Student Performance

Introduction

Education institutes commonly use exam metrics including pass/failure rates to measure organisational performance. The associated data is often properly collected and stored making it good candidate for data analysis and the use of machine learning practices. Paulo Cortez and Alice Silva from University of Minho conducted a study in 2008 to investigate how machine learning may be used to predict student performance. The data used in this study has been made available on the UCI Machine Learning Repository, and is the main data source for this report. Student performance predictions can be used to inform the education process and in turn improve final outcomes. For example if students that are at a high risk of failing can be identified early, then remedial processes can be put in place to try and improve overall outcomes. For this process to work effectively educators need to have accurate predictions of students final outcomes as early as possible.

This reports seeks to described how machine learning practices can be used to predict students performance. In order to allow educators to take early remedial action the study focuses on using survey data collected before the start of a course. As the model output is intended to inform educators it is important to consider what information might be needed or required. By predicting the students final grade, educators can be informed of both the likely outcome in terms of pass or fail as well as the magnitude of the deficit in performance. Such a model will allow teachers to better understand the requirement for remedial action and therefore take steps to improve overall performance.

The findings of this report show that an ensemble machine learning model can make useful predictions for student outcomes. The final predictions have Root Means Square Error of 2.26 when predicting a score from 1-20. The report concludes that this level of accuracy is sufficient to make the predictions of use to educators. It is recommended that future studies should investigate the utility and accuracy of classification models to the predicts whether or not a student will pass or fail the subject

Project aim - Develop a machine learning model that accurately predicts a students final grade.

Method and Analysis

Data Exploration

The selected data set contains data from two subjects, Mathematics and Portuguese. This project aims to develop a model that will inform the education process. In contrast to Cortez and Silva this study focuses on developing models for individual subjects. It is assumed that this approach will better serve the aims of the project by providing predictions that can be used to inform specific remedial action. The Portuguese data set was selected as it contains more observations and the date is of equal quality to the maths data set. This should allow for the development of a more accurate model.

To start to understand the data I began by referring to the data dictionary, which can be found here. From initial observation it is clear that the majority of the data has been collected using questions in student surveys. The only exception to this is the exam scores (G1,G2,G3)

Before further exploration I first divided the port data frame into a training and test set. There are 649 observations in the port dataset in order to maximise the available training data the set was split 90%/10% training and test. A cross validation method was selected for model evaluation allowing all most all training data to be used to for model training and evaluation. With 63 observations the test set is similar to the number of students in a class or intake. By applying the final model to the test set we can gain an appreciation of how the model may perform with real world data.

In this report the training data set was used for data exploration purposes. This approach allowed for preprocessing and model design decisions to be informed by a large portion of the data, whilst retaining unseen data or "hold out data" for final testing.

Looking at the quality of the data, the code below shows us that the data is complete with no NA values.

```
df_port[rowSums(is.na(df_port)) > 0,]
```

```
##  [1] school     sex        age        address    famsize    Pstatus
##  [7] Medu       Fedu       Mjob       Fjob       reason     guardian
## [13] traveltime studytime  failures   schoolsup  famsup     paid
## [19] activities nursery    higher     internet   romantic   famrel
## [25] freetime   goout      Dalc       Walc       health     absences
## [31] G1         G2         G3
## <0 rows> (or 0-length row.names)
```

Additionally we can see from the table below that a significant number of variables are "character" class.The data dictionary also tells us that the "integer" data is mostly categorical.

	Class
school	character
sex	character
age	integer
address	character
famsize	character
Pstatus	character
Medu	integer
Fedu	integer
Mjob	character
Fjob	character
reason	character
guardian	character
traveltime	integer
studytime	integer
failures	integer
schoolsup	character
famsup	character
paid	character
activities	character
nursery	character
higher	character
internet	character
romantic	character
famrel	integer
freetime	integer
goout	integer
Dalc	integer
Walc	integer
health	integer
absences	integer
G1	integer
G2	integer
G3	integer

Taking a closer look at each of the predictors we can see that some are significantly imbalanced. The table below shows the predictors that have more than 80% of responses in one category.

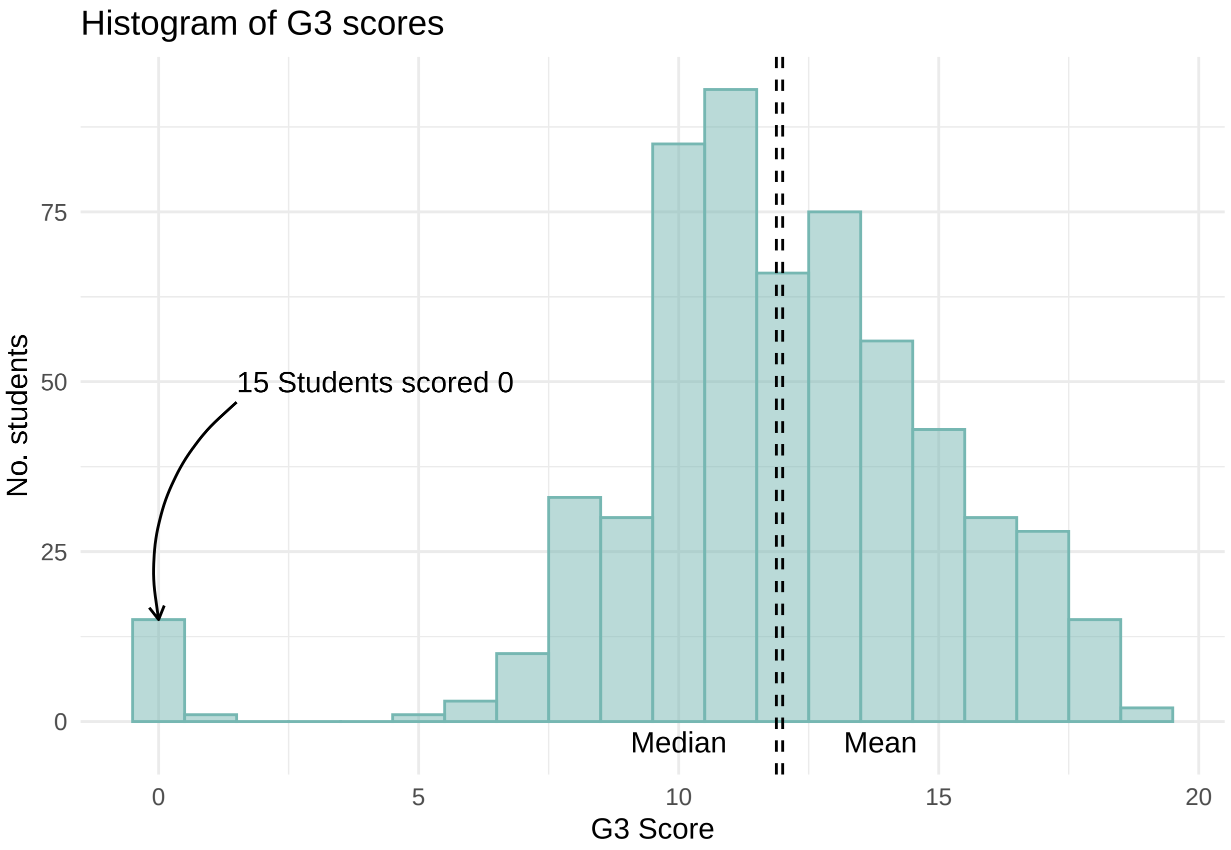
Characteristic	**N = 586**
Pstatus	
A	72 (12%)
T	514 (88%)
failures	
0	494 (84%)
1	64 (11%)
2	15 (2.6%)
3	13 (2.2%)
schoolsup	61 (10%)
paid	35 (6.0%)
nursery	467 (80%)
higher	523 (89%)

Exploring the data further we can see that Mjob and Fjob have a significant portion of responses in the 'other' category. These predictors are not informative as most responses are in a very broad category.

Characteristic	**N = 586**
Mjob	
at_home	123 (21%)
health	40 (6.8%)
other	237 (40%)
services	120 (20%)
teacher	66 (11%)
Fjob	
at_home	40 (6.8%)
health	19 (3.2%)
other	329 (56%)
services	167 (28%)
teacher	31 (5.3%)

The aim of this project is to accurately predict final exam scores (G3) for any given student. Looking through the G3 column we can see that all values are integers, it is unclear if this is a discrete or continuous variable. For the purpose of this study exam scores are regarded as continuous variables and regression algorithms are applied in order to predict scores. This report uses the Root Mean Square Error to optimize the models and therefore describes the loss in the same units as the exam scores.

