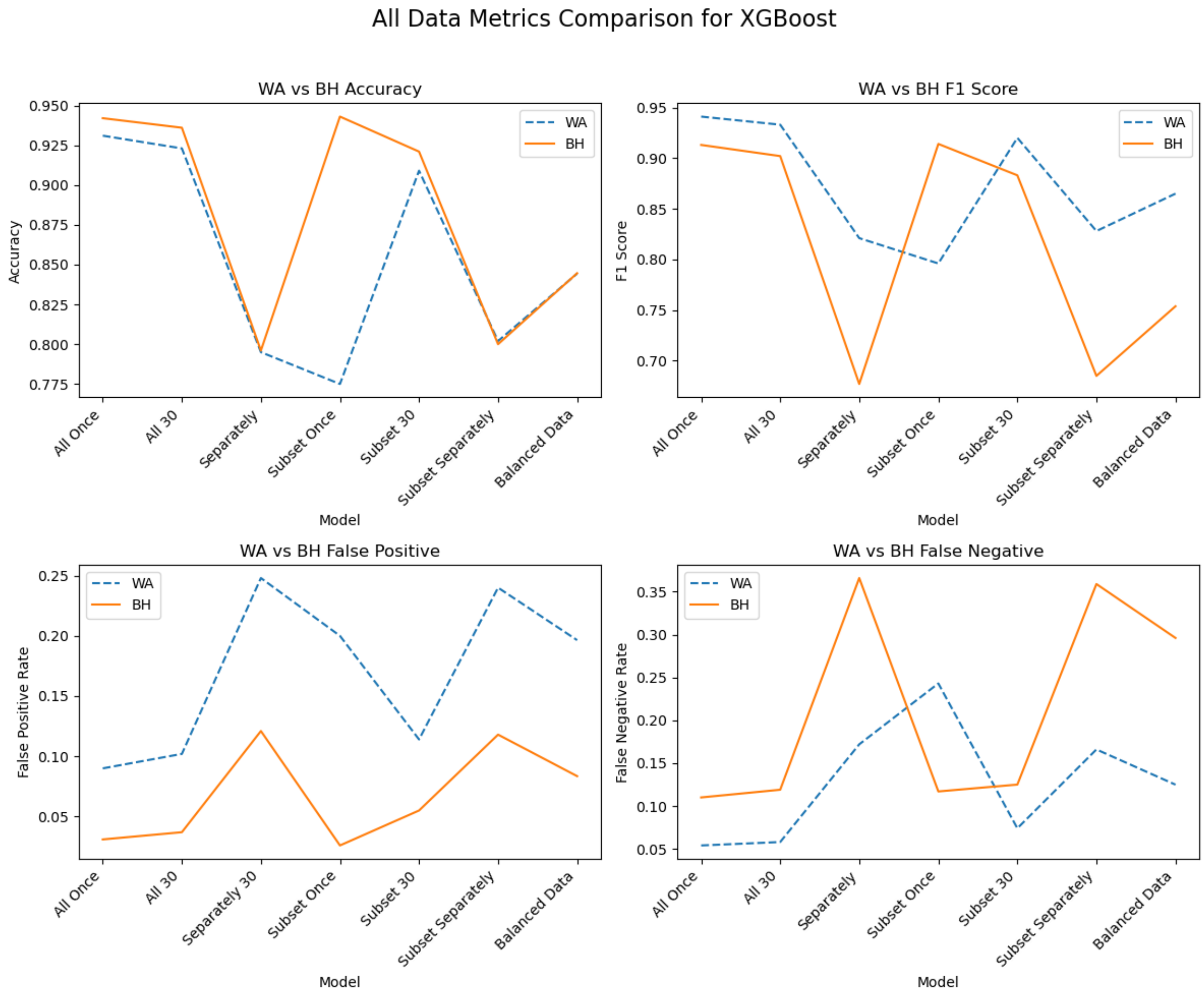


Figure 5.11: XGBModel Results Comparison. The summary of the results of the model depending how the model was trained.

