STUDENT GRADE PREDICTION

Xyrus Rafael C. Gallito, MSc

I. Introduction

Student_grades.csv is the dataset I have chosen. You will see how I have investigated the data, pre-processed it, utilized feature engineering and feature selection, used several classification models, model evaluation using different metrics, used parameter tuning for the classification model with the lowest performing model metrics, and obtained and reflected on the results from here on out.

II. Exploratory Data Analysis

A. Table 1 was produced using describe() function from “psych” package and using flextable() function from “flextable” package.

Table 1. Initial descriptive statistics of the dataset.

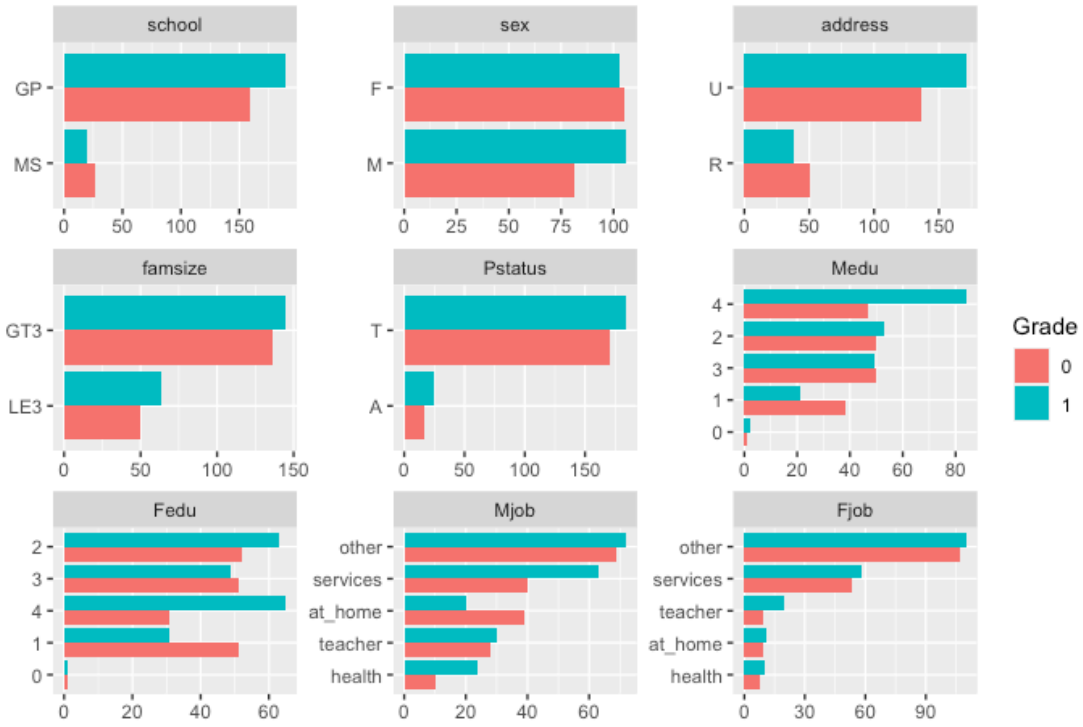
vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
school	395	1.1164557	0.3211774	1	1.0220820	0.0000	1	2	1	2.38231776	3.68478569	0.01616019
sex	395	1.4734177	0.4999261	1	1.4668770	0.0000	1	2	1	0.10607560	-1.99377629	0.02515401
age	395	16.6962025	1.2760427	17	16.6277603	1.4826	15	22	7	0.46273485	-0.03144581	0.06420468
address	395	1.7772152	0.4166428	2	1.8454259	0.0000	1	2	1	-1.32734030	-0.23875296	0.02096357
famsize	395	1.2886076	0.4536897	1	1.2365931	0.0000	1	2	1	0.92952179	-1.13885320	0.02282761
Pstatus	395	1.8962025	0.3053844	2	1.9936909	0.0000	1	2	1	-2.58820974	4.71077489	0.01536556
Medu	395	2.7493671	1.0947351	3	2.8201893	1.4826	0	4	4	-0.31596669	-1.10106854	0.05508210
Fedu	395	2.5215190	1.0882005	2	2.5331230	1.4826	0	4	4	-0.03143195	-1.20768218	0.05475331
Mjob	395	3.1696203	1.2274726	3	3.2113565	1.4826	1	5	4	-0.33264692	-0.69032643	0.06176085
Fjob	395	3.2810127	0.8635419	3	3.3186120	0.0000	1	5	4	-0.35955631	0.97965353	0.04344951
reason	395	2.2556962	1.2082360	2	2.1955836	1.4826	1	4	3	0.40646235	-1.40335492	0.06079295
guardian	395	1.8531646	0.5366836	2	1.8391167	0.0000	1	3	2	-0.11069990	0.14821902	0.02700348
traveltime	395	1.4481013	0.6975048	1	1.3123028	0.0000	1	4	3	1.59484442	2.27267107	0.03509527
studytime	395	2.0354430	0.8392403	2	1.9589905	0.0000	1	4	3	0.62734922	-0.04442326	0.04222676
failures	395	0.3341772	0.7436510	0	0.1388013	0.0000	0	3	3	2.36892701	4.88636521	0.03741714
schoolsup	395	1.1291139	0.3357513	1	1.0378549	0.0000	1	2	1	2.20369761	2.86355186	0.01689348
famsup	395	1.6126582	0.4877606	2	1.6403785	0.0000	1	2	1	-0.46077117	-1.79220795	0.02454190
paid	395	1.4582278	0.4988839	1	1.4479495	0.0000	1	2	1	0.16703845	-1.97708422	0.02510157
activities	395	1.5088608	0.5005555	2	1.5110410	0.0000	1	2	1	-0.03531408	-2.00380663	0.02518568
nursery	395	1.7949367	0.4042599	2	1.8675079	0.0000	1	2	1	-1.45544995	0.11865417	0.02034052
higher	395	1.9493671	0.2195250	2	2.0000000	0.0000	1	2	1	-4.08363024	14.71330398	0.01104550
internet	395	1.8329114	0.3735281	2	1.9148265	0.0000	1	2	1	-1.77801132	1.16429103	0.01879424
romantic	395	1.3341772	0.4723003	1	1.2933754	0.0000	1	2	1	0.70041453	-1.51323124	0.02376400
famrel	395	3.9443038	0.8966586	4	4.0378549	1.4826	1	5	4	-0.94466441	1.08945930	0.04511579
freetime	395	3.2354430	0.9988620	3	3.2302839	1.4826	1	5	4	-0.16211221	-0.32673913	0.05025820
goout	395	3.1088608	1.1132782	3	3.0851735	1.4826	1	5	4	0.11561908	-0.78693434	0.05601510
Dalc	395	1.4810127	0.8907414	1	1.2681388	0.0000	1	5	4	2.17415123	4.64544872	0.04481807
Walc	395	2.2911392	1.2878966	2	2.1514196	1.4826	1	5	4	0.60732003	-0.80716656	0.06480111
health	395	3.5544304	1.3903034	4	3.6908517	1.4826	1	5	4	-0.49085343	-1.02646942	0.06995376
absences	395	5.7088608	8.0030957	4	4.2397476	5.9304	0	75	75	3.64374060	21.30650507	0.40267945
Pass	395	0.5291139	0.4997847	1	0.5362776	0.0000	0	1	1	-0.11621091	-1.99151763	0.02514690

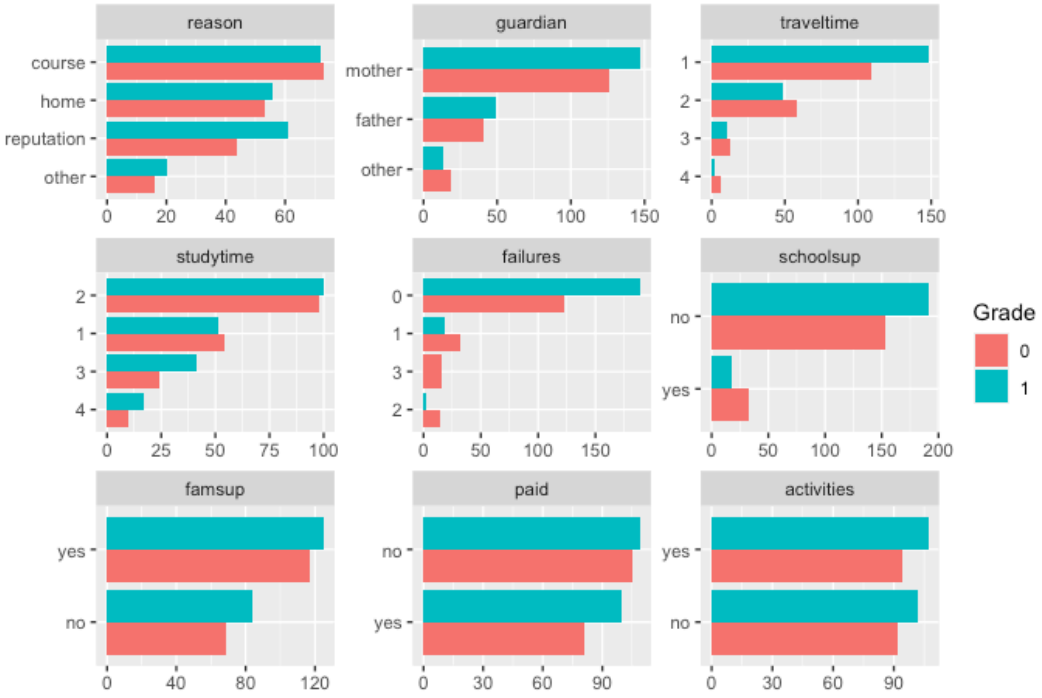
Along with describe() function from “Hmisc” package, the observations are as follows:

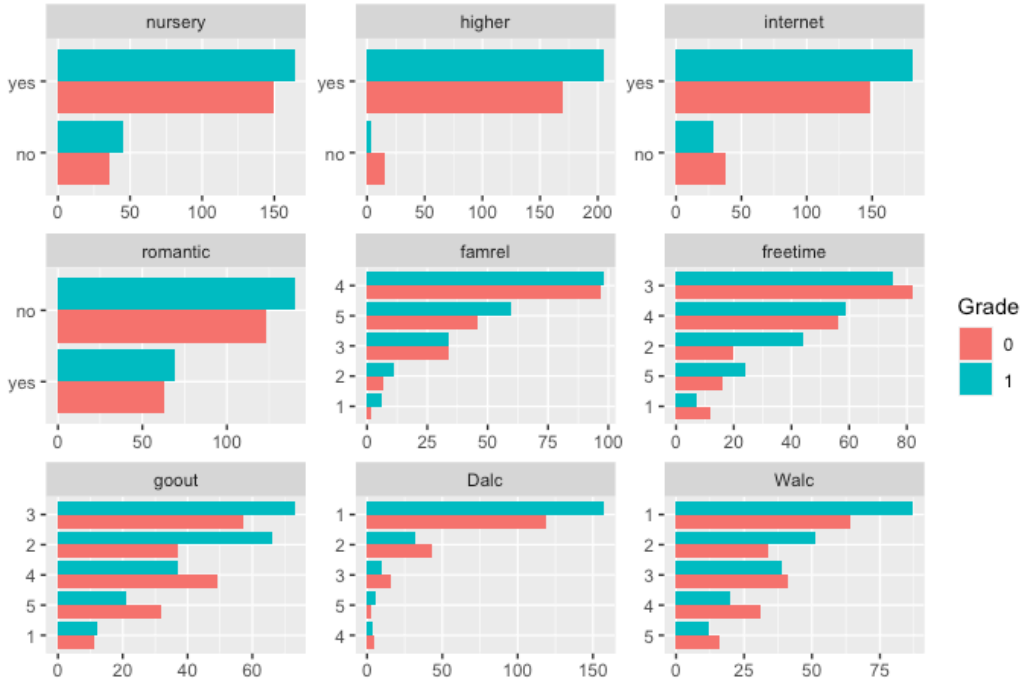
1. The dataset has 395 rows and 31 columns.
2. Only “absences” columns can stay as numerical type while “age” will be treated as categorical variable after data pre-processing. Other columns can be treated as categorical variables.
3. “school” variable has GP and MS values with frequency of 349 and 46, respectively. 88% of the students were attending GP school while other are from MS.
4. “sex” variable has “F” (female) and “M” (male) with frequency of 208 and 187, respectively. Most of the students are female (53%).
5. The “age” range of the students is 15 and 22 years old with the mean age of about 17. Majority of the student were 16 years old, and most ages fall between 15-18.
6. “address” has two values “R” (88 counts) for rural and “U” (307) for urban. 77% of the students live in urban areas.
7. 71% of the students have a family size of greater than 3 while the remaining 29% students live with a family less than or equal to 3.
8. Students whose parents are still living together account for 90% of the data, whereas 10% of students have parents who are separated.
9. 33% of the students have a mother with higher education (“medu”) and on 0.8% without any sort of qualifications.
10. The highest proportion for students’ father education qualification (“fedu”) is school-level (29%) while students whose father have no qualifications is only 0.8%
11. Students' mother ("Mjob") and father ("Fjob") occupations are disproportionately represented in the "others" category, with 36% and 55%, respectively.
12. Majority of the students chose their school based on “home” (28%) and “reputation” (27%). Let us assume the “home” means “near house of the students”.
13. Majority of the students (69%) have their mother as the guardian while only 23% have their father as the guardian.
14. 65% of the students travel to school for less than 15 minutes and 27% students have travel time between 15 to 30 minutes. Also, very few students travel to school more than 30 minutes.
15. Half of the students studied 2 to 5 hours per week. Only 7% of them study for more than 10 hours per week.
16. 79% of the students do not have previous assessment failures. Only 8% of them have either 2 or 3 previous assessment failures.
17. Only 13% of the students have school supplies.
18. 61% of the students have family education support while 39% of them do not have any support from their family.

19. 214 of the students have no extra paid in class while the 181 students have extra.
20. Students who participate in extracurricular activities and those who do not are nearly identical.
21. Most of the students attended nursery school (80%).
22. 95% of the students want to pursue a higher education.
23. Only 17% of the students do not have access to the internet.
24. 67% of the student do not engage in romantic relationship.
25. Nearly half of the students have high quality relationships with their family. Only 2% of them have very bad relationship rating.
26. Majority of the students have fair amount of free time after school. Same with time going out with friends.
27. 70% of the students consume very low amount of alcohol during weekdays. and 38% of them drink alcohol during weekends.
28. Majority of the students (37%) have very good overall health conditions and 23% of them have just good health conditions.
29. The average absences of the students are around 6. The maximum number of absences is 75.
30. 53% of the student passed while 47% failed. "Pass" column will be renamed to "Grade".

B. Below is distribution of most of the features grouped by Grade column.

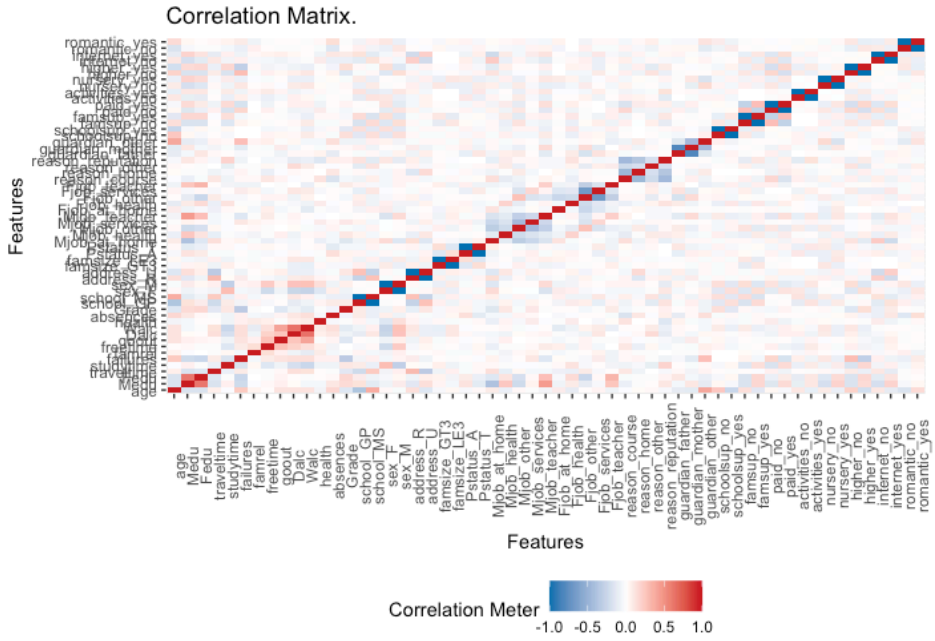



reason
course
home
reputation
other
0
20
40
60
guardian
mother
father
other
0
50
100
150
traveltime
1
2
3
4
0
50
100
150
studytime
2
1
3
4
0
25
50
75
100
failures
0
1
3
2
0
50
100
150
schoolsup
no
yes
0
50
100
150
200
famsup
yes
no
0
40
80
120
paid
no
yes
0
30
60
90
activities
yes
no
0
30
60
90
Grade
0
1


nursery
yes
no
0
50
100
150
higher
yes
no
0
50
100
150
200
internet
yes
no
0
50
100
150
romantic
no
yes
0
50
100
famrel
4
5
3
2
1
0
25
50
75
100
freetime
3
4
2
5
1
0
20
40
60
80
goout
3
2
4
5
1
0
20
40
60
Dalc
1
2
3
5
4
0
50
100
150
Walc
1
2
3
4
5
0
25
50
75
Grade
0
1

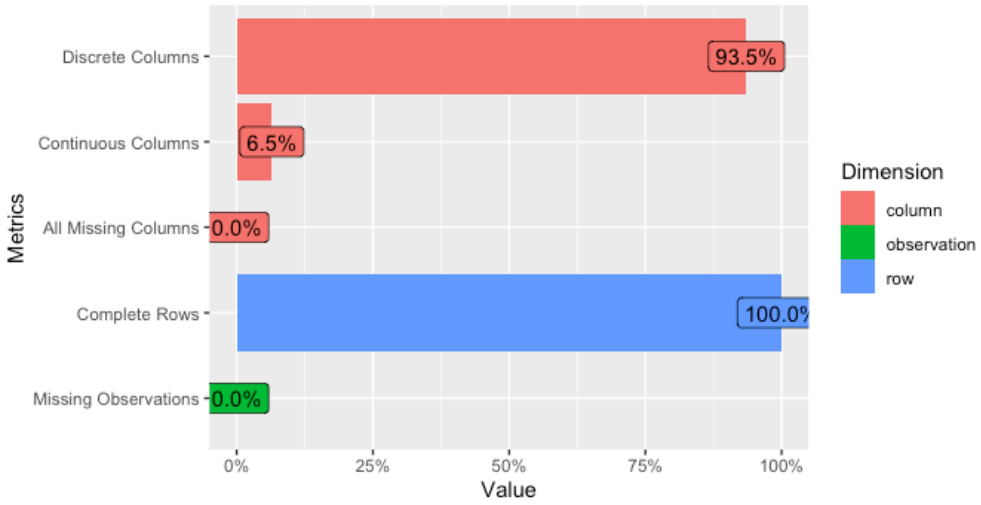
C. Correlation Matrix and Data Glimpse

As you may see, the correlation matrix below is not very useful in determining the relationship between variables. Thus, I will not use correlation test as one of the measurements of the relationship between the predictors and the response variable "Grade".



According to Figure 1, there are no missing data with complete rows. 93.5% of the columns or features are categorical (discreet) variables and 6.5% are continuous variables.

Figure 1. Glimpse of the dataset.