Exploring the distribution of G3 within the training data set we can observe some interesting characteristics. The plot below highlights that out of 586 students 15 scored 0. The remainder of the the G3 results approximate a normal distribution.



Mean	Median	SD
11.87884	12	3.311311

The data dictionary does not specify if a score of 0 is due to the student no taking the test or if they were unable to answer any questions correctly. To try and better understand how a student comes to score 0 we can look at there scores in other tests. The data frame below shows all of the records that contain a 0 score in anyone of the 3 tests.

G1	G2	G3
4	0	0
5	0	0
5	0	0
7	0	0
7	0	0
8	0	0
9	0	0
7	5	0
8	6	0
7	7	0
7	7	0
8	7	0
5	8	0
8	8	0
11	9	0

We can see from the exam results that no students scored 0 in the first test (G1), and that students that scored 0 in the second test (G2) also scored 0 in the third test (G3). This pattern of results suggests that students are leaving the course at some point in the program and therefore not taking any further exams. This assumption is further supported by the observation that only 1 student scored below 5 at the G3 examination.

Preprocessing

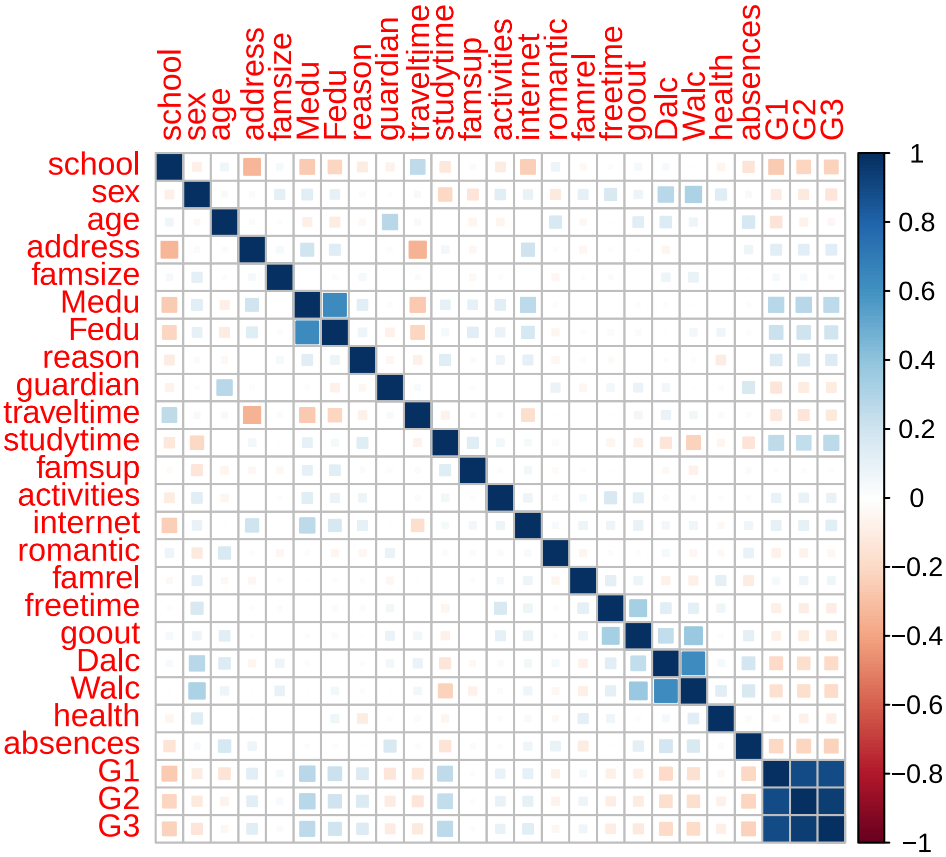
In order to ensure the best accuracy for each machine learning model the data needs to be processed before being passed to an algorithm. The pre processing for this project followed a number of steps all of which are outlined below.

1. Outliers were removed from the G3 category. During data exploration we saw that some students score 0 and 1 student scored 1. These scores are significantly different for the other data. Some reasons for this might include, student did not attend exam or a mistake was made when inputting data. During preprocessing all scores less than 1 were removed from the data.

2. Remove predictors with very few non-unique values or close to zero variation. Data exploration revealed several predictors that had more than 80% of responses in a single category and 1 predictor that contained information that was non-specific. The following predictors were removed from the data due to the reasons above; Pstatus, failures, schoolsup, paid, nursery, higher, Mjob,Fjob.

3. All data points were transformed into numeric values this allows several different types of models to be used during the predictions phase of the project.

4. Predictors that have a strong correlation were removed or combined, the following text provides more detail. The grid below shows the correlation of each of the predictors using the Pearson's correlation coefficient.



There is a correlation coefficient of 0.63 Mother (Medu) and Father (Fedu) education levels. Intuitively we would expect parent education to be a good predictor of a students academic performance, so we want to preserve the predictor. We will do this by calculating the mean of Medu and Fedu and drop both variables,creating a new predictor Parents Education Pedu. The code snippet below shows how this was calculated.

```
ts_p4 %>% mutate(Pedu = (Medu+Fedu)/2) %>% select(Pedu) %>% head() %>% kable()
```

Pedu
1.0
1.0
3.0
3.0
3.5
2.0

Daily and weekly alcohol intake share a similar relationship to the education predictors above, correlation coefficient 0.63. Therefore a identical process was followed generating a mean for alcohol intake called Average Alcohol (Aalc) and then removing the original two predictors from the data frame.

Aalc
1.0
2.5
1.0
1.5
1.5
1.0

The ‘Go out’ predictor is correlated with two other predictors (freetime and Aalc) this maybe because students with free time are going out and drinking. It is assumed that this information is captured in other predictors, therefore "goout’ was removed during preprocessing.

There is also a correlation between school, travel time and address. “School” is correlated with G3 and is assumed to be a good predictor, therefore travel time and address were removed. The aim of this project is to make accurate exam score predictions without the use of previous exam results, therefore during preprocessing G1 and G2 we removed. At this stage the data was normalized for use in multiple regression model and standardized for use elsewhere in the model development phase.

Modeling Approaches

During this project 8 machine learning models were fitted, tested and compared.The models were evaluated using their Root Mean Square Error (RMSE). The following paragraphs describe how each model was fitted, with a final paragraph that compares the results for each model.

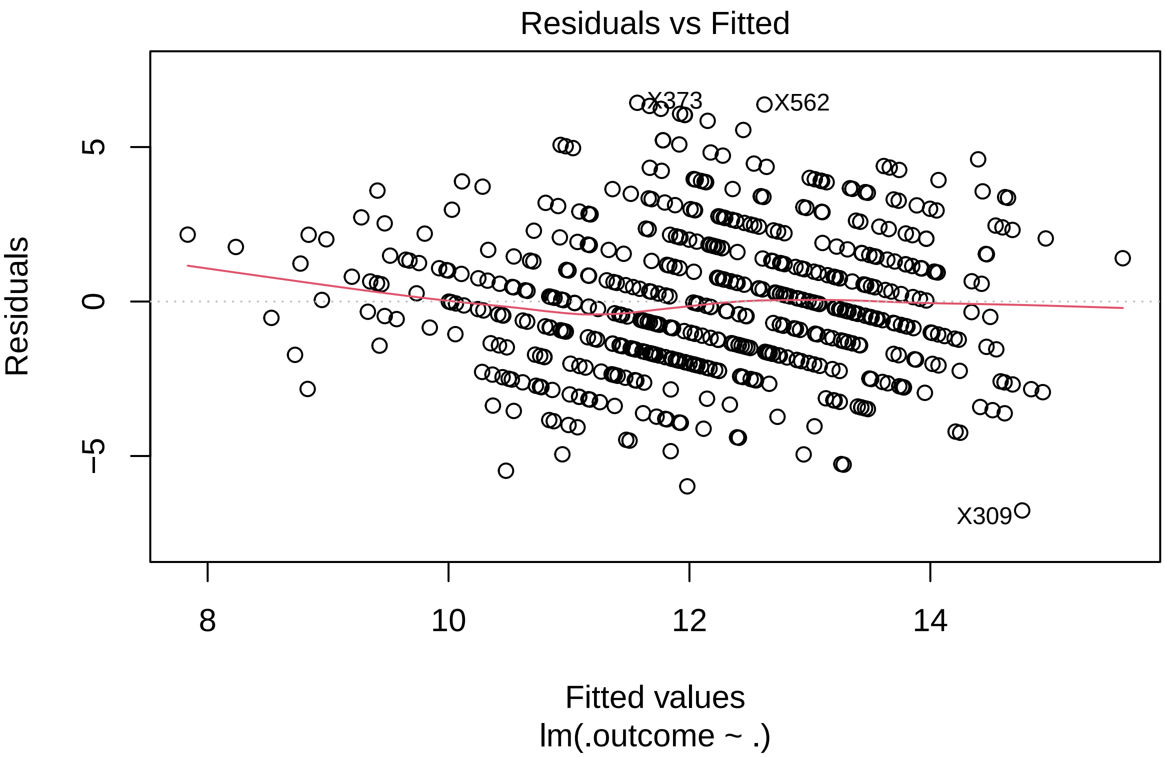
Naive model In order to provide a benchmark for model development I started by predicting the G3 exam score simply using the G3 mean value. This naive approach produced a RMSE of 3.73.