5.8 Overall Results

Seven different machine learning algorithms were trained and tested. Throughout the analysis, 30% of the data was used as a testing dataset and 70% of the data was used as a training dataset. All seven machine learning algorithms give a higher benefit of the doubts to the WA group while pessimistically underestimating the BH group's math performance.

There were three big rounds of training and testing the dataset. The first round is with the entire dataset, the second round is with the subset of data without implicit racial features, and the third round is with the balanced dataset. The models that were trained by the subset of data without implicit racial feature did not have much different result compared to the models that were trained by the entire data.

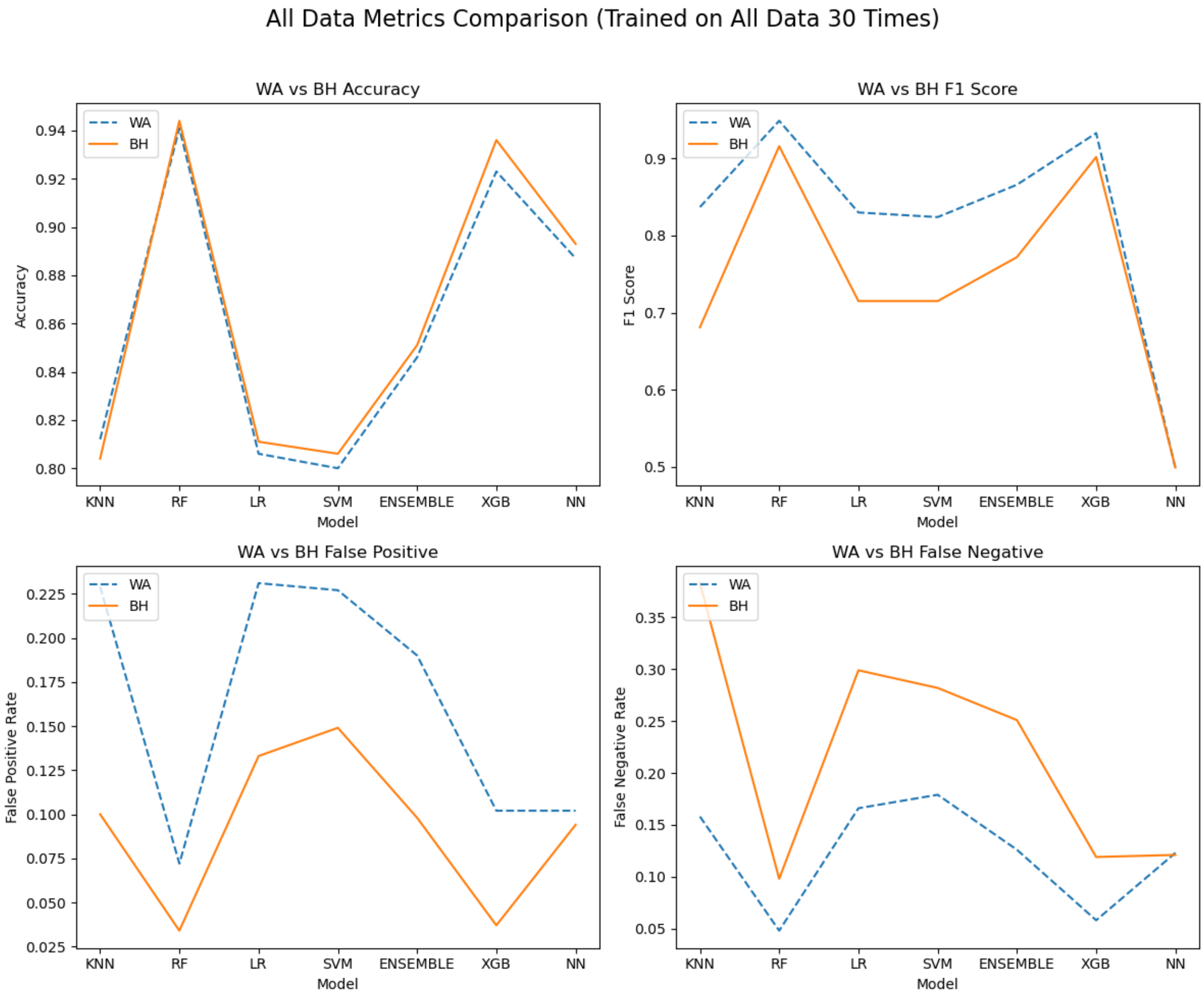


Figure 5.12: Model Results Comparison. The summary of the results of the model that was trained by the WA and the BH data 30 times.