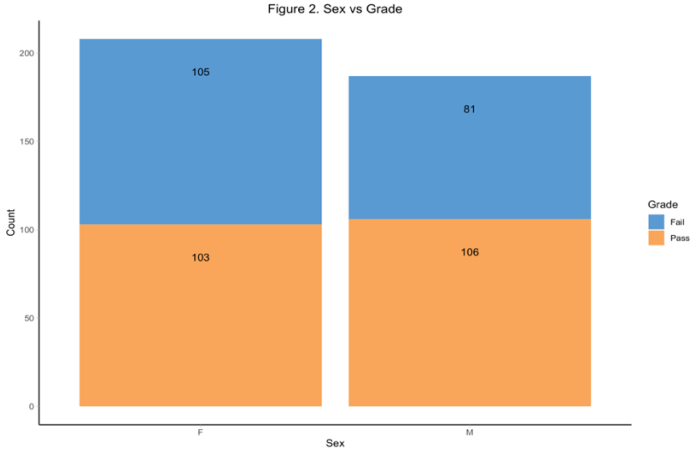
D. Graphs and Distributions

The section are the graphs that will show the relationship of some features that I think might affect the grade and will present them with more details.

1. Sex and Grade

Figure 1 shows the distribution of grades based on the gender of the students. Females are more likely to fail than males. Both genders receive similar "Pass" grade.

Figure 2. Sex vs Grade



2. Age and Grade

The boxplot of "age" vs "Grade" is shown in Figure 3. There are clear outliers or noise from data. Table 1 shows that most students are between the ages of 15 and 18. Only 5 students between the ages of 20 and 22 were present. Only a total 5 of students with the age from 20 to 22 years old. To clear the outliers, I will remove the age bracket of 20 to 22. After this, we can consider the "age" column as categorical variable.

Figure 3. Age vs Grade

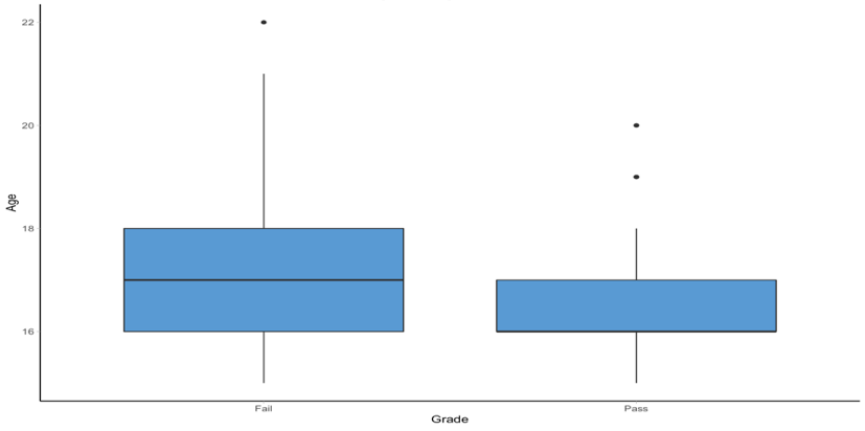
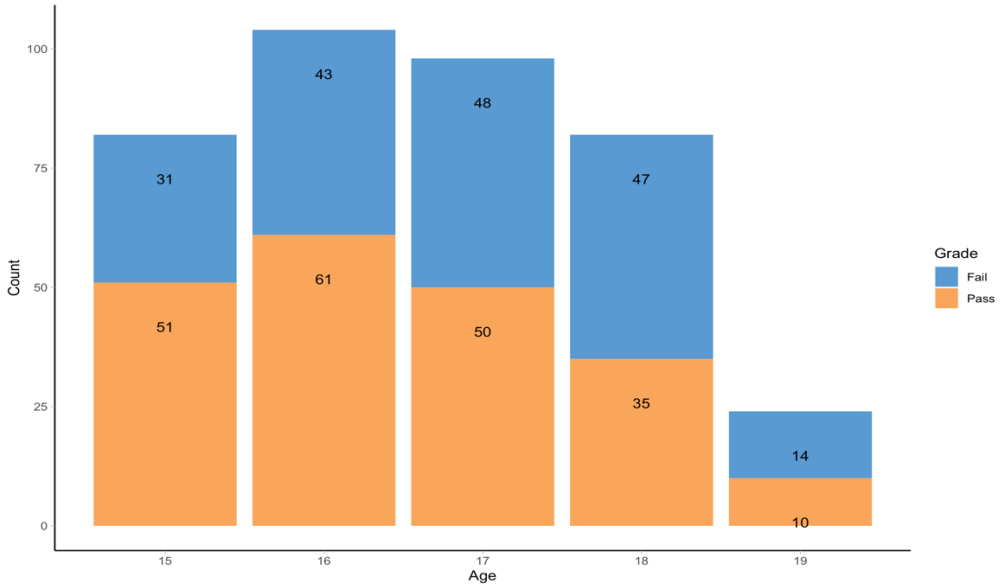


Figure 4 shows the updated distribution of “age”. There are more students who pass at age 15 to 18. Student who failed are slightly higher than who have passed.

Figure 4. Age vs Grade



3. Absences and Grade

Figure 5 shows the boxplot of absences and looks like those students who pass, and fail have nearly the same median. Furthermore, we can see in Figure 6 that the increase in absences does not affect the grades of the students.

Figure 5. Absences vs Grade

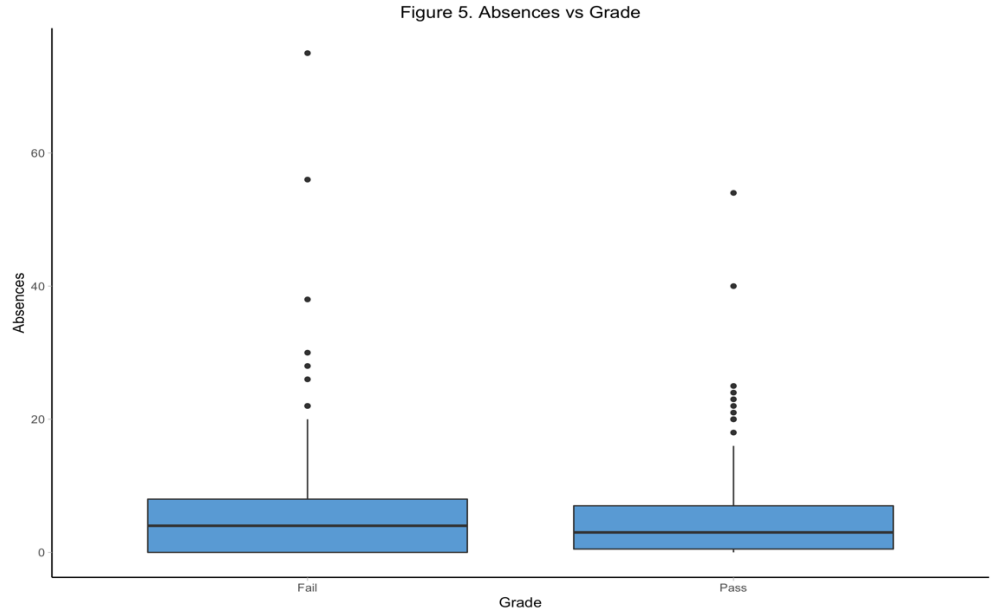
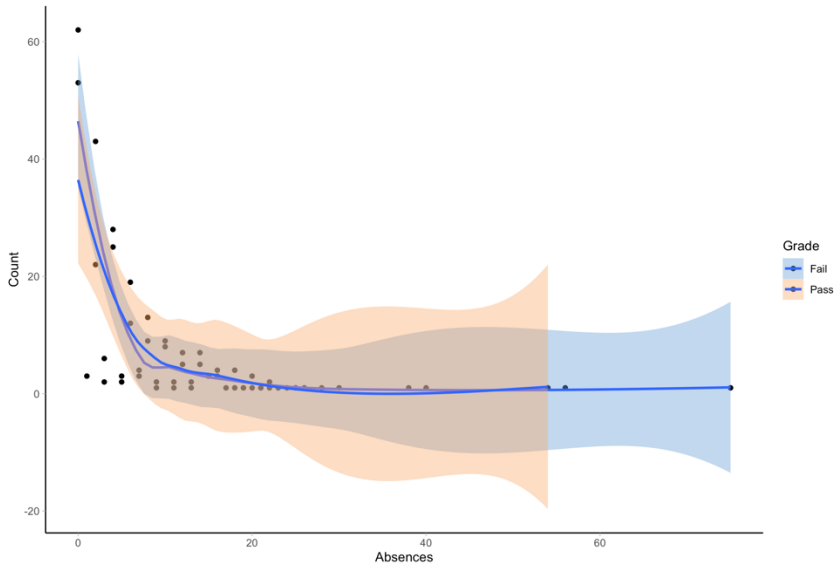


Figure 6. Abesences vs Grade

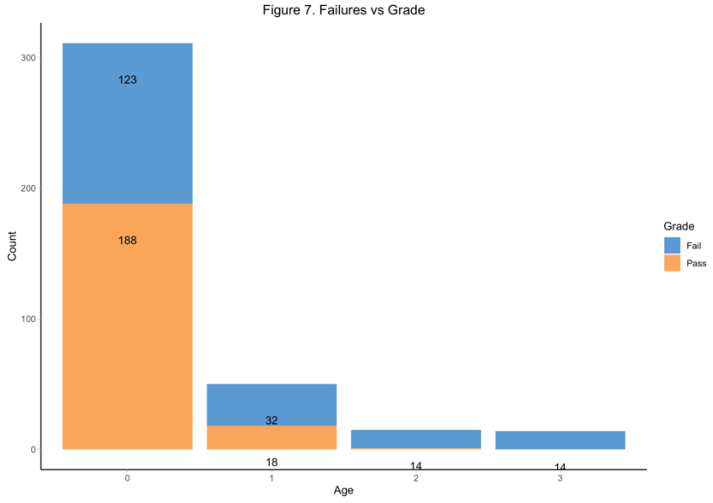


According to Shapiro-Wilk Normality test, both absences for Fail and Pass grades do not follow normal distribution as p-values were less than 0.05. I will then use Wilcoxon test, a non-parametric alternative to the paired two-samples t-test for comparing two independent groups of samples. It gives a p-value = 0.9887 which is greater than the significant values. Thus, the difference of the median is not significant.

4. Failures and Grade

Figure 7 shows the distribution of failures against the Grade. We can see that as number of absences increases, the number of Grade (Pass and Fail) decreases. At this point, we cannot tell if the relationship between “failures” and “Grade” is statistically significant.