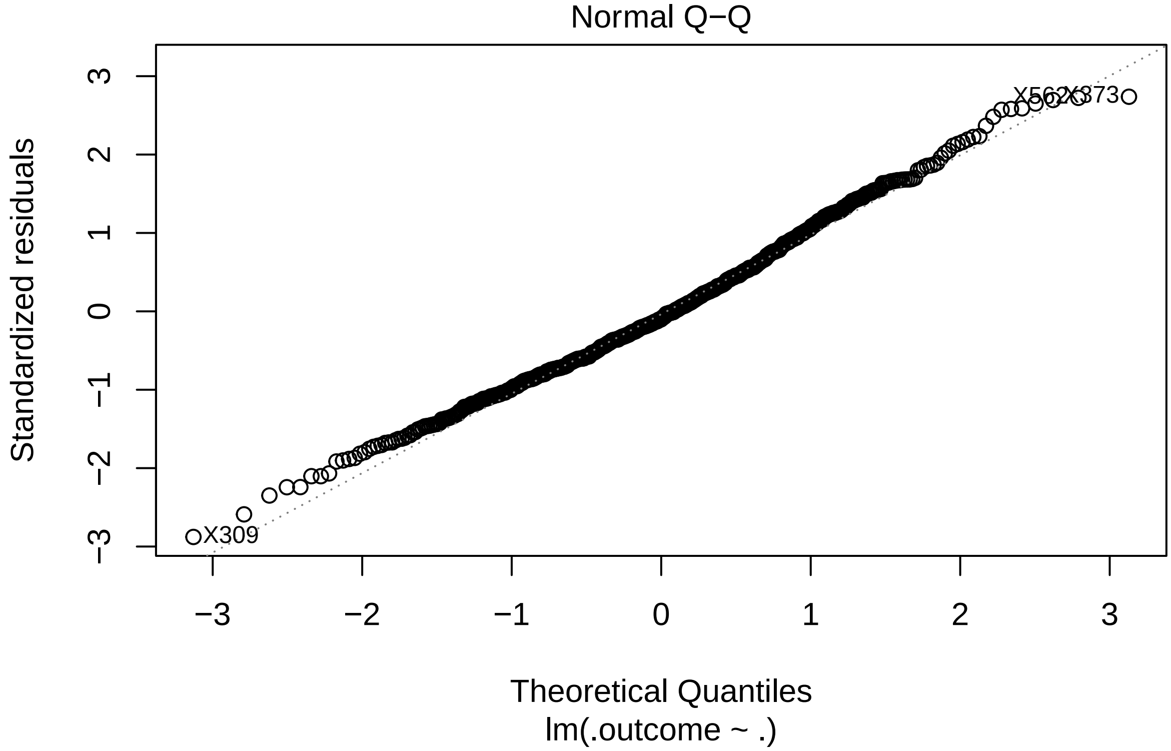
```
  # Use the mean training_set G3 score as the prediction for all G3 scores.
  naive_pred <- mean(ts_p0$G3)
  # Calculate RMSE for this model
  naive_RMSE <- RMSE(test_set$G3,naive_pred)
```

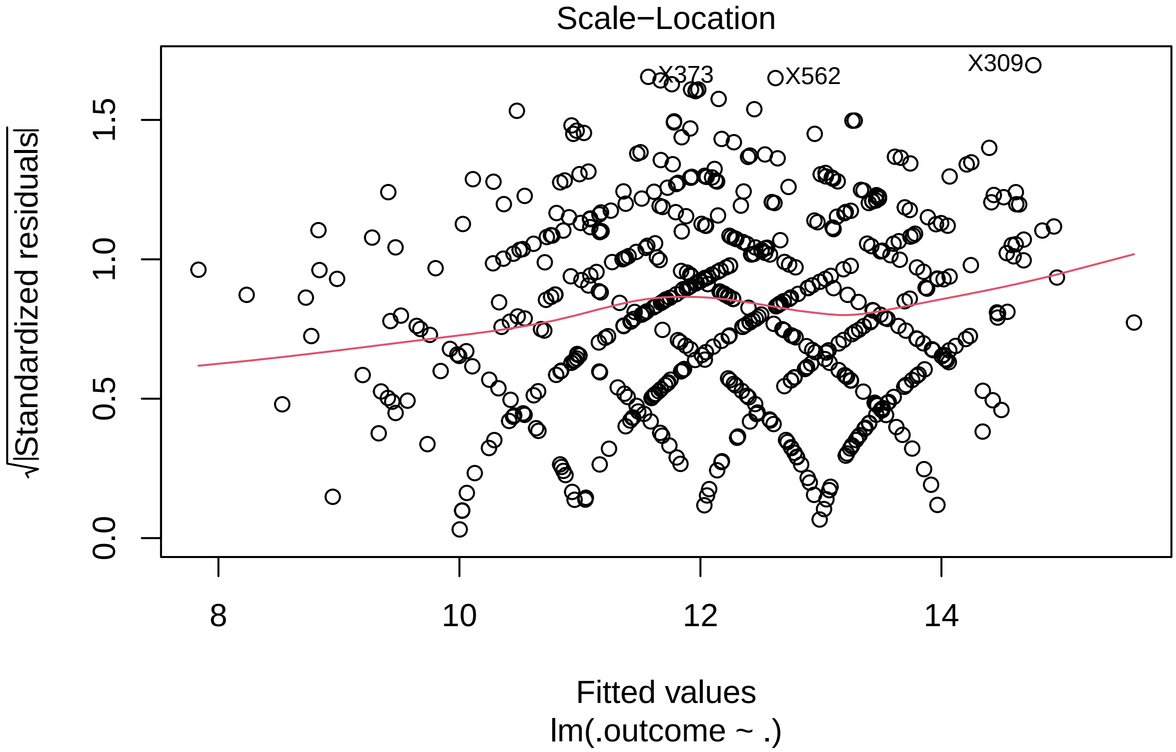
Multivariate Linear Regression The first model developed was a Multivariate Linear Regression model. Using the caret package, cross validation and all of the available predictors we get an RMSE of 2.40 and the following summary.