Accuracies for all models were very close to one another. However, F1's difference is more noticeable than the difference of accuracy in each model. Random Forest and Neural Network had the smallest difference in terms of false positive rates and false negative rates, meaning it had the least amount of bias in predicting student's math performance between the WA and the BH group. All seven models have consistently overestimated the WA's math performance while underestimating the BH group's math performance.

Accuracy formula is

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{5.1}$$

and it measures the overall correct number of the model: true positive and true negative out of all the where it does not take false positive and false negative into consideration in the numerator. Generally, accuracy is a good indicator of a model when the data itself is balanced and all errors are equally costly. However, in situations where the false negative and false positives could indicate giving benefit of doubts to one group over another, an accuracy may not be the most accurate measure to look at. F1 score is much better as it is preferred for imbalance data since you need a balance between precision and recall.

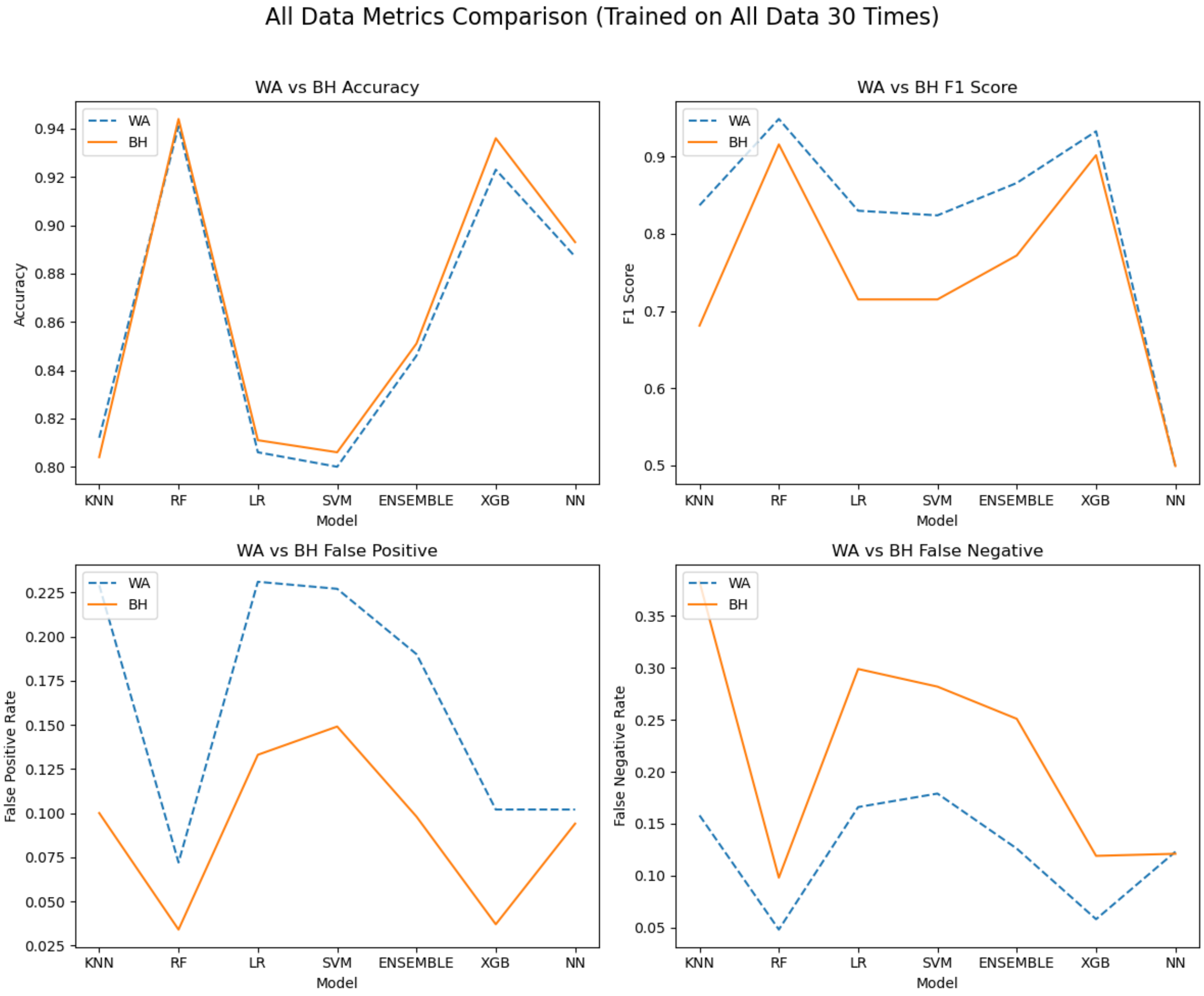


Figure 5.13: Model Results Comparison. The summary of the results of the model that was trained by the WA and the BH data 30 times.