Figure 7. Failures vs Grade



E. Hypothesis Testing

Chi-Square test, also a non-parametric alternative to t-test, will be used to know which of the predictors have a significant correlation to the response variable “Grade”. The “absences” columns, which is the only non-categorical variable in this dataset had its hypothesis tested using Wilcoxon Test.

Features	p-value
school	0.1613665
sex	0.1340564
age	0.06074484
address	0.06133394
famsize	0.4306935
Pstatus	0.3436295
Medu	0.009243514
Fedu	0.003594972
Mjob	0.003349944
Fjob	0.4628998
reason	0.5544845
guardian	0.3153641
traveltime	0.03616726
studytime	0.217741
failures	0.000000005243991
schoolsup	0.009910084
famsup	0.5474266
paid	0.477022
activities	1
nursery	0.5942428
higher	0.008483872
internet	0.1015775
romantic	0.9142962
famrel	0.4541653
freetime	0.02182295
goout	0.0181501
Dalc	0.09545385
Walc	0.113931
health	0.2813967

If the p-value is less than the significant value of 0.05, the null hypothesis will be rejected and will conclude that the predictor and response variable have a significant relationship. According to the Chi-square test, the predictors with a significant connection from the response variable "Grade" are "Medu", "Fedu", "Mjob", "traveltime", "failures", "schoolsup", "higher", "freetime", and "gout." I willl base the feature engineering and feature selection on the chi-square test result.

III. Feature engineering

The feature engineering technique is discussed in this section. This section also contains a brief description of the models and establishes the dataset's baseline accuracy.

There is no need to make distinction between the two types of the school. The “address” appears to have less predictive capacity compared to "traveltime" which determines how long each student will have to commute. Based on the Chi-square test above, these two features do not have significant relationship from the response variable. Thus, “school” and “address” variables will be excluded in this part of discussion. This section will not perform predictive model performance yet. This will only give initial accuracy without parameter tuning.

A. Generalized Linear Models (GLM)

Generalized Linear Models are a type of regression and classification model that models the response variable, Y, and the random error term using exponential distributions as normal, Poisson, Gamma, Binomial, inverse Gaussian, and so on. The exponential family of distributions is assumed in GLM. This differs from conventional linear regression models, which need the response variable, Y, and the random error term to have a normal distribution. For GLM, log-likelihood is used to generalize the analysis of variance (Nelder & Wedderburn, 1972).

Using k-fold (10-fold) cross validation to measure the skill of the data, GLM was used and gave an accuracy of 62%. The model runs 390 samples with 28 predictors and two classes “0” and “1” (Pass or Fail). Data where not split in this model.

B. Decision Tree (DT)

Decision Tree is a supervised machine learning algorithm that can used for both regression and classification analysis. The purpose of this technique is to develop a model that predicts the value of a response variable, and the decision tree solves the problem by using the tree representation, where the leaf node corresponds to a class label and characteristics are represented on the internal node of the tree (Sharma, 2021).

Using 390 samples with cross-validation of 10-fold, the model gave 55%. The final value used for the model was cp = 0.02003643.

C. Random Forest (RF)

While decision trees are a typical supervised learning technique, they can have issues including bias and overfitting. When numerous decision trees are combined in a random forest algorithm, the results are more accurate, especially when the individual trees are uncorrelated. An RF is nothing more than a series of decision trees with their findings combined into a single result. They are so powerful because of their ability to limit overfitting without significantly increasing error due to bias (Liberman, 2017)

The algorithm was cross validated with 10 folds. It also used 390 samples and tested three different values of mtry: 2, 36, 70. The final value used for the model was mtry = 36 with an accuracy of 63%.

D. Support Vector Machine (SVM)

SVM, a supervised machine learning model, uses classification algorithm with two or more classes (Telrandhe, et al., 2016). SVM uses a straight line or hyper plane to separate each plotted data points in n-dimensional space into two or more classes. Using 10-fold cross-validation using Linear Kernel with 390 samples, an accuracy of 63% was obtained.

E. Extreme Gradient Boost (XGBoost)

Gradient boosting is a method for predicting the residuals or errors of prior models, which are then merged to generate the final prediction. Gradient boosting gets its name from the fact that it uses a gradient descent approach to minimize loss when adding new models (Brownlee, 2016).

Extreme gradient boosting incorporates regression penalties into the boosting equation (like elastic net) and takes advantage of the structure of the hardware to reduce computation times and memory usage.

Using 10-fold cross-validation, 395 samples, eta of 0.4, *max_depth* of 1, gamma default of 0, *colsample_bytree* of 0.6, *min_child_weight* of 1, subsample of 0.75, *nrounds* equal 100, the model generated a 64%. The highest accuracy among the model tested above.

This assignment's baseline accuracy is summarized in table below. The XGBoost model had the best accuracy, while the DT model had the worst. The following section will cover feature selection utilizing two different strategies.

Model	Baseline Accuracy
1. GLM	62%
2. Decision Tree	55%
3. Random Forest	63%
4. Support Vector Machine	63%
5. XGBoost	64%

IV. Feature Selection

Due to redundancy and irrelevancy, some features in any dataset might be unimportant. As a result, considering such characteristics is counterproductive and usually results in low classification accuracy. As a result, feature selection aims to improve classification efficiency by selecting only a small subset of relevant features from a large initial collection. As a result, removing irrelevant and redundant features will reduce data dimensionality, speed up the learning process by simplifying the learnt model, and improve performance (Al-Tashi, et al., 2020).

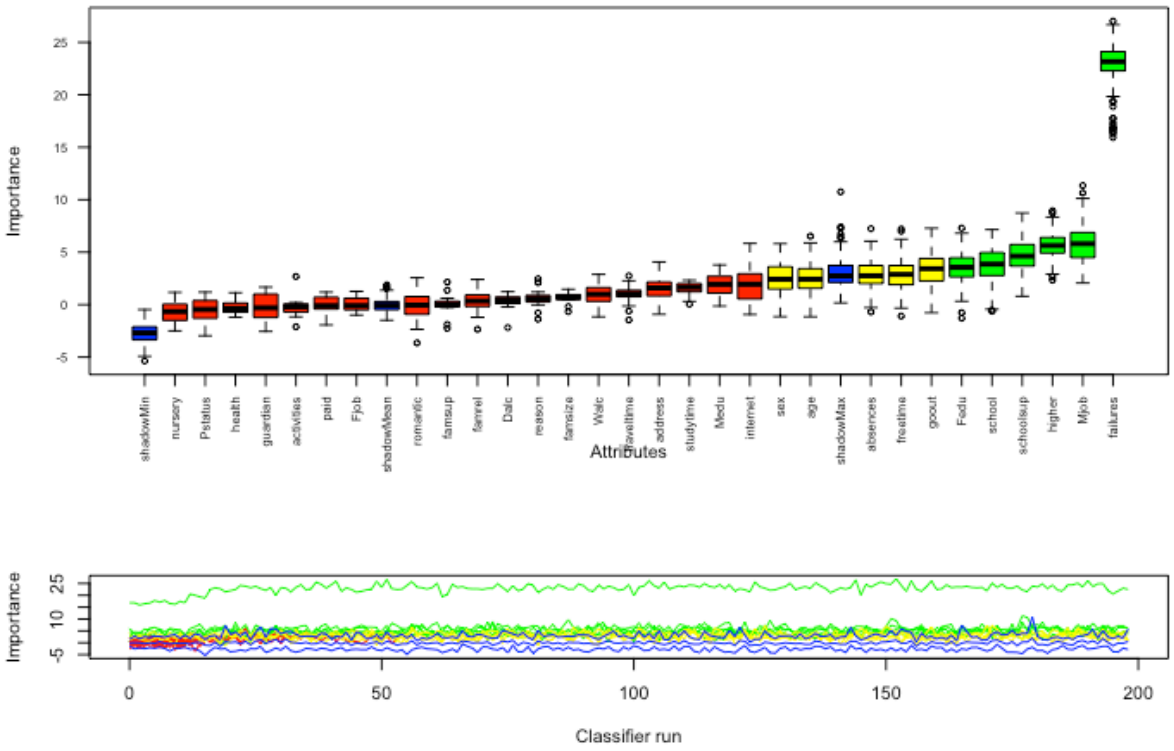
Because the dataset only comprises 31 features, it can be considered small in a real-world setting. However, I have chosen to apply feature selection approaches like Boruta and Recursive Feature Elimination to determine the most essential features that could improve the accuracy of my predictive models. The results from these techniques will be compared to chi-square test result.

A. Boruta

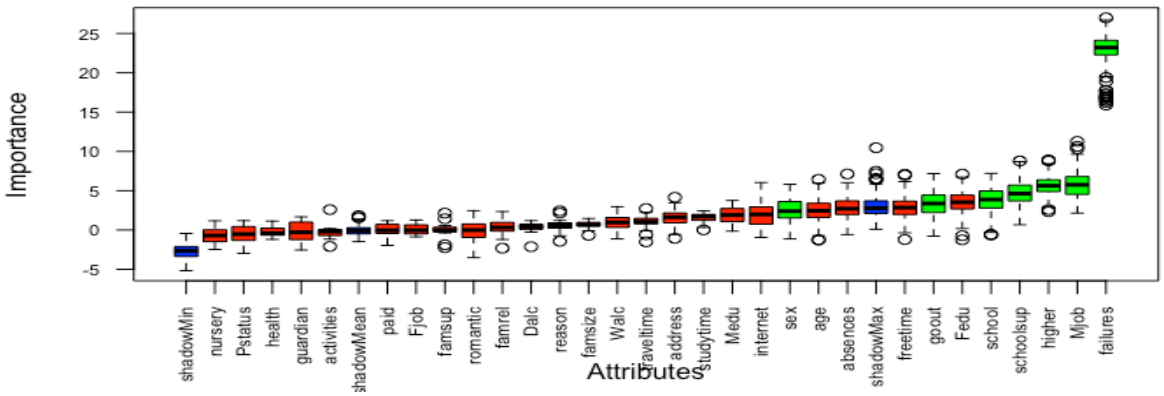
Boruta follows the logic behind Random Forest classifier. Random fluctuations and correlations can be decreased by introducing randomization into the system and collecting data from a set of randomized samples. This added randomization called shadow features will allow us to see which variables are genuinely significant to the model. The general process of Boruta is quite simple. It will first add randomness or shadow features in the data set and then train the random forest classifier to determine or evaluate the most important features using Mean Decrease Accuracy. After every iteration, it will compare the Z score of real features to its shadow features' Z scores. If the Z score is lower than the Z score among its shadow features, it will be categorized as "unimportant" feature (Kursa & Rudnicki, 2010).