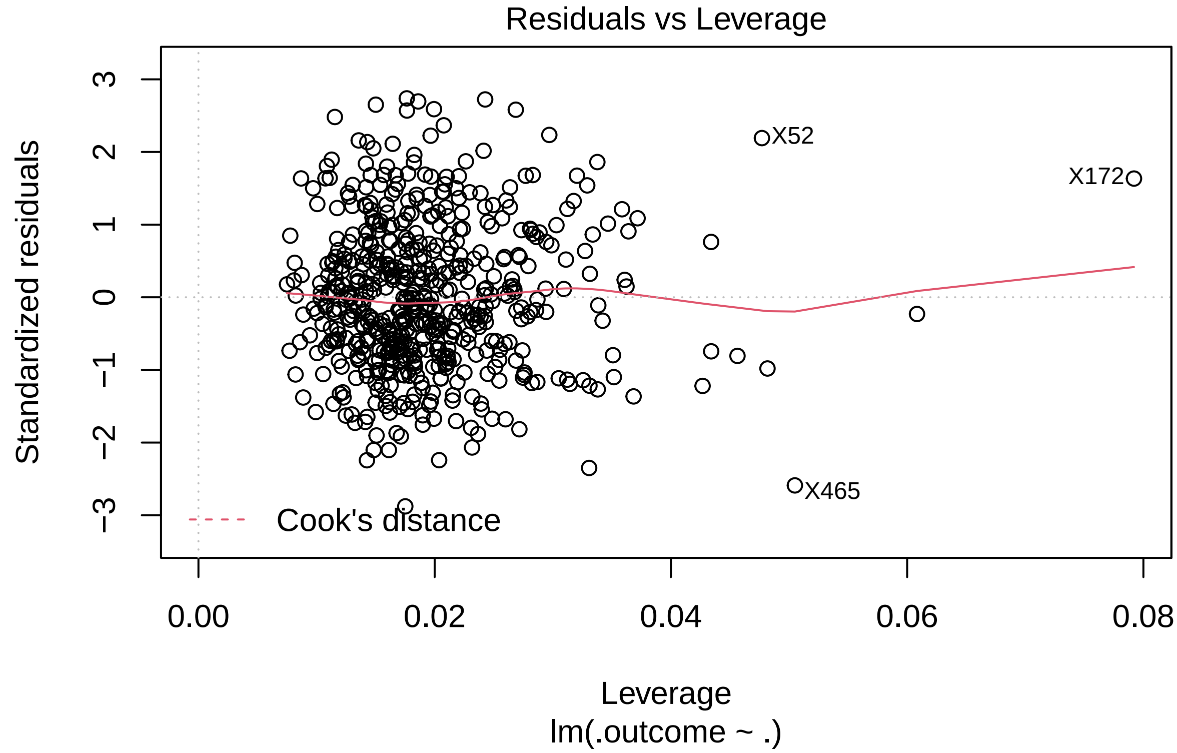
```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.0650 -1.6158 -0.1186  1.4960  6.8003
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.8112     0.5775  20.451  < 2e-16 ***
## school       -0.9850     0.2300  -4.283 2.17e-05 ***
## sex          -0.6912     0.2277  -3.036 0.002513 **
## age           0.8616     0.6211   1.387 0.165934
## famsize       0.3085     0.2171   1.421 0.155879
## reason        0.4529     0.2581   1.755 0.079845 .
## guardian     -0.7931     0.4002  -1.982 0.047995 *
## studytime     1.3795     0.3804   3.626 0.000314 ***
## famsup       -0.4150     0.2096  -1.980 0.048215 *
## activities    0.3168     0.2049   1.546 0.122609
## internet      0.3072     0.2495   1.232 0.218627
## romantic     -0.1475     0.2134  -0.691 0.489715
## famrel        0.4955     0.4343   1.141 0.254375
## freetime     -0.6083     0.3889  -1.564 0.118300
## health       -0.5228     0.2803  -1.866 0.062634 .
## absences     -3.6258     0.7403  -4.898 1.27e-06 ***
## Pedu          2.1542     0.4241   5.079 5.19e-07 ***
## Aalc         -1.0447     0.4442  -2.352 0.019013 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.362 on 552 degrees of freedom
## Multiple R-squared:  0.252,  Adjusted R-squared:  0.2289
## F-statistic: 10.94 on 17 and 552 DF,  p-value: < 2.2e-16
```

We can see from the summary of coefficients that there are a number of insignificant predictors which I subsequently removed for the next iteration of the model. This reduced the RMSE to 2.39. The plots below were used to further evaluate and refine the model.


Residuals vs Fitted
X373
X562
X309
Residuals
5
0
−5
8
10
12
14
Fitted values
lm(.outcome ~ .)


Normal Q−Q
X562
X373
X309
Standardized residuals
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(.outcome ~ .)

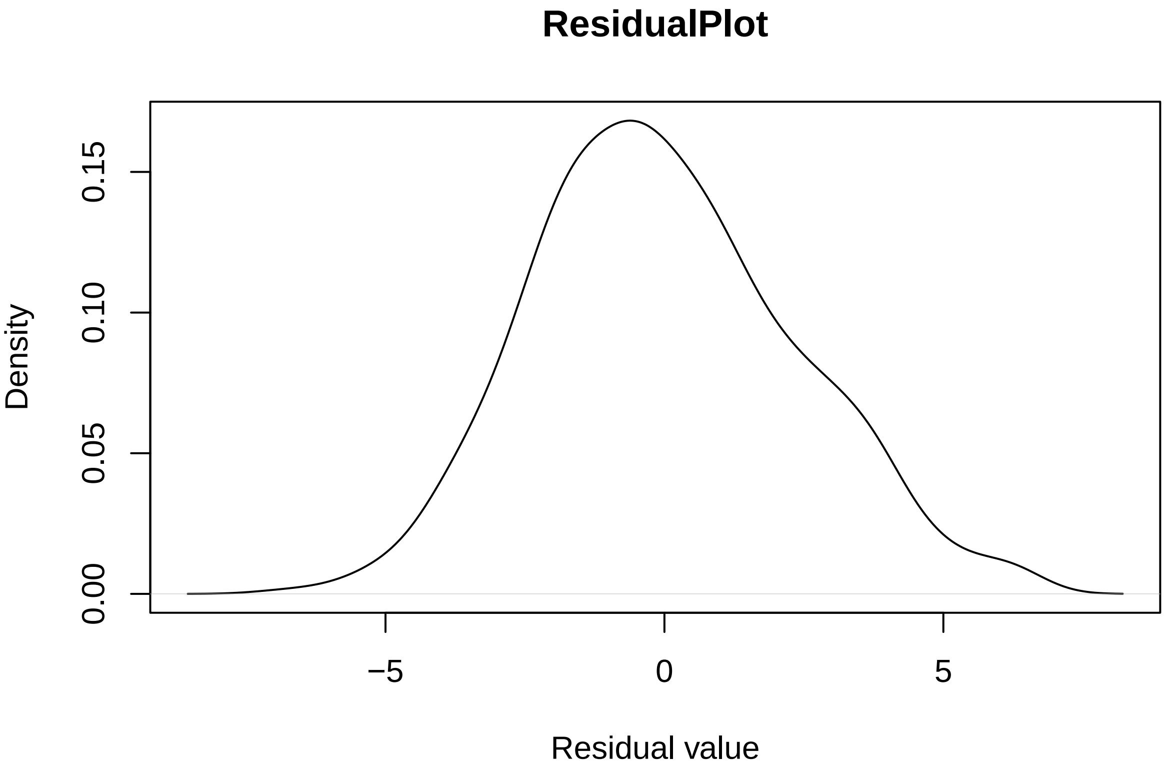




The following observation were made from the plots above.

Residuals vs fit plot The residuals are partially randomly distributed around 0. The relationship appears to be close to linear. The data is not equally spread across the score range so there is some uncertainty. 3 outliers identified 309,373,562

Normal Q-Q The residuals approximate a normal distribution however there is some departure around the tails. A density plot of the residuals shown below shows that there is a slight negative skew.



Residuals Vs Leverage No observations are outside cook's distance.