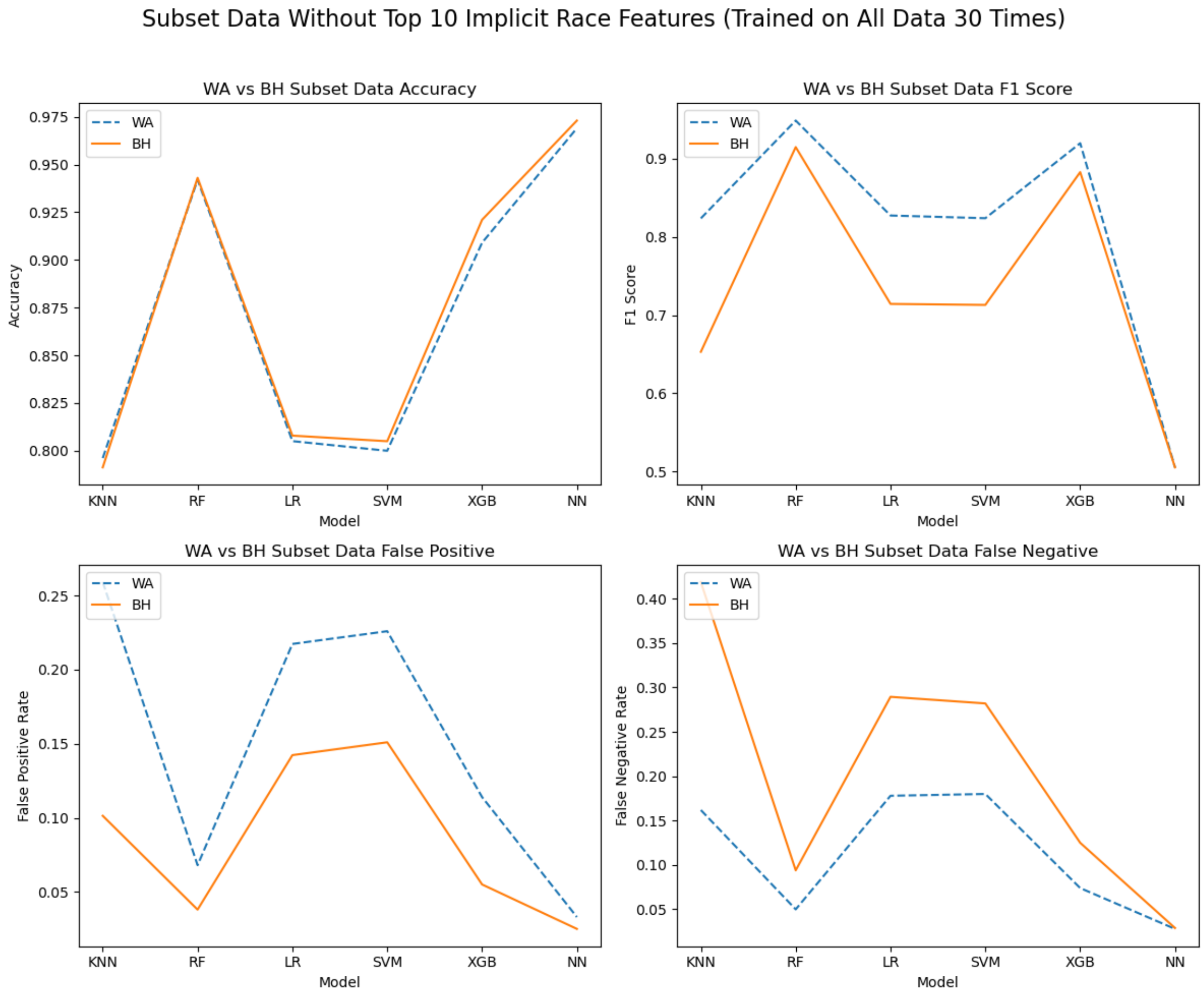


Figure 5.14: Model Results Comparison. The summary of the results of the model that was trained by the subset data.

Due to the imbalance of the input data, another data set is created by under-sampling the WA group. In this balanced data set, the WA is sampled so that the BH group and the WA group has the same composition. Each machine learning algorithm is trained by the balanced data set 30 times and tested on the balanced data set's WA and BH group. Random forest had the second smallest difference between the WA and the BH group in terms of the false positive and false negative rates, which indicates that the model did a pretty good job in terms of predicting the WA's and the BH's group with small bias. One of the most striking results is that the neural network had almost no difference on all the metrics between the WA and BH group, which means it equally predicted the low performing and high performing students regardless of the student's race.