Boruta is easy to use because it has only few parameters. In the code, I have used 100 and 200 for the maximal number of importance source runs (maxRuns) which gave

me same confirmed important attributes. According to the Boruta documentation, increasing the iterations will resolve the tentative features. So, I have used 200 instead of 100. Using all the features from the data set, ranking of features based on importance was generated using boxplot



After 200 iterations, the confirmed important attributes are “failures”, “Mjob”, “higher”, “schoolsup”, “school”, and “Fedu” (colored in green) while the tentative attributes left are “goout”, “freetime”, “absences”, “age”, and “sex” (colored in yellow). All those attributes colored in red were rejected by Boruta. As you may recall, as part of feature engineering, I removed the "address" and "school" attributes, and Boruta chooses “school” as one of the important variables. The tentative attributes, which Boruta is unsure of their classification because of the closeness to their respective best shadow attributes, will be fixed by using a built-in function TentativeRoughFix(). The tentative attributes will be identified as confirmed or rejected by comparing the median Z score of the characteristics to the median Z score of the best shadow attribute. The Boruta result given below promoted “sex” from tentative to confirmed important attribute.



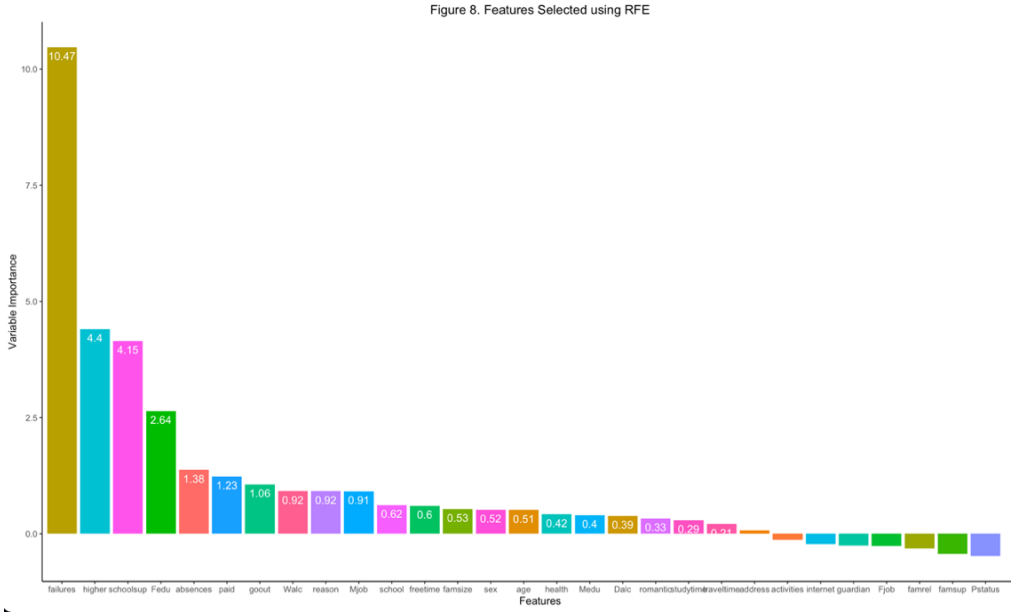
B. Recursive Feature Elimination (RFE)

I would like to compare the selected features using Boruta against RFE, a conventional data science feature elimination technique. RFE, like Boruta, is a wrapper algorithm for Random Forest but uses an internal filter-based feature selection technique. When it comes to feature selection, RFE's rationale is comparable to that of random forest. The primary distinction is that RFE does not rank the weights of the features; instead, it eliminates the smallest weights on each iteration. Thus, removing those attributes with low predicting ability for the response variable (Lin, et al., 2017).

Using all the features, the RFE control was produced using random forest function (rfFuncs). And the outer resampling method used was cross validation (cv) with 10-fold. The dataset was split into 70-30 partition (i.e., 70% training set, 30% test set).

Figure 8 shows that the selected features using RFE are “failures”, “higher”, “schoolsup”, “Fedu”, and “absences”.

Figure 8. Features Selected using RFE



4 out 5 features chosen by RFE were also present in both Chi-square test Boruta chosen features. “Fedu” were chosen by RFE and Chi-square test. “Mjob” was chosen by Boruta and Chi-square test. I will add those “Fedu” and “Mjob” based on EDA and feature selection results. The results employing the selected features [Table 2] will be shown in the next section, and you will see how they affect model accuracy.

Table 2. Selected Features
1. failures
2. schoolsup
3. higher
4. goout
5. Fedu
6. Mjob

V. Comparison of the results

Model	Baseline Accuracy	Accuracy (Selected Features)
1. GLM	62%	68%
2. Decision Tree	55%	57%
3. Random Forest	63%	63%
4. Support Vector Machine	63%	65%
5. XGBoost	64%	68%

Only RF did not generate improvement in the accuracy. Although, it did not get worse. All other models have their accuracy improved by using the selected features.

GLM, DT, and RF models have improved accuracy. The accuracy of SVM and XGBoost has decreased slightly. Accuracy is not the only basis of how a certain model preforms. The next section will be the model evaluation of each model and will generate other model metrics.

VI. Model Evaluation

To avoid bias and reduce the overfitting in the model evaluation, data will be split into training and test sets. The main idea behind data partitioning is to retain a subset of accessible data out of the analysis process and use it afterwards for model testing. The partition would be 70% for training set and 30% for test set. According to my machine learning laboratory instructor, this ratio is appropriate for small datasets because a larger portion of the testing set leads to smaller variance in performance measures. This would make much more sense using the selected features. Confusion matrix, ROC and AUC, accuracy, precision, recall, F1-score, and MCC will be calculated for each model.

The dimension of the train set has 276 rows and 6 columns while the test set has a dimension of 119 rows and 6 columns.

A confusion matrix is a way of summarizing a classification algorithm's performance. When there are an unequal number of observations in each class or more than two classes in your dataset, classification accuracy alone can be misleading. A confusion matrix can help you see what your classification model gets right and where it goes wrong (Brownlee, 2020; Chan, 2022)

According to Simplilearn.com, these are the four different combinations in confusion matrix:

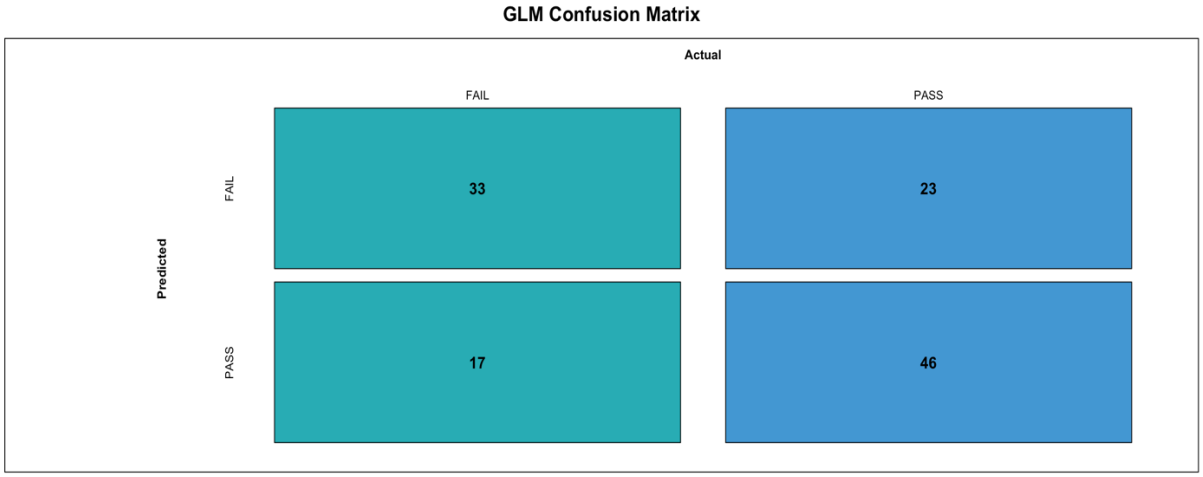
1. True Positive (TP): Actual positive values were predicted correctly.
2. False Positive (FP): Actual positive values were incorrectly predicted as negative values.

3. True Negative (TN): Actual negative values were predicted correctly.
4. False Negative (FN): Actual negative values were incorrectly predicted as positive values.

Since accuracy may not be enough to measure the performance of the model, the Precision (positive predictive value), Recall (the true positive rate), and F1-score (metric of the Precision and the Recall) were also calculated for the model evaluation. Mathew's correlation coefficient (MCC), a metric that summarizes the confusion matrix, is also calculated. Closer to 1 is the ideal score for MCC and 0 is no agreement and prediction is totally random according to the actual values

The Receiver Operator Characteristic (ROC) curve is a graphical representation of binary classifiers' diagnostic ability. It illustrates the relationship between sensitivity and specificity. The performance of classification models with curves that are perfectly aligned to the upper section square mean a perfect performance which is rare. To compare several classifiers, combining their performance into a single measure can be advantageous using the area under the ROC curve (AUC) method. The greater the AUC, the better the classification model's performance [Chan, C., 2022].