Based on the above observations I took the following steps to improve the model. Remove the 3 extreme values. Correct with skew by using a log transformation on G3. The following code shows how this was done and the subsequent reduction in RMSE(2.31).

```
# Eliminate extreme values
  mul_reg_pp <- mul_reg_pp[-which(rownames(mul_reg_pp)%in% c("562","373","309")),]
# Correct with skew with log
  set.seed(2009)
  mul_reg_fit_log <- train(log10(G3)~school+sex+studytime+
                             reason+guardian+famsup+Pedu+
                             Aalc+health+absences,
                           data = mul_reg_pp,method="lm",
                           trControl = trainControl(method = "repeatedcv",
                                                    number = 10,repeats = 3))
# Inverse log to get RMSE
  skew_adu_results<-RMSE(mul_reg_pp$G3,10^mul_reg_fit_log$finalModel$fitted.values)
  skew_adu_results
```

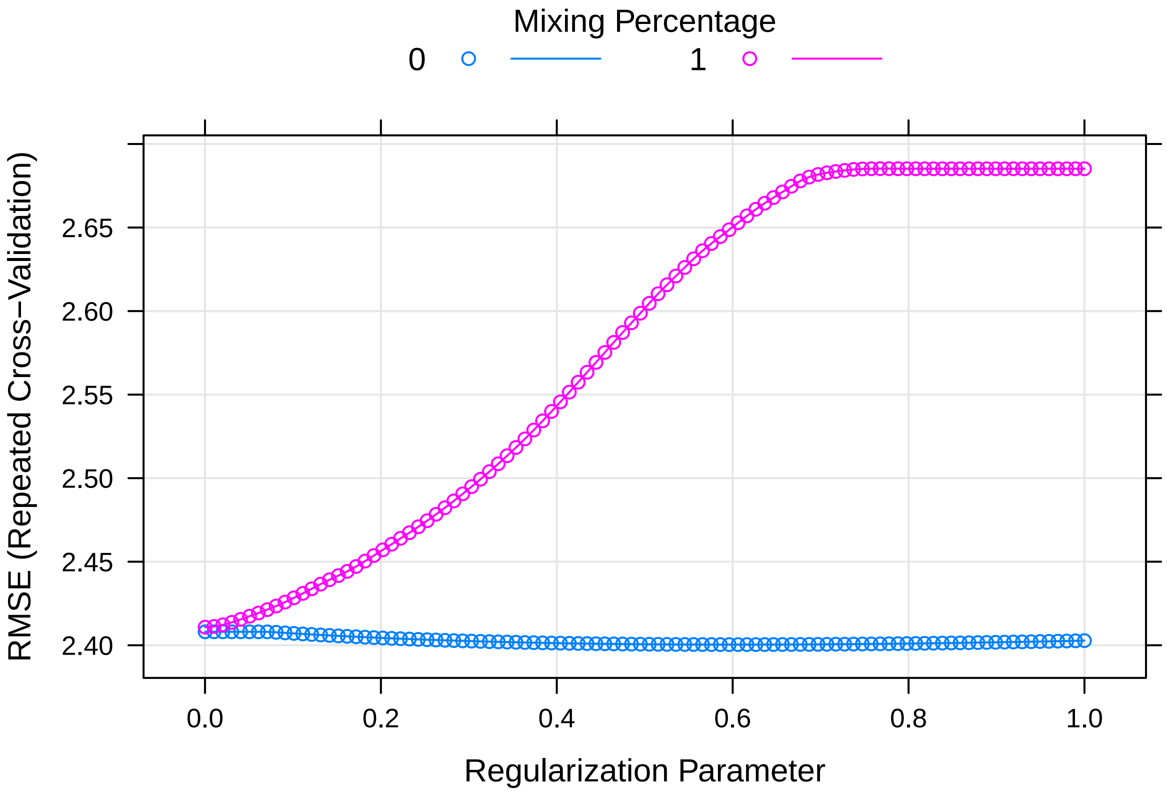
```
## [1] 2.314278
```

Although it is possible to further refine this model by repeating the steps above, this could cause the model to be over fitted to the training data.

Penalized Linear Regression Rather than removing predictors we can use a model that penalizes those predictors that contibute less. By using the caret package and the glmnet (ridge&lasso) algorithm we can automate some of the processes above. The following code shows how this algorithm was optimized for the

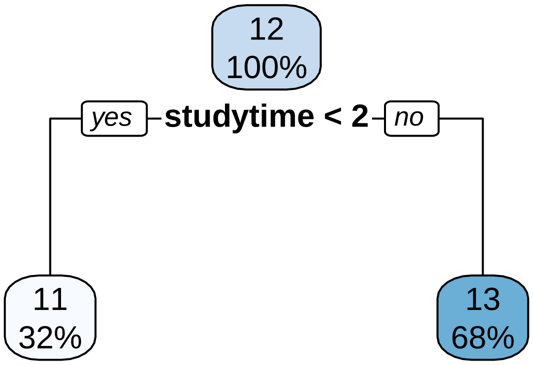
Lamda parameter. Additionally glmnet allows us to choose how the penalty is applied, ridge or lasso. We can see from the plot of the fit that the lowest RMSE is achieved with ridge and lamda around 0.6.

```
# Tune using grid method
lambda <- seq(0.0001, 1, length = 100)
glmnet_fit_tune = train(G3 ~ .,
                  data=stan_pp,
                  method="glmnet",
                  metric = "RMSE",
                  trControl = trainControl(method="repeatedcv", number=10, repeats=3),
                  # alpha = 0:1 try lasso (1) and ridge (0)
                  tuneGrid = expand.grid(alpha = 0:1,lambda = lambda)
                  )
plot(glmnet_fit_tune)
```



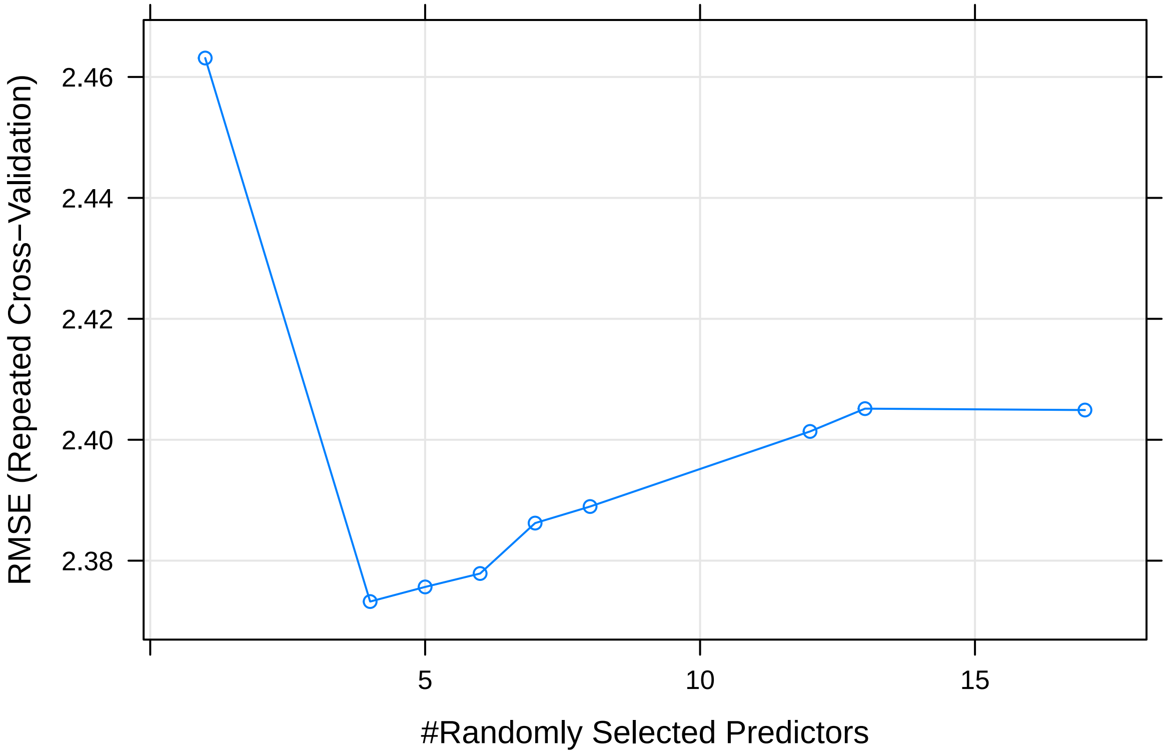
Once this model was tuned its produced and RMSE of 2.4.

Regression Tree Regression tree models are a fast and easy to understand model that can provide accurate results. This model does not require normalized data and the fit can be easily plotted.