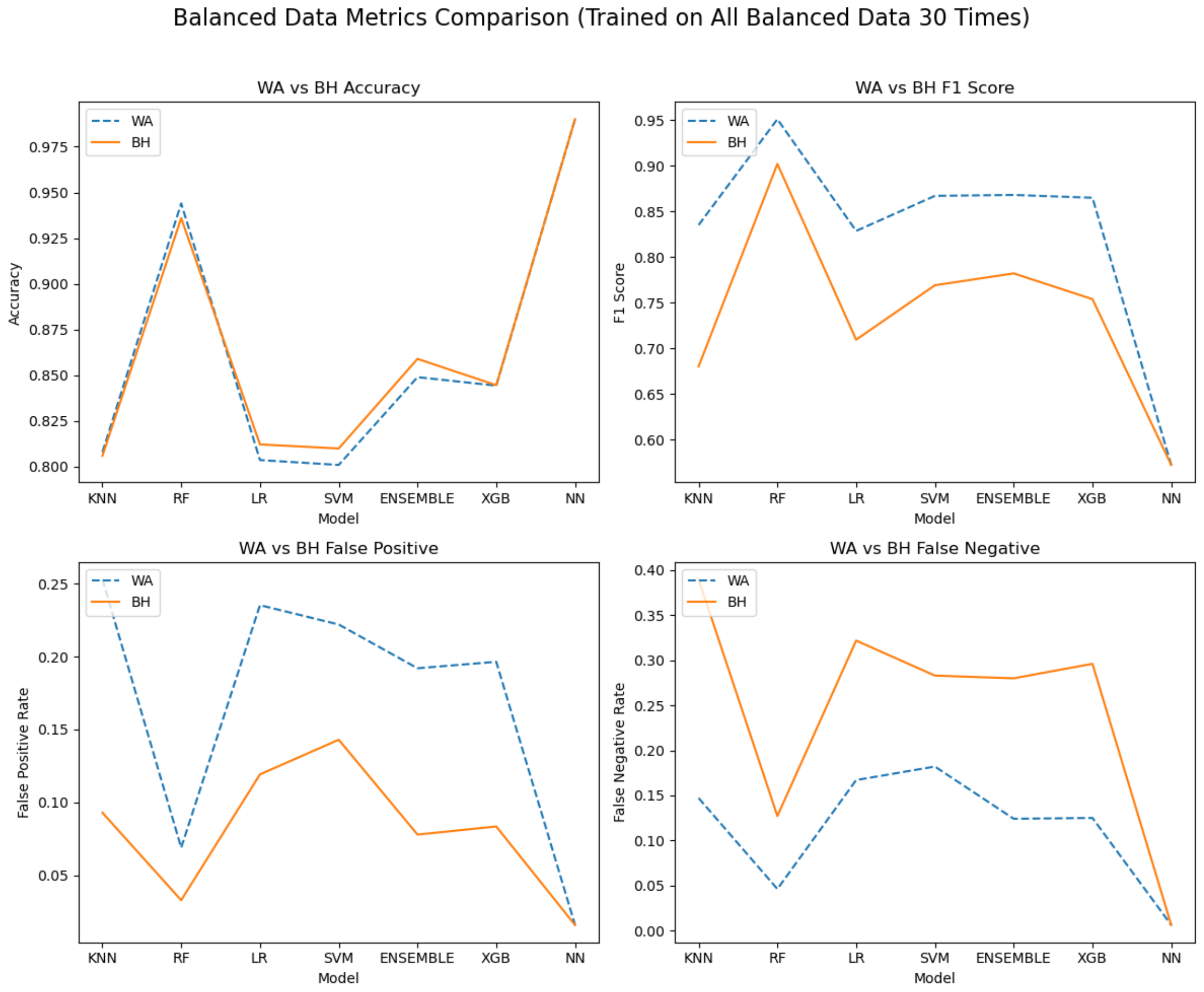


Figure 5.15: Model Results Comparison. The summary of the results of the model that was trained by the balanced data.

CHAPTER 6

Discussion

6.1 Limitations

I, the author of this thesis, originally attempted to get the full dataset from the National Center of Education Statistics (NCES) as I wished to use more variables from the data as there are many potential relationships that could affect students' math scores besides the 59 features that were listed in the data which was used in this paper. However, the access to the full data was denied due to the fact that I was not a researcher or a doctoral student. Due to the lack of full access to the data, I had to reach out to the author of the paper "Who Gets the Benefits of the Doubts?"

6.2 Further Work

There are a few questions that rise from the results. First, what happened to the neural network model results with balanced data? The result contained 0% difference between the WA and the BH group. Why didn't other machine learning algorithms produce this kind of result with the balanced data? Further investigation needs to be done with neural network that was trained by the balanced data.

Secondly, group fairness is defined by equalized odds, which means each student race would have the same positive and false positive rates. Equalized odds are a pretty well-known fairness criterion. It takes the merit different groups of people, such as race, into account by considering the underlying ground truth distribution of the labels, which can ensure the errors across groups are somewhat more 'fair.' [15].

When using the equalized odds, both the false positive rates and the false negative rates need to be the same, which is known as Conditional Procedure Accuracy Equality while also achieving a certain predictive performance with the model. Using the equalized odds with this dataset would be appropriate and potentially produce more insightful results.