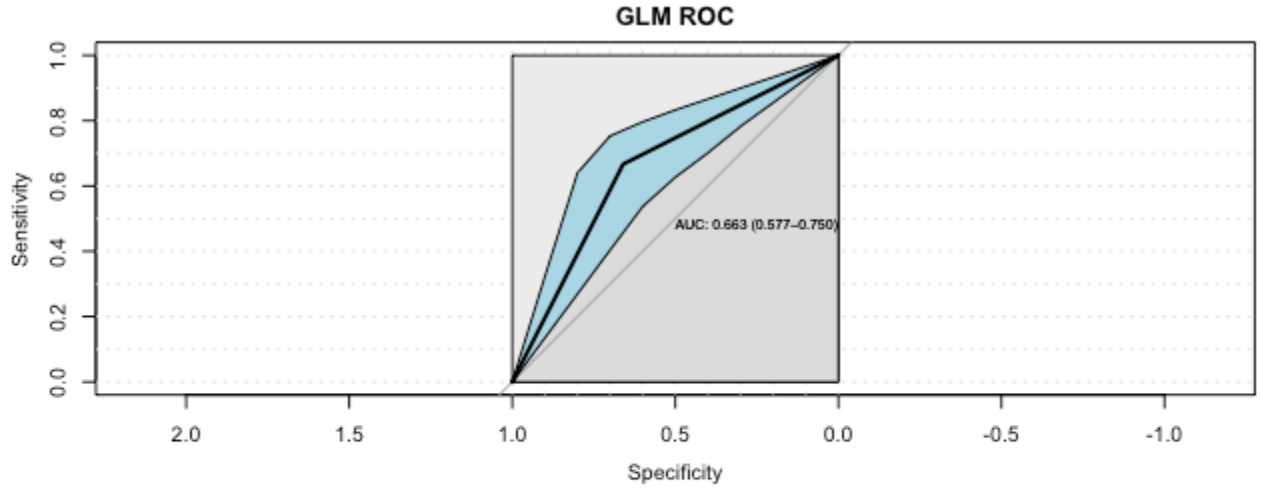
A. GLM



Confusion matrix statistics

Sensitivity	Specificity	Precision	Recall	Balanced Accuracy
0.66	0.67	0.59	0.66	0.66

Accuracy	Kappa
0.66	0.32

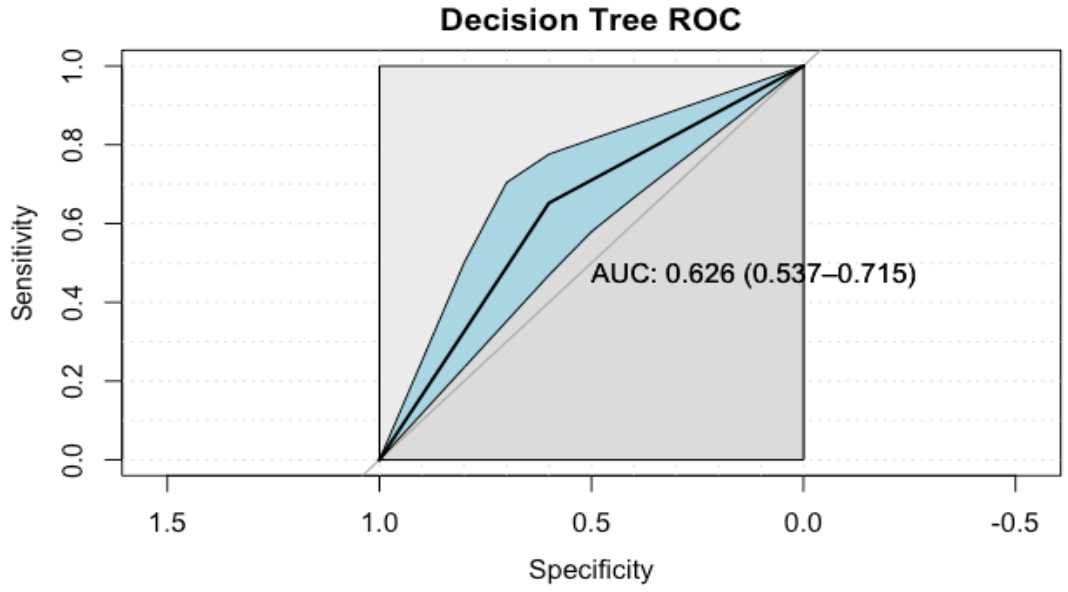


B. Decision Tree

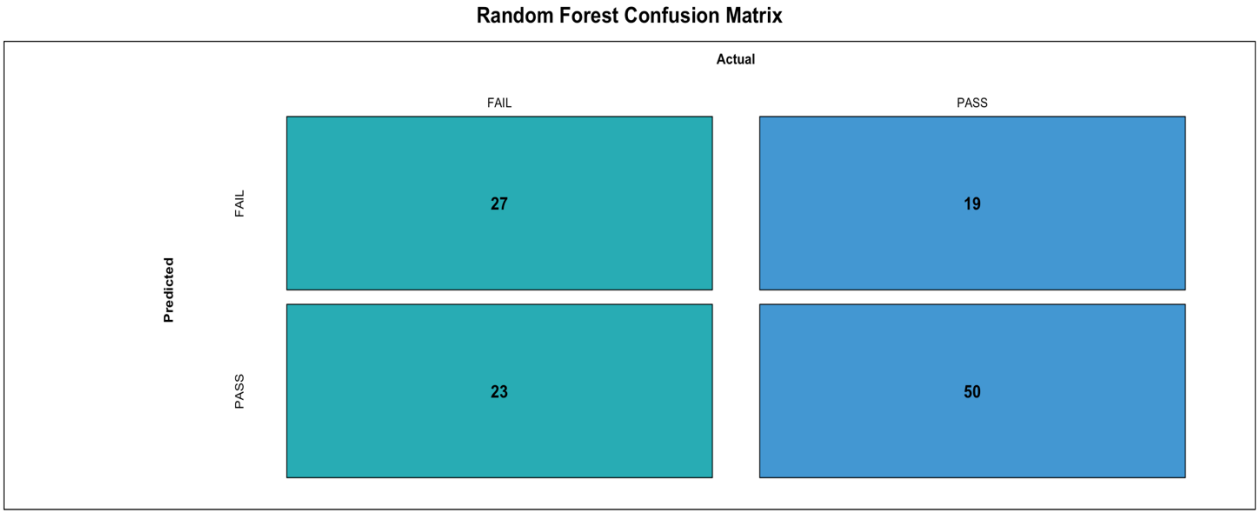
Confusion matrix statistics

Sensitivity	Specificity	Precision	Recall	Balanced Accuracy
0.6	0.65	0.56	0.6	0.63

Accuracy	Kappa
0.63	0.25



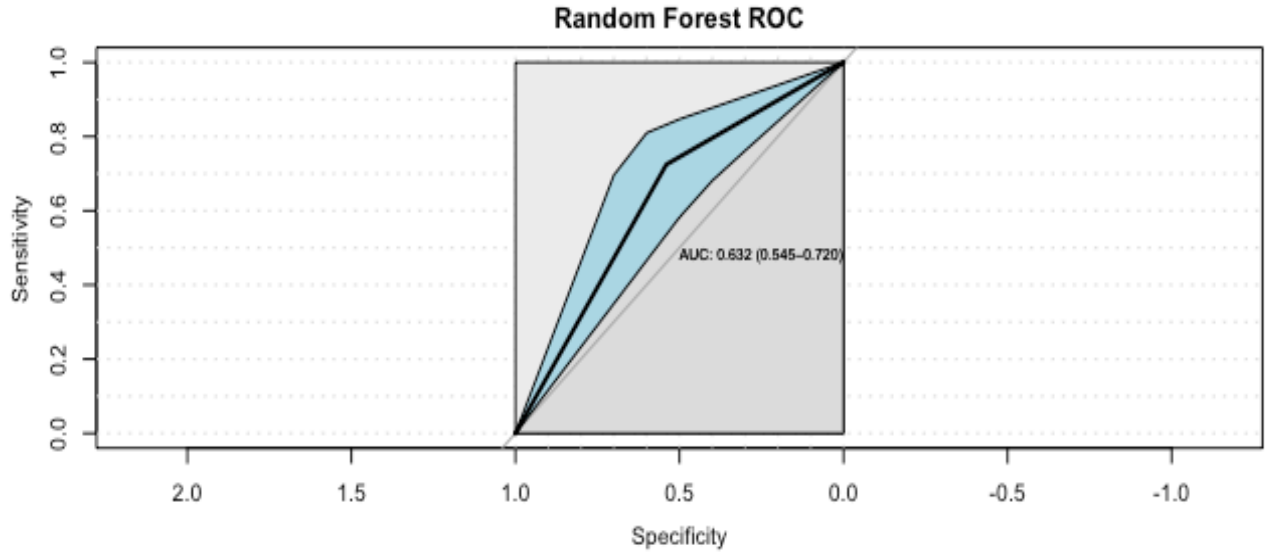
C. Random Forest



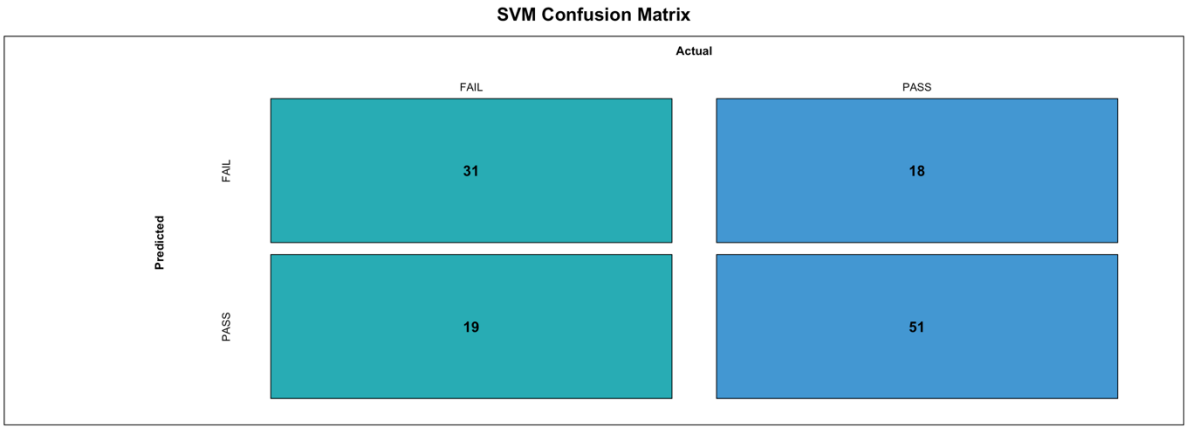
Confusion matrix statistics

Sensitivity	Specificity	Precision	Recall	Balanced Accuracy
0.54	0.72	0.59	0.54	0.63