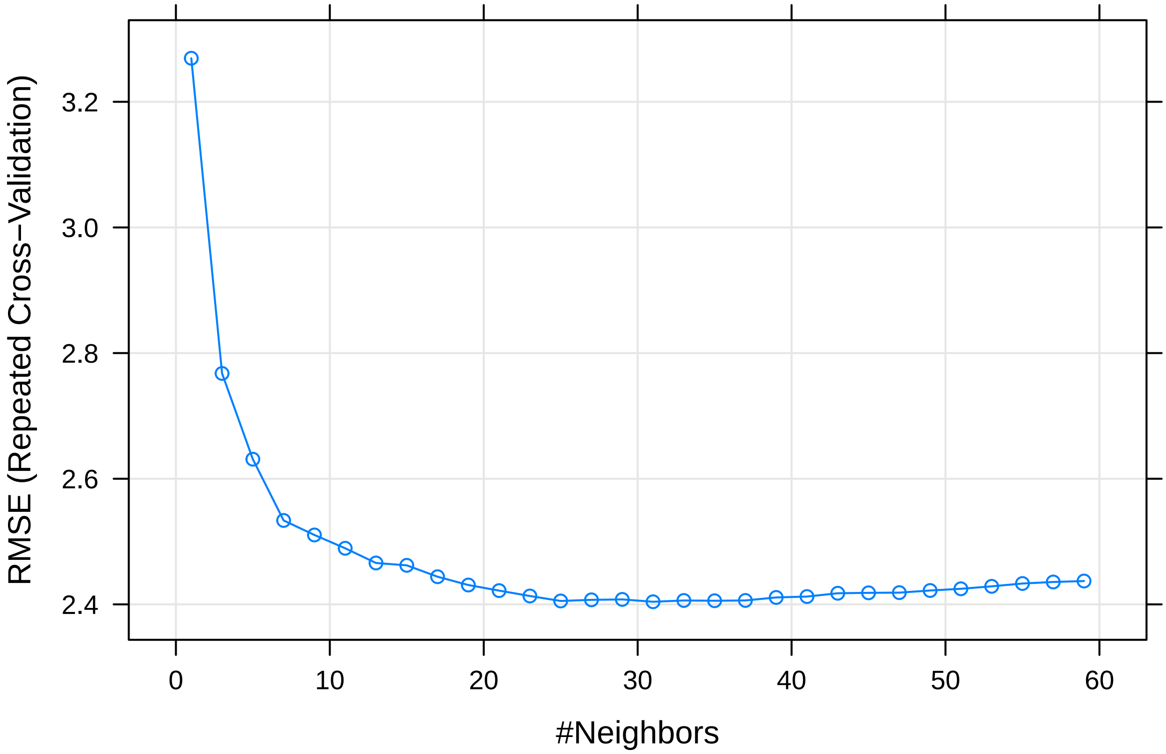


Although the model can be easily understood from the plot it produces a relatively high RMSE (2.6) when compared to our previous models.

Random Forest A random forest algorithm was trained using the caret package and optimized for the mtry parameter. We can see from the plot below that the optimal value for mtry is 4. This produces a RMSE 2.37, a significant improvement on the regression tree.

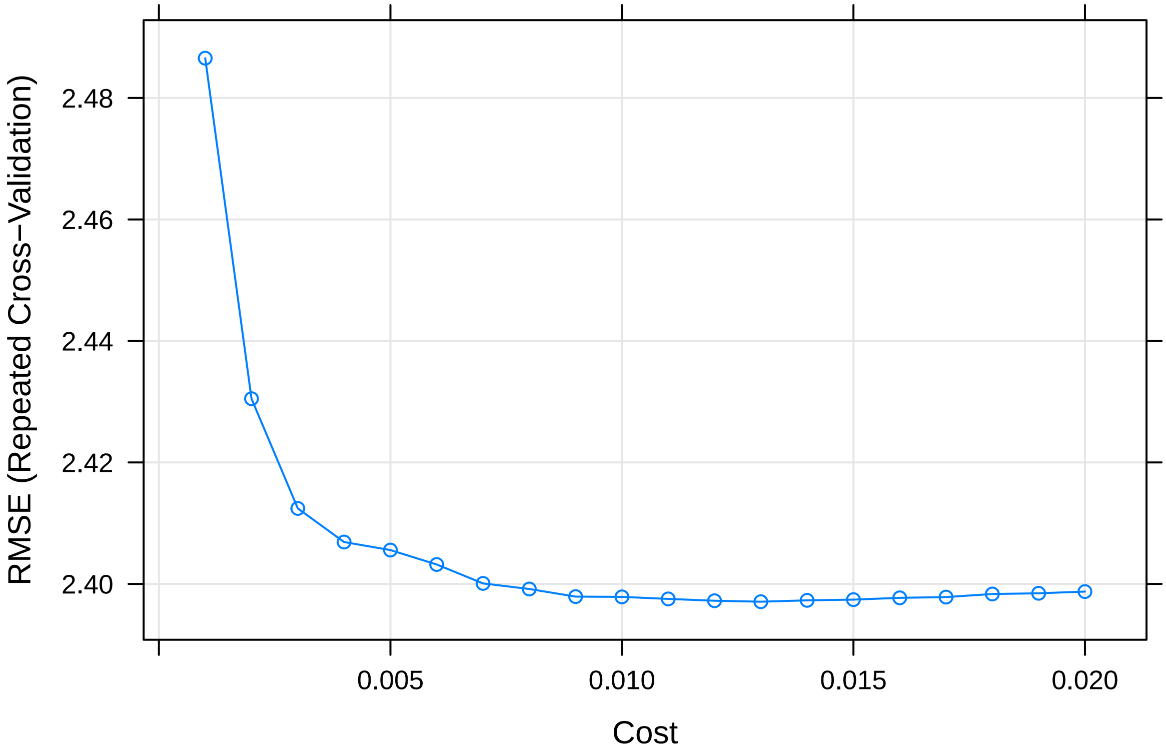


KNN The model was trained using the standardized data and tuned to optimize K. The code and tuning plot are show below.



The final model has a RMSE 2.40, a lesser performance than random forest and multivariate regression model.

SVM The SVM linear algorithm was trained with standardized data, the model was tuned for the cost parameter.



The optimal cost parameter was 0.016 and the final RMSE 2.40. The table below compares each of of the models in terms of RMSE.

Model	RMSE
Multivariate Regression	2.314278
Naive	2.338707
Random Forest	2.373240
SVM	2.397073
Penalized Linear Regression	2.400395
KNN	2.404171
Regression Tree	2.643254

We can see that Multivariate Regression has a significantly lower RMSE. The next 4 lowest RMSE models are closely grouped. The following paragraph focuses on the construction of an ensemble predicative model in an attempt to further improve model accuracy.

Ensemble An ensemble model was created using the caretEnsemble package. The advantage of this package is that an ensemble algorithm can be quickly trained using the caret list and stack functions. The disadvantage is that each model cannot be individually tuned and that the multivariate regression model shown early cannot be included due to the modifications that have been made outside of the model. To construct the ensemble I have selected 5 of the top preforming models.

```r
set.seed(2017)
model_list <- caretList(G3 ~ .,
    data=ts_p9,
    trControl = trainControl(method="repeatedcv", number=10, repeats=3,
                             savePredictions = "final"),
    methodList = c("lm","glmnet", "rf", "knn","svmLinear"),
    tuneList = NULL,
    continue_on_fail = FALSE)
set.seed(2017)
  # Create stack using rf
rf_ensemble <- caretStack(
    model_list,
    method="rf",
    metric="RMSE",
    trControl=trainControl(
      method="repeatedcv",
      number=10,
      repeats=3,
      savePredictions="final"
    )
  )
min(rf_ensemble$error$RMSE)
```

```
## [1] 2.246057
```

The ensemble model produces a minimum RMSE of 2.25

The final comparison table including the ensemble model is included below.

Model	RMSE
Ensemble	2.246057
Multivariate Regression	2.314278
Naive	2.338707
Random Forest	2.373240
SVM	2.397073