Thirdly, using adversarial learning algorithm would be another next step. Adversarial learning has the potential to learn bias-free representations of model input data by removing the bias information about sensitive attributes [16]. In the paper "Towards Equity and Algorithmic Fairness in Student Grade Prediction," the author mentions using adversarial learning achieved the best fairness scores on all metrics of true positive rates, true negative rates, and accuracy [16]. During the model analysis, adversarial learning method was attempted, however the result was omitted due to a poor performance or an error in the codes.

Next, train the model with different racial composition. First round of the model training process was done with the entire dataset. Second round of the model training process was performed without the implicit racial features yet still with all students. The third round of the model training process was done with the balanced data, where it had the same number of the WA and the BH students. However, it is worth looking how the model performs differently with data that is made of different racial composition.

Additionally, using packages such as 'fairlearn' or 'AI Fairness 360' which was developed by developers of artificial intelligence (AI) systems to assess the system's fairness can possibly mitigate any observed unfairness issues.

Lastly, comparison between just the white students and Hispanic/Black students would be another potential future work. 80% of the Asian students are in the top 50 category whereas 70% of the Black students are in the bottom 50. Separating the Asian students from White students could produce other insightful results. Therefore, another step that could be done is compare the false positive rates and false negative rates across different race.

CHAPTER 7

Conclusion

This paper was an extended study on the existing paper ”Who Gets the Benefits of the Doubt?” [7] in the secondary mathematics education settings. The goal was to examine racial bias among seven machine learning algorithms between different racial groups by using the America's longitudinal high school students dataset. Through training and testing several machine learning algorithms, the accuracy metrics were compared between specifically the WA (White and Asian) group and the BH (Black and Hispanic) group.

