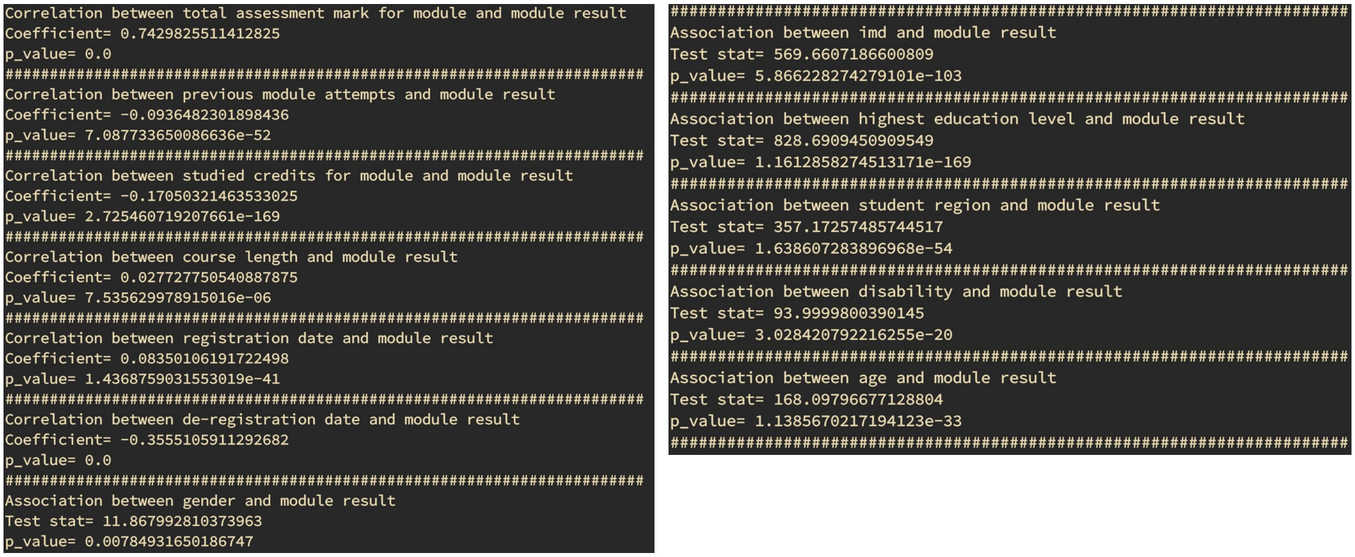
Machine Learning Report

Introduction:

The purpose of this report is to describe and explain the experimental procedure followed while investigating how 2 different machine learning models predict labels from the same data set. For the purposes of this experiment the Open University Learning Analytics Dataset (OULAD) is to be used as the data set and the goal is to predict the final module result of a student based on other attributes in the data. For my experiment I have chosen to use a random forest classifier and a support vector classifier. The random forest classifier was chosen as there are 4 different possible values for the final result making this problem a multi class classification problem, and the training time for a random forest is not affected by an increase of 2 to greater than 2 classes like other models would be. The support vector classifier was chosen to compare against the random forest as in order to use them for multi class classification problems you need to train a separate classifier for each pair of features, this harms training time; however, support vector classifiers have less of a tendency to over-fit the data set when compared to random forests therefore there are interesting comparisons to be made between the trade-offs between training time and overfitting.

Data Analysis:

Before starting to train either of the selected models I performed some statistical analysis on the data set to find any signs of a correlation or association between attributes and the final result of a module. The attributes I chose to investigate were all of those in the student info table, the course length (from the courses table), student registration and unregistration dates for a module (from the student registration table), and the total assessment score achieved by a student in a module. The rest of the attributes, belonging to the vle and student vle tables do not seem particularly useful and when combined with the student info table the number of records increases dramatically making manipulation of the data very difficult and training times incredibly long. To investigate for correlation the final results were encoded on a scale of 0 to 3 with 0 representing withdrawal (the worst result) and 3 representing distinction (the best result) with fail and pass in between. These new values were then used in turn with each set of numerical attributes from the data to carry out a Spearman's rank correlation coefficient calculation as well as calculate the associated p-value for said test. The p-value was then used to determine if the correlation was statistically significant using a 0.05 significance level. To investigate association between features and the final result I calculated the contingency matrix for each categorical feature and the final result (assuming that there was no association) and then performed a Chi-squared test, again with a significance level of 0.05 to determine statistical significance. The results are shown below:



```
Correlation between total assessment mark for module and module result
Coefficient= 0.7429825511412825
p_value= 0.0
########################################################################
Correlation between previous module attempts and module result
Coefficient= -0.0936482301898436
p_value= 7.087733650086636e-52
########################################################################
Correlation between studied credits for module and module result
Coefficient= -0.17050321463533025
p_value= 2.725460719207661e-169
########################################################################
Correlation between course length and module result
Coefficient= 0.027727750540887875
p_value= 7.535629978915016e-06
########################################################################
Correlation between registration date and module result
Coefficient= 0.08350106191722498
p_value= 1.4368759031553019e-41
########################################################################
Correlation between de-registration date and module result
Coefficient= -0.3555105911292682
p_value= 0.0
########################################################################
Association between gender and module result
Test stat= 11.867992810373963
p_value= 0.00784931650186747
```