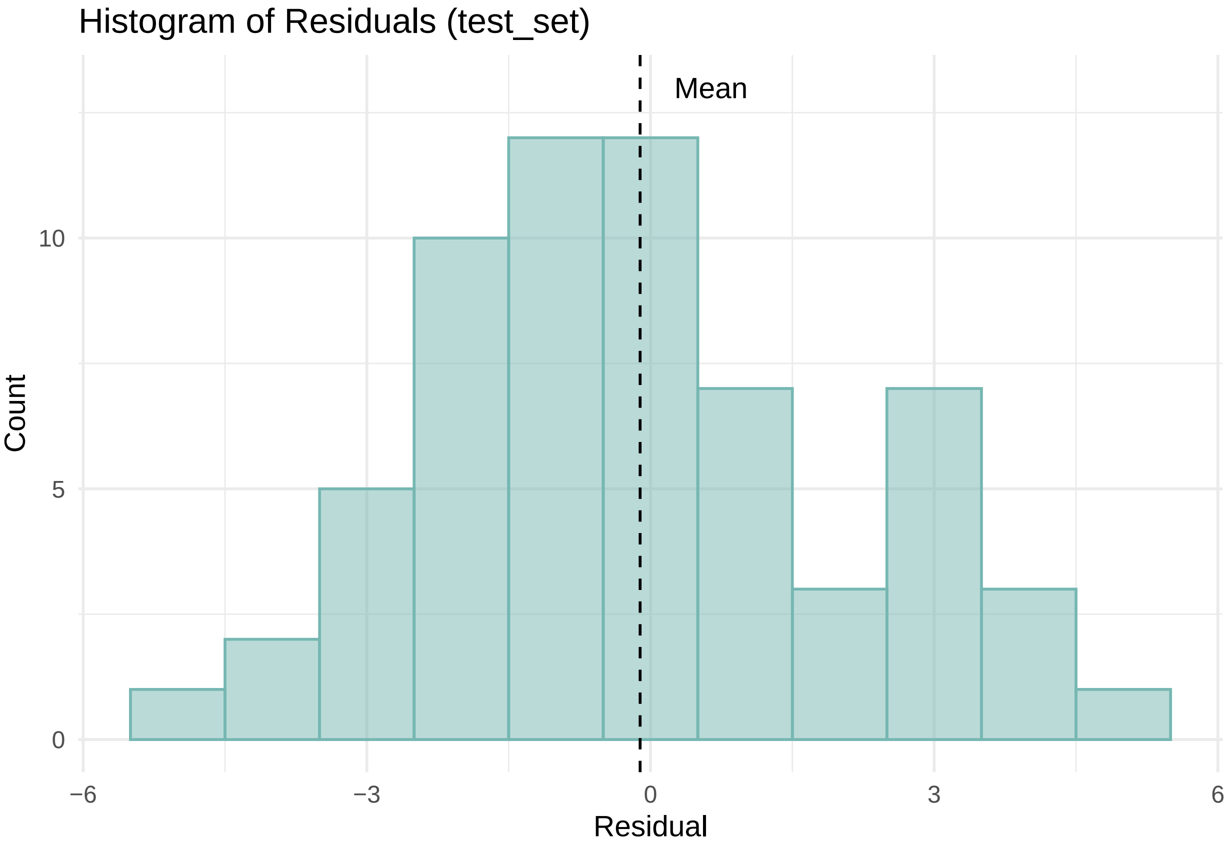
Model	RMSE
Penalized Linear Regression	2.400395
KNN	2.404171
Regression Tree	2.643254

The ensemble model has the lowest RMSE, this is closely followed by the Multivariate model. The following paragraph discusses the prediction results from the Ensemble model using the test data.

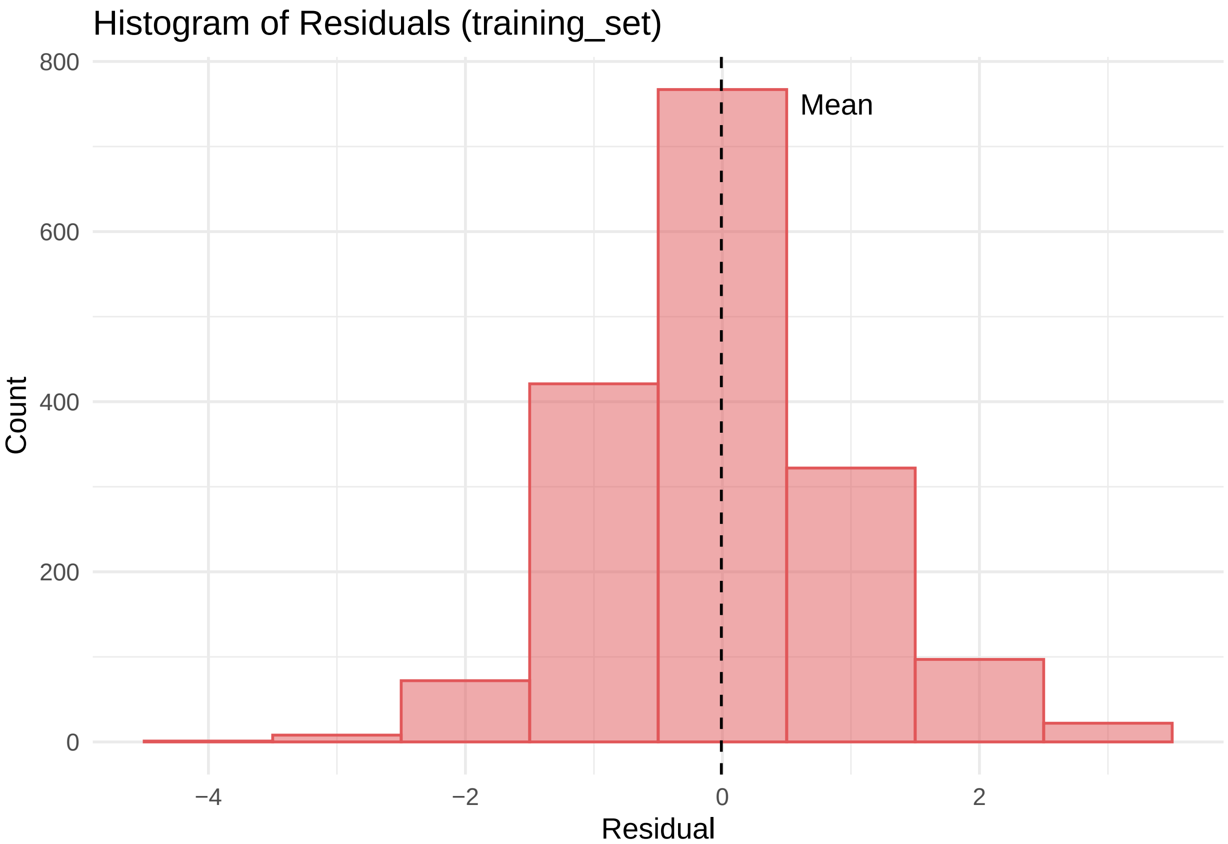
Results

If we treat the test data set as if it were a new class of 63 students our aim is to predict their final grades. When the model is used to predict scores using the test data set we achieve a RMSE - 2.26. This is slightly higher than RMSE calculated during model training, a change in RMSE is to be expected and this change is not statistically significant.

Looking closely at the predictions and the residuals we can see that there is a slight negative bias. The plot below shows a histogram of the residuals, the mean is below 0 at -0.11. This means that the model is on average predicted scores that are slightly higher than the observed scores.



The reason for this over prediction is unclear. It is possible that the test data has an inherent pattern that has not be captured during model training. It is also possible that there is a problem with the model that has not been identified during training. To investigate if there is a previously unidentified problem we can look at the residual distribution for training date, plotted below.



There is no bias in these residuals so it is assumed the the bias above is due to a sample error.

Definitively determining whether or not this model is sufficiently precise requires a subject matter expert, in this case a teacher. We can however assume that the closer the predictions are to the final score the more useful they will be to the teacher. The maximum error is 5.19 considering that there are 20 points available of the exam this is a significant error. Despite this several predictions are close to the actual score. For the test data set the model predicted 78% of scores within 3 points of the actual G3. It is clear from the results that the models predictions cannot be totally relied upon however in most cases they are a good indication of a students final exam score.