There are a few findings from this study. All seven models consistently underestimated the percentage of students who would be in the low 50 for the students' 12th graders' math performance. Secondly, using accuracy metrics alone when measuring group fairness may not be the best metrics especially when the group fairness is a sensitive matter. Therefore, examining the false negative rates, false positive rates, and the F1- score is recommended instead of just glancing at the accuracy alone. For all machine learning models' results, the false negative rates were higher every time while the false positive rates were lower than the WA group's. Despite the attempt to train the models with different subsets of data, the results were still biased against the BH group. For instance, even when the models were trained with a subset data without any implicit racial features, the results were not much different. Additionally, even when the models were trained with a balanced data, which had the same number of the WA and the BH students, the results still favored the WA group, which means the models consistently overestimated the WA's 12th grade math grade performance while underestimating the BH's 12th grade math performance. However, the results were not significant except for the neural network model, which had zero difference on all four metrics.

For any future studies related to measuring group fairness, adversarial learning is recommended and compositing a dataset with different racial groups and weights and use a package that carefully assesses the system's fairness and mitigate any observed unfairness issues such as a Python package 'fairlearn.'

Unbiased machine learning algorithms are incredibly critical as the usage of the machine learning and artificial intelligence has increased significantly in the education sector. Biased machine learning performance results towards the historically underrepresented groups and underprivileged students can continue to widen the existing educational equity gap and the opportunity gap, especially as these underrepresented groups in the education sector are often Black or Latino students.

CHAPTER 8

References

1. A. Julia and J. Larson. *Bias in Criminal Risk Score Is Mathematically Inevitable, Researchers Say.* 2016. https://www.propublica.org/article/bias-in-criminal-risk-scores-is-mathematically-inevitable-researchers-say.

2. A. Julia et al. *Machine Bias.* ProPublica, 2016. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

3. Executive Office of the President. *Big Data: A Report on Algorithmic Systems, Opportunity, and Civil Rights.* 2016.

4. N. Mehrabi et al. ”A Survey on Bias and Fairness in Machine Learning.” *ACM Computing Surveys* 54.6 (2021).

5. Congressional Research Service. *Artificial Intelligence (AI) and Education.* 2018. https://crsreports.congress.gov/product/pdf/IF/IF10937.

6. K. Adam. *The U.K. used an algorithm to estimate exam results. the calculations favored elites.* 2020. https://www.washingtonpost.com/world/europe/the-uk-used-an-algorithm-to-estimate-exam-results-the-calculations-favored-elites/2020/08/17/2b116d48-e091-11ea-82d8-5e55d47e90ca_story.html.

7. H. Jeong et al. *Who Gets the Benefit of the Doubt? Racial Bias in Machine Learning Algorithms Applied to Secondary Math Education.* 2022.

8. Tyrone C Howard. *Why Race and Culture Matter in Schools: Closing the Achievement Gap in America's Classrooms.* Teachers College Press, 2019.

9. Javeria Munir and Mehreen Faiza. "The Impact of Socio-economic Status on Academic Achievement." *Journal of Social Sciences Review* (2023).

10. Vonnie McLoyd. "Socioeconomic disadvantage and child development." *American Psychological Association* (1988).

11. D. Jurfafsky and J. Marin. `https://web.stanford.edu/~jurafsky/slp3/5.pdf`.

12. S. Raschka. `https://sebastianraschka.com/pdf/lecture-notes/stat479fs18/02_knn_notes.pdf`.

13. M. Islam, G. Chen, and S. Jin. "An Overview of Neural Network." *American Journal of Neural Networks and Applications* (2019).

14. J. Faraway. *Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models, Second Edition.* Chapman and Hall/CRC, 2016.

15. M. Mayer and J. Wilber. `https://mlu-explain.github.io/equality-of-odds/`.

16. W. Jiang and Z. Pardos. "Towards Equity and Algorithmic Fairness in Student Grade Prediction." *American Journal of Neural Networks and Applications* (2019).