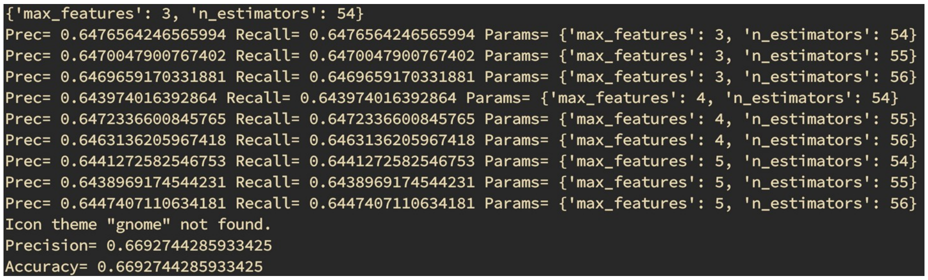
```
############################################################################
Association between imd and module result
Test stat= 569.6607186600809
p_value= 5.866228274279101e-103
############################################################################
Association between highest education level and module result
Test stat= 828.6909450909549
p_value= 1.1612858274513171e-169
############################################################################
Association between student region and module result
Test stat= 357.17257485744517
p_value= 1.638607283896968e-54
############################################################################
Association between disability and module result
Test stat= 93.9999800390145
p_value= 3.028420792216255e-20
############################################################################
Association between age and module result
Test stat= 168.09796677128804
p_value= 1.1385670217194123e-33
############################################################################
```

Figure 1: *Statistical test results*

The p-values for all of the statistical tests falls below 0.05 indicating a statistical significance. This may seem surprising due to the low values for the correlation coefficients and the Chi-squared statistic; however, due to the large sample size of around 27000 even a small correlation is considered to be significant. The largest correlation is from the total assessment mark which is to be expected, because of this I decided to no longer include it in the features in my training set as if I were to use assessment marks it would defeat the purpose of using machine learning as I would be better of taking an analytical approach and just working out module result boundaries instead. The second most significant statistic is the highest education therefore when splitting the training and test set I decided to use stratified sampling using the highest educaton feature to ensure a balanced training set.

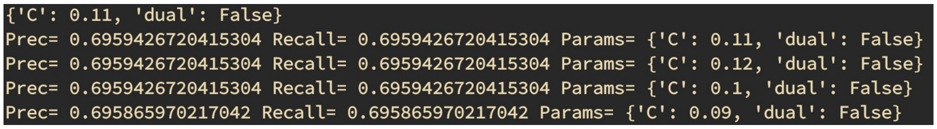
Model Training:

To evaluate the success of each model during training I used micro averaged precision over all classes being predicted, this statistic measures the proportion of correctly predicted instances of a class with total predicted instances of a class. I chose this measure of performance as it penalises models for over predicting a certain class. Both models were then trained on the training set using a grid search in order to determine the best hyperparameter combinations for each model and cross-validation to evaluate the model without overfitting the training set. The results of training are shown below:



```
{'max_features': 3, 'n_estimators': 54}
Prec= 0.6476564246565994 Recall= 0.6476564246565994 Params= {'max_features': 3, 'n_estimators': 54}
Prec= 0.6470047900767402 Recall= 0.6470047900767402 Params= {'max_features': 3, 'n_estimators': 55}
Prec= 0.6469659170331881 Recall= 0.6469659170331881 Params= {'max_features': 3, 'n_estimators': 56}
Prec= 0.643974016392864 Recall= 0.643974016392864 Params= {'max_features': 4, 'n_estimators': 54}
Prec= 0.6472336600845765 Recall= 0.6472336600845765 Params= {'max_features': 4, 'n_estimators': 55}
Prec= 0.6463136205967418 Recall= 0.6463136205967418 Params= {'max_features': 4, 'n_estimators': 56}
Prec= 0.6441272582546753 Recall= 0.6441272582546753 Params= {'max_features': 5, 'n_estimators': 54}
Prec= 0.6438969174544231 Recall= 0.6438969174544231 Params= {'max_features': 5, 'n_estimators': 55}
Prec= 0.6447407110634181 Recall= 0.6447407110634181 Params= {'max_features': 5, 'n_estimators': 56}
Icon theme "gnome" not found.
Precision= 0.6692744285933425
Accuracy= 0.6692744285933425
```