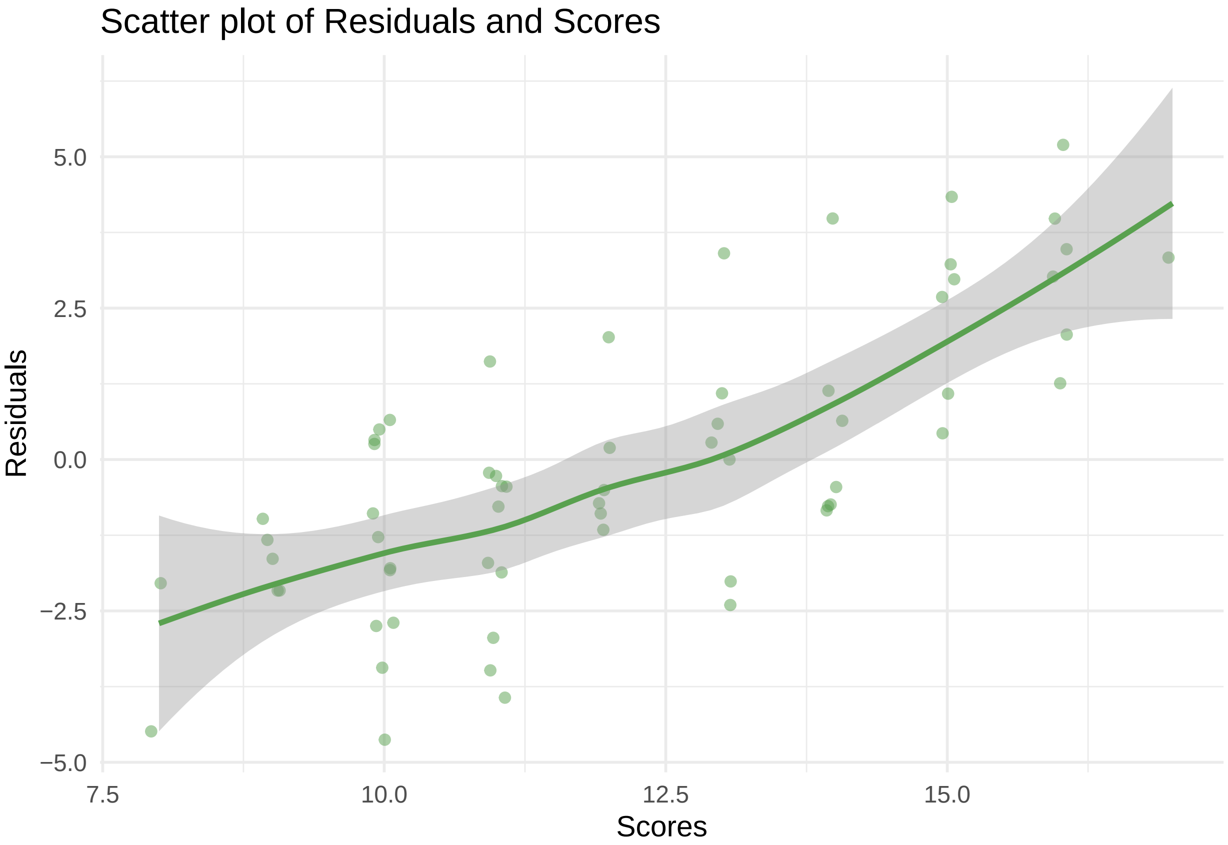
We can see from the residuals histogram that there are some predictions that are significantly different from the actual score. The table below shows predictions that have residuals greater than an absolute value of 4.

```
results %>% filter(abs(residual)>3) %>% arrange(by=residual) %>% kable()
```

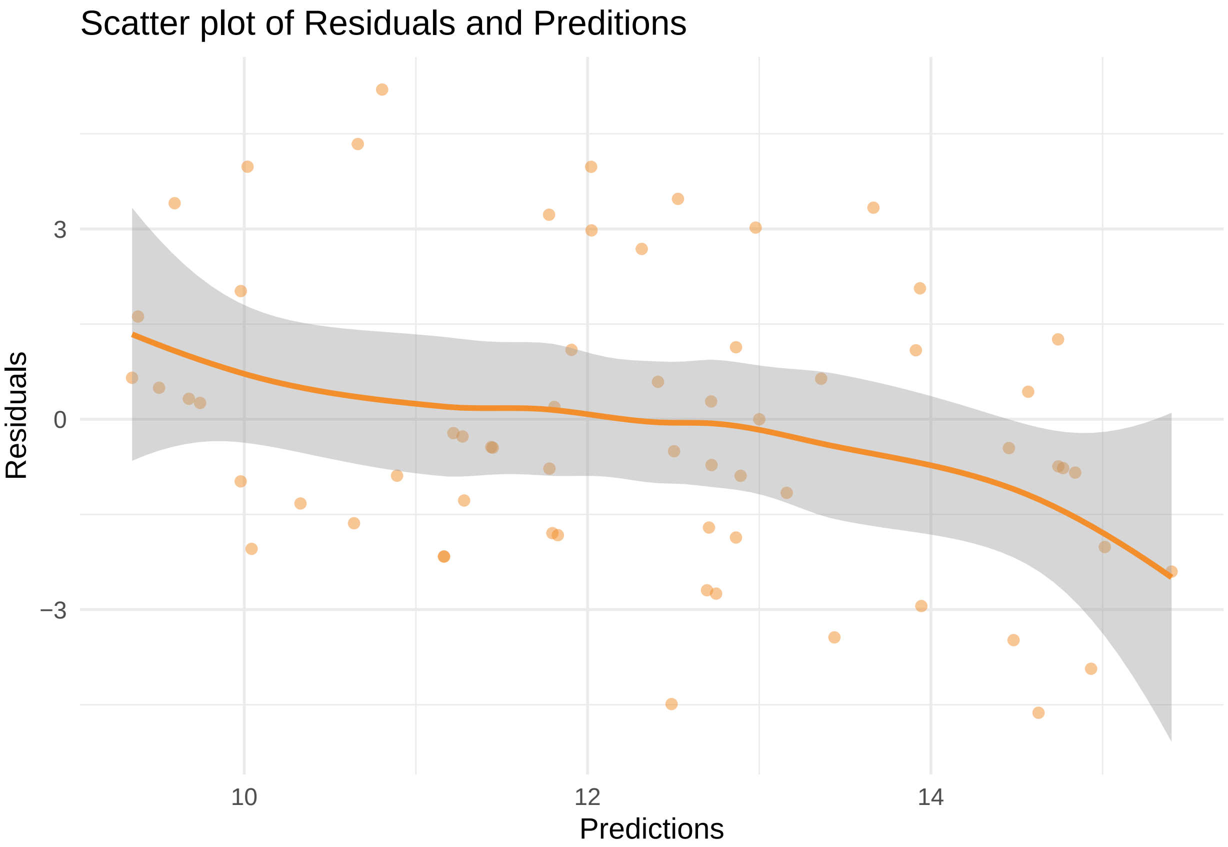
	predictions	scores	residual
645	14.626733	10	-4.626733
132	12.489300	8	-4.489300
46	14.932967	11	-3.932967
1	14.481567	11	-3.481567
522	13.437900	10	-3.437900
52	12.979067	16	3.020933
129	11.775567	15	3.224433
411	13.665200	17	3.334800
218	9.594533	13	3.405467
635	12.526600	16	3.473400
58	12.020867	16	3.979133
497	10.019267	14	3.980733
415	10.661600	15	4.338400

	predictions	scores	residual
213	10.803267	16	5.196733

From the above table we can see that there appears to be a relationship between scores and residuals. The plot below shows all of the residuals and scores for the test set.



The plot indicates that there is a positive correlation between scores and residuals. It is important to note that the residual is the actual score minus the prediction. This plot shows that the model is over predicting scores when the actual score is low and under predicting scores when the actual scores are high. This makes senses as the predictions are tending towards a mean. Although precision of the predictions might be improved we can expect to see a similar relationship in future models.



A plot of the residuals against the prediction shows that there is close to even variability across the range of predictions. This is a good indicator that the model is working correctly. In summary the RMSE and residual analysis show that the ensemble model functions correctly on the test set and that the level of accuracy is similar to that observed during training.

Conclusion

Exploratory analysis showed that survey and exam data for Portuguese is intact and cleaned. The final exams scores (G3) appear to follow a normal distribution with the exception of outliers around 0. The outlier results have been excluded from this project, further study and consultation is required to determine the exact meaning of a 0 score.

Following predictor selection and preprocessing this reports shows that a number of models were successfully applied to the data. All algorithms achieved a greater level of accuracy than a naive approach with an ensemble model proving to be the most effective at predicting students final scores. The ensemble model achieved a similar level of accuracy on the "hold out" test set. The writer believes that the predictions would prove to be a useful references for teachers, however to be certain in this view a subject matter expert would need to be consulted.

Future studies should investigate different approaches to this problem. One approach that may prove to be effective is developing a classification model the predicts whether or not a student will pass or fail the subject. This study has only focused on a single subject, the next step in this project should be to investigate if the final model can make good predictions for the maths data set. Detailed analysis could determine the difference between the data and justify an general model or support the single subject approach.

In summary this report has demonstrated that survey data collected prior to a Portuguese course can be used to make predictions of final grades. The accuracy of this model is sufficient to be used a reference for educators and maybe useful for allocating remedial intervention.