Accuracy	Kappa
0.65	0.27



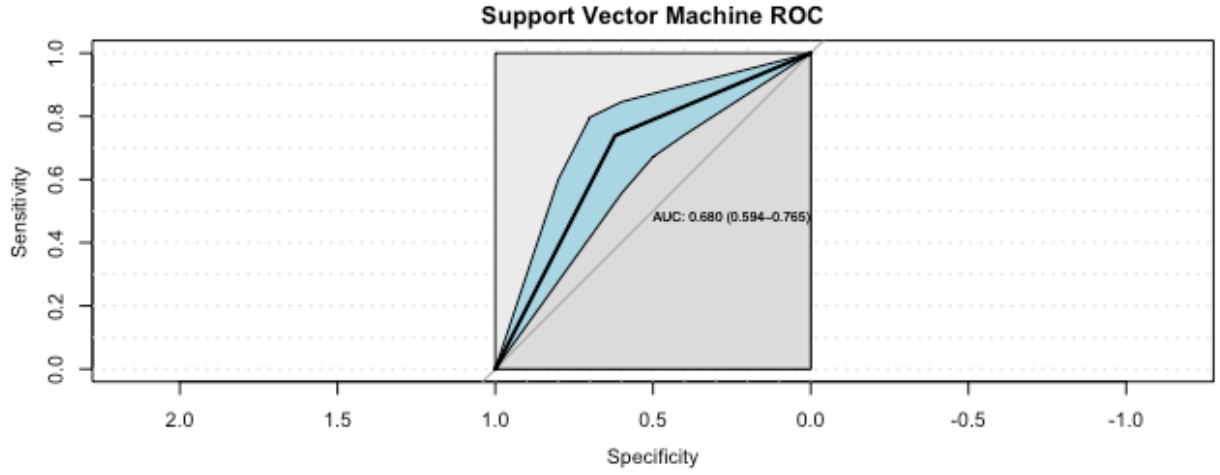
D. SVM



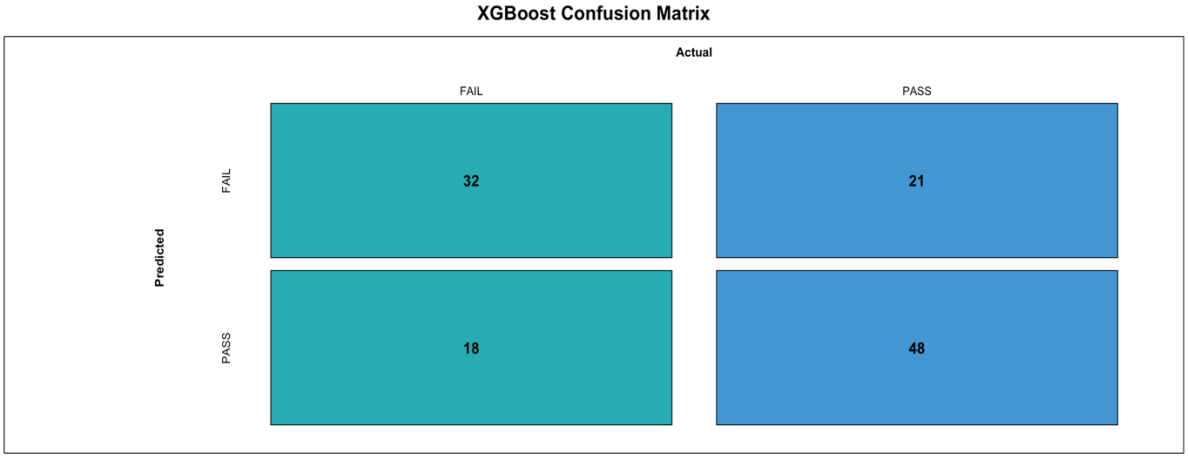
Confusion matrix statistics

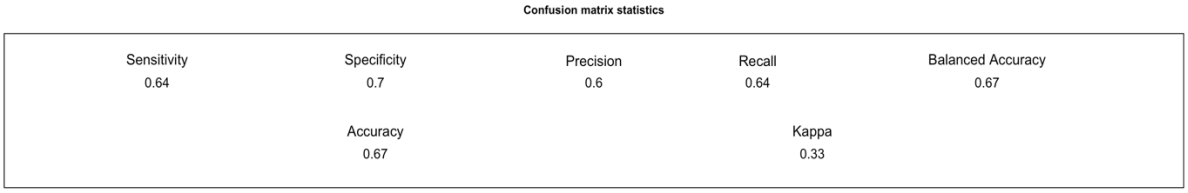
Sensitivity	Specificity	Precision	Recall	Balanced Accuracy
0.62	0.74	0.63	0.62	0.68

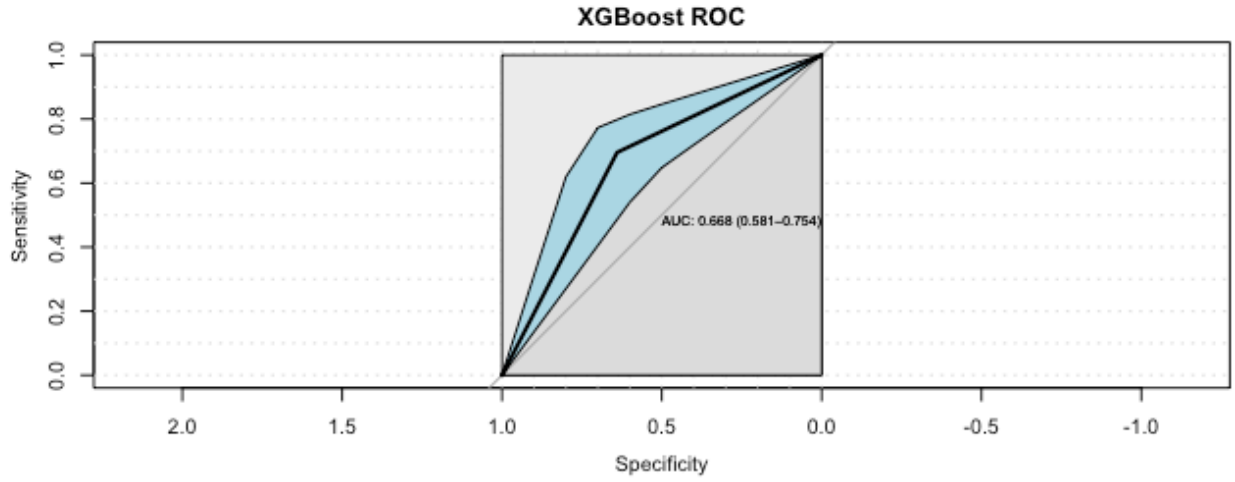
Accuracy	Kappa
0.69	0.36



E. XGBoost


XGBoost Confusion Matrix
Actual
FAIL
PASS
Predicted
FAIL
PASS
32
21
18
48


Confusion matrix statistics
Sensitivity
0.64
Specificity
0.7
Precision
0.6
Recall
0.64
Balanced Accuracy
0.67
Accuracy
0.67
Kappa
0.33


XGBoost ROC
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
AUC: 0.668 (0.581–0.754)
2.0
1.5
1.0
0.5
0.0
-0.5
-1.0
Specificity

Summary table of the model metrics.

Model	Accuracy	Precision	Recall	F1-Score	AUC	MCC
GLM	66%	59%	66%	62%	0.663	0.3230354
DT	63%	56%	60%	58%	0.626	0.2500093
RF	65%	59%	54%	48%	0.632	0.2682384
SVM	69%	63%	62%	52%	0.680	0.3601759
XGBoost	67%	60%	64%	57%	0.668	0.3333413

The highest accuracy was generated by SVM. It was followed by XGBoost and GLM. GLM got the highest F1-score of 62% compared to SVM's 52%. DT got the lowest accuracy even with feature selection. Although, from 57% accuracy from the feature engineering, its accuracy was improved by 6% using selected features. Based on other model metrics, RF seems to be not as good as when it comes to F1-score (48%). AUCs among the models are fairly the same. The top two lowest models in terms of MCC are DT and RF. Overall. I would suggest using SVM, XGBoost and GLM. RF, however, is not bad as it gave high accuracy and competitive AUC.

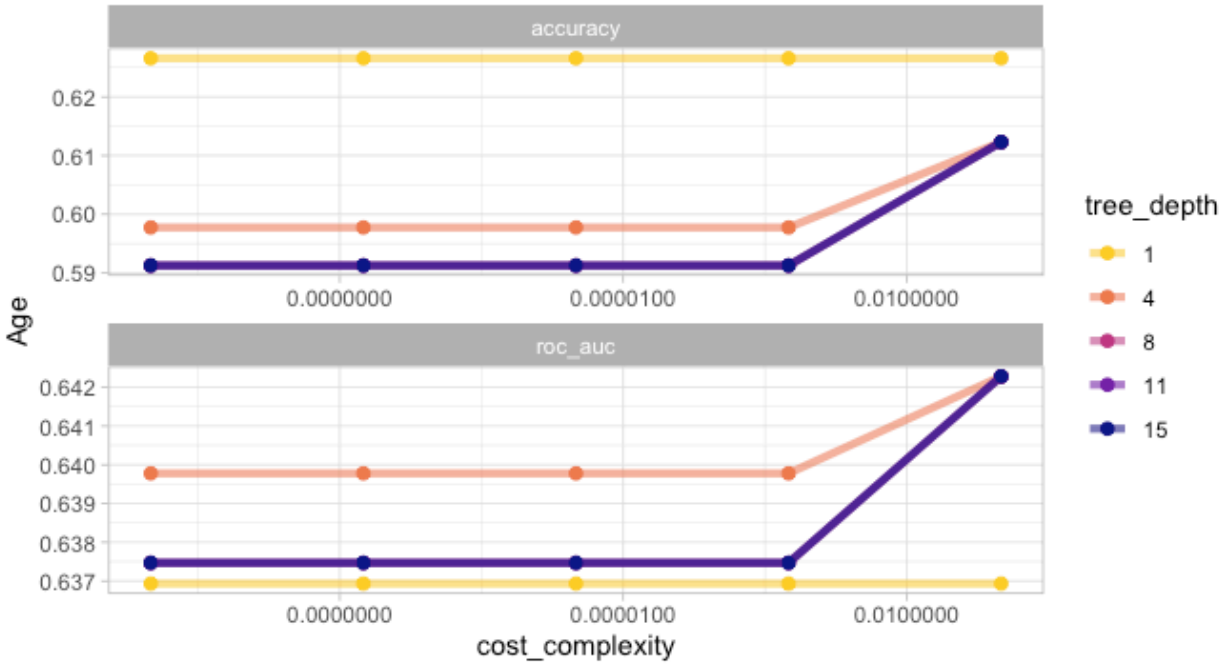
The next section, I will try to do the parameter tuning for the Decision Tree model and will try to improve it by adjusting some of its hyperparameters.