Figure 2: *Random forest grid search*



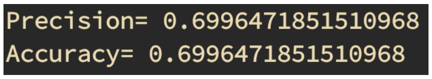
```
{'C': 0.11, 'dual': False}
Prec= 0.6959426720415304 Recall= 0.6959426720415304 Params= {'C': 0.11, 'dual': False}
Prec= 0.6959426720415304 Recall= 0.6959426720415304 Params= {'C': 0.12, 'dual': False}
Prec= 0.6959426720415304 Recall= 0.6959426720415304 Params= {'C': 0.1, 'dual': False}
Prec= 0.695865970217042 Recall= 0.695865970217042 Params= {'C': 0.09, 'dual': False}
```

Figure 3: *SVC grid search*

During training weaknesses of both models came to light. Due to the huge sample size in the training set I could not use any polynomial features in the SVC as training times became hours long. Due to the random nature of random forest classifiers hyperparamter tuning was hard as scores varied randomly between training runs.

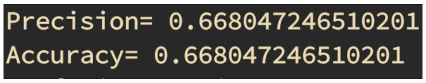
Final Results:

After hyperparamter tuning the models were both evaluated on the test set. The results are shown below in the form of the precision scores and the confusion matrices for both models:



```
Precision= 0.6996471851510968
Accuracy= 0.6996471851510968
```

Figure 4: *SVC final scores*



```
Precision= 0.668047246510201
Accuracy= 0.668047246510201
```

Figure 5: *Random forest final scores*

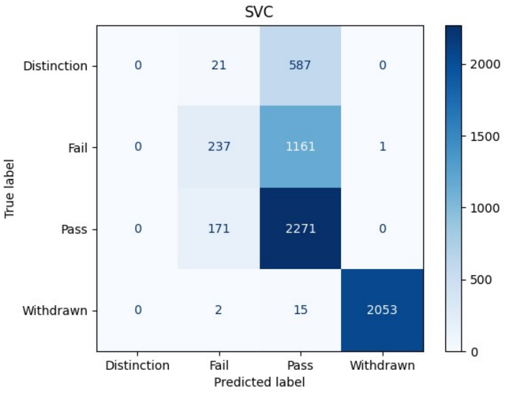


Figure 6: *SVC confusion matrix*

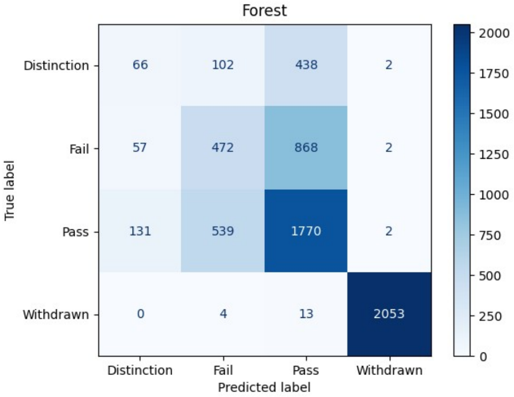


Figure 7: *Random forest confusion matrix*

The final results show that the support vector classifier performed better than the random forest as it was more likely to correctly predict a students final result; however, the difference in precision score is only 0.03 so this could just be a result of the training and test set split; therefore, I believe it would be more fair to conclude that both models performed similarly instead. This is further reflected in the confusion matrices that show both models were very good at predicting withdrawals both only failing to predict 17 of them. The SVC was better at predicting passes than the random forest but the confusion matrix shows this is likely due to the SVC massively over predicting passes. This over predicting of passes was also seen in random forest (to a lesser extent) for this reason I have concluded that both models suffered from overfitting the training set resulting in poor predicting of fails and distinctions.