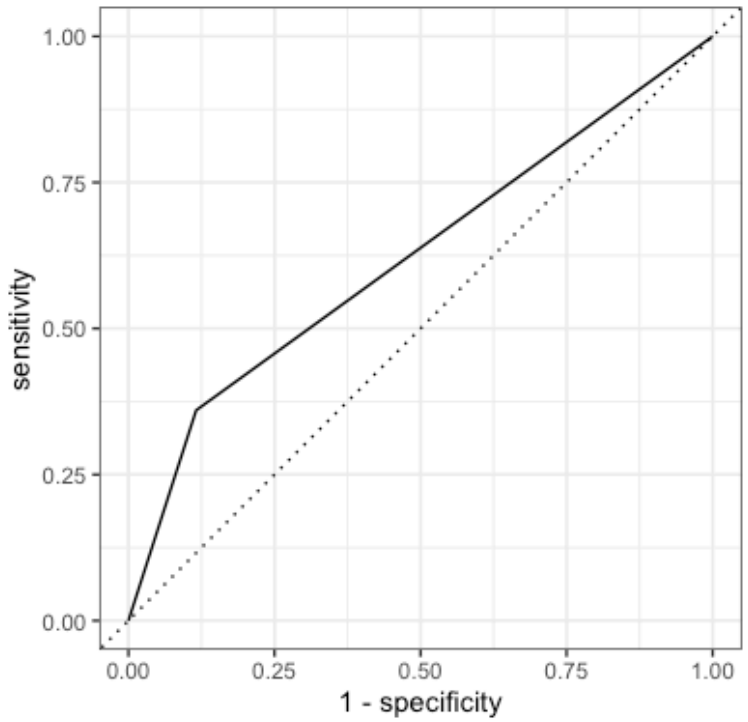
VII. Parameter Tuning

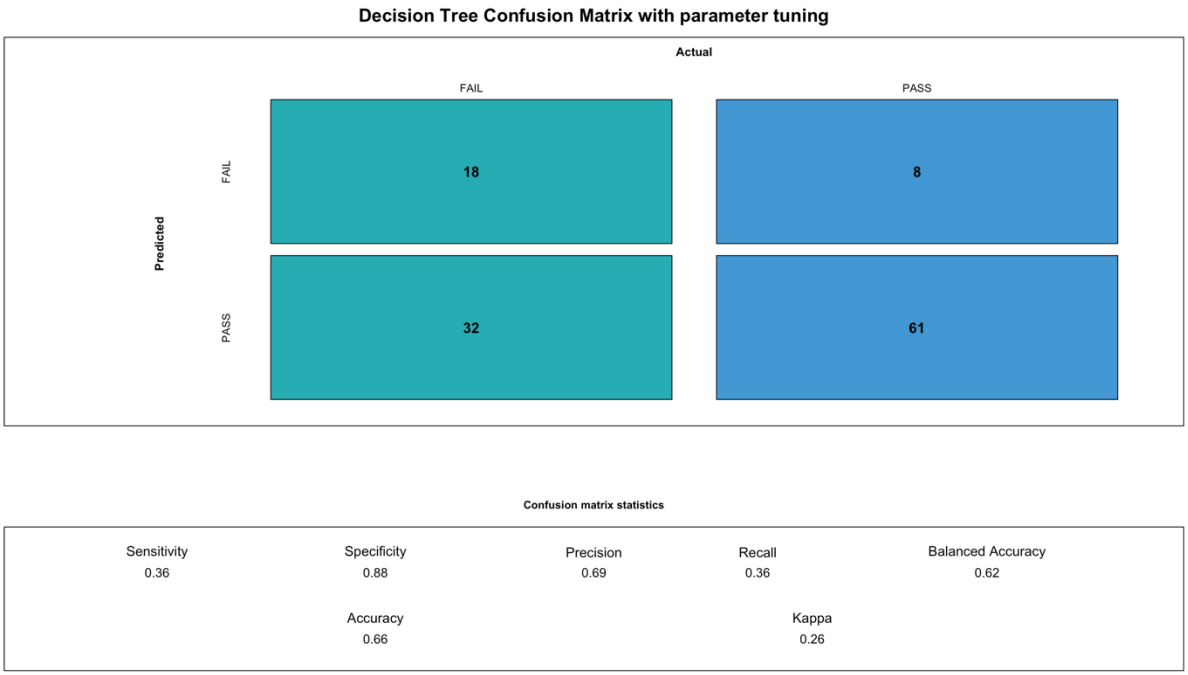
Decision Tree (DT) Hyperparameter tuning. Since DT rely heavily on the trained data, the model tends to perform poorly because it cannot handle a new set of values when deployed. Pruning the decision tree by minimizing the cost complexity will "fix" the overfitting problem or improve the fit of the model. This section will also determine the optimal depth of tree. So, the hyperparameter to be tuned are "cost_complexity" and "tree_depth". The new training and test sets were also split with a ratio of 70:30.

The figure 9 shows the accuracy and ROC_AUC metrics for the 25 model candidates. Depth of 1 for ROC_AUC metric is the worst candidate for "cost_complexity". The best value for "cost_complexity" is 0.0000000001 while the best value for "tree_depth" is 1.

Figure 9. Accuracy and ROC_AUC







Decision Tree	Accuracy	Precision	Recall	F1-Score	AUC	MCC
w/o parameter tuning	63%	56%	60%	58%	0.626	0.2500093
w/ parameter tuning	66%	69%	36%	47%	0.622	0.295972

The model had a 66% in accuracy and an MCC of 0.30 after hyperparameter adjustment, however the AUC was slightly lower. The ROC-AUC value was reduced by 0.004 from the DT without parameter adjustment. Although precision increased by 13%, recall declined. As a result, the F1-score declined by 11% as well. I believe the DT model has improved based on the overall model performance metrics after finding the right “cost_complexity” and “tree_depth”.

VIII. Reflections

I have gained a wealth of knowledge and experience from working on this assignment. From conducting exploratory data analysis to performing feature extraction, I had the opportunity to complete a full cycle of a data science project using the R programming language and apply the concepts I learned throughout the module. Initially, I had planned to use only three models, but as I progressed through the Principles of Data Science module, I discovered the power of other algorithms such as Support Vector Machine (SVM) and Extreme Gradient Boosting (XGBoost) for solving classification problems.

In my opinion, transparency is crucial in any data science project. Real-world projects often undergo peer review to ensure their accuracy and fairness. With this in mind, I designed my algorithms to be reproducible, allowing others to validate and verify the results. Additionally, I came to realize that my own knowledge alone was not sufficient to complete a data science

project successfully. A competent data scientist should also consider previous research and compare their own ideas with well-established projects and articles. I used to believe that model accuracy was the sole determinant of algorithm performance, but this module taught me that a skilled data scientist should not rely solely on accuracy as a metric.

While model evaluation is an important aspect of every data science project, it is essential to be aware of potential analytical biases and ethical issues that can arise. I strongly believe that there is no one-size-fits-all approach to selecting the best model based solely on overall model metrics, as demonstrated in the module. Additionally, I acknowledge that the process of feature selection can introduce bias. Although I employed various feature extraction techniques, I must admit that I did not provide a scientific foundation for selecting the features. Instead, I relied primarily on statistical analysis, which may not always yield perfect recommendations.

Through this assignment, I have not only expanded my technical skills but also gained a deeper understanding of the complexities and considerations involved in data science projects. I am excited to continue my journey as a data scientist, continually learning and improving my practices to ensure the utmost accuracy, fairness, and transparency in my future projects.

Bibliography

Nelder, J. A. & Wedderburn, R. W. M., 1972. Generalized Linear Models. *Journal of the Royal Statistical Society. Series A (General).*

Sharma, A., 2021. *Analytics Vidhya.* [Online]
Available at: https://www.analyticsvidhya.com/blog/2021/02/machine-learning-101-decision-tree-algorithm-for-classification/

Liberman, N., 2017. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/decision-trees-and-random-forests-df0c3123f991

Telrandhe, S. R., Pimpalkar, A. & Kendhe, A., 2016. Detection of brain tumor from MRI images by using segmentation & SVM,. *2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave).*

Brownlee, J., 2016. *Machine Learning Mastery.* [Online]
Available at: https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/

Al-Tashi, Q., Rais, H., Abdulkadir, S. J. & Mirjalili, S., 2020. Feature Selection Based on Grey Wolf Optimizer for Oil & Gas Reservoir Classification. *2020 International Conference on Computational Intelligence (ICCI).*

Kursa, M. B. & Rudnicki, W. R., 2010. Feature Selection with the Boruta Package.. *Journal of Statistical Software, 36(11), 1–13. https://doi.org/10.18637/jss.v036.i11.*

Lin, X., Chao, L., Zhang, Y. & Su, B., 2017. Selecting Feature Subsets Based on SVM-RFE and the Overlapping Ratio with Applications in Bioinformatics.

Brownlee, J., 2020. *Machine Learning Mastery.* [Online]
Available at: https://machinelearningmastery.com/confusion-matrix-machine-learning/

Chan, C., 2022. *DisplayR Blog.* [Online]
Available at: https://www.displayr.com/what-is-a-roc-curve-how-to-interpret-it/