Table of Contents

Executive Summary

In this data mining report, we have used WEKA as our main software to identify the question we faced. Data mining on WEKA is an effective method to help us improve our findings.

It is believed that student graduates have been a factor in the future income. Multiple factors and variables that are provided are the data gained on the student income 2 years after the graduation. Moreover, the covid-19 pandemic caused uncertainties. Using WEKA and its multiple algorithms will eventually remove unnecessary variables and gain some knowledge from the data provided.

In a glance, the class of the data is the income_group variable, and some of the variables are not set accordingly. The variables seem to be having missing values and outliers on the numerical data. We used transformation to change the datatype accordingly, remove the unwanted attribute, and replace missing values before discretizing and SMOTE.

As to train the datasets, we used ZeroR as the benchmark, followed by other 4 models are J48, NaiveBayes, IBk, and SimpleLogistic. The train results showed that IBk has the best training model but overfitted the models. Therefore our second best model is tuned J48 which has the best result in evaluating the model.

In the analysis part of the writing process, we will visualise the data and compare the income group in each category. We realised that graduates with a Bachelor degree in Multidisciplinary, health degree holders, graduates who completed the four-year program will likely get a higher pay. As most of the data has illustrated that they will fall in “above average” or “high” income groups.

Besides that, we also found out a bachelor degree holder is more likely to be paid a higher salary than a graduate who only obtains a certificate degree after graduation. Companies will probably be willing to pay higher salaries to graduates from well-known "private for non-profit" institutions.

In conclusion, J48’s accuracy, weighted F-measure, and also the weighted ROC have been increased significantly. It is clearly shown that the algorithm is suitable for us as our final model. For the recommendation, we think that it is better to have more time to complete it as the exceedingly huge amount of data and its features makes it more complicated to straightway understand some insights, and it is time-consuming to bring out the best optimal model.

TeamC5 - Group Reflective Video: *https://youtu.be/tvcazateshk*

Introduction and Methodology

In order to determine the factors behind a graduate's future income, we are going to discuss the options and education level of a student along with other variables that are given in the datasets. What are the advantages of continuing your education? Students wonder if their academic achievements will be worthwhile in the long run. They choose to pursue higher education for a variety of reasons. The expectancy of future salary based on educational achievement is one of the most motivating.

We believe that most students have chosen their universities and academic programs based on the desired incomes in their future goals. Therefore, people are asking the question which option they made will provide the most income in their future life. So, we would like to find out what attributes are important and meaningful for the students to get a higher pay after their graduation.

The datasets provided include training and testing data, containing information on 13,818 students' income, 2 years after their graduation. The variables in the given datasets are focused on four sectors, namely **academic program** (academics program in different field, percentage of degrees awarded in field, cost of tuition fees, etc), **school related variables** (highest degree awarded at institution, staff salaries, level of institution, state of the institutions, etc), **student characteristics** (gender, part time/full time, marital status, etc), **degree related variables** (such as number of degree-types offered at institution). There are 265 variables in total and income group is the class label of the datasets. The result of the income group has been classified into four groups: **below average, average, above average** and **high**.

In this period of uncertainty, many youths are feeling uncertain about the future especially for high school and university students, the COVID-19 pandemic has brought a huge impact to their lives. Job possibilities are scarce and limited in the coming years, no matter what your degree or educational level is. It is often believed that the better your educational background, the greater your possibilities of being hired for a decent job (Elmi, 2010). The main goal of this report is to come up with an idea of a positive decision guideline for identifying relevant variables for generating the most income for the future of graduate students.

We are going to solve the problem by merging the training and testing data, then giving them a row of new id. There will be 13,818 instances and 265 attributes (train= 1-10,169; test= 10,170-13,818). For our data pre-processing part on WEKA, we filter out the attributes by applying *NumericToNominal*. Then, we will set missing values threshold and correlation threshold, remove those attributes which have more than 5% of missing values while have a correlation lower than 0.1. After the removal, we will use the feature of *ReplaceMissingValues* in the filter for those attributes which are not removed.

Now, we will discretise the remaining attributes. Furthermore, we will remove the attributes with attribute selection of *InfoGainAttributeEval* and *Ranker search method*, we decided to set a threshold at 0.1. As a result, we will have 32 attributes left for our data.

Then, we filter the instances by applying the *RemoveRange* feature for our train and test data. Our instances remain unchanged. Lastly, we resample the training dataset by applying the Synthetic Minority Oversampling TEchnique (SMOTE).

In the modelling part, we will first set a benchmark by using the ZeroR algorithm. Then, we will do the modelling by using different classifiers, namely **NaiveBayes, J48, IBk** and **SimpleLogistic.** For the evaluation of test data, we will apply the same algorithms by selecting the supplied *test set* to get the result. After getting the result from our modelling, we will focus on the accuracy, F-measure, ROC Area and time-taken. We will choose the model with high accuracy, high F-measure, high ROC Area with least time-taken as our standard.

Moreover, we will analyse our report by focusing on the valuable insights we have generated from the data mining process and we will highlight our visualisations in that stage. In the end of the report, we will select our best model and bring along with our recommendations. We will define the key factors that generate the most income in the future.

Data Preprocessing

The first step of data preprocessing is merging the original train and test datasets provided using MS Excel. This is done to tackle the variable and instance label discrepancy issues that existed in both datasets. The combined dataset has a total of 265 variables including the row ID and 13,818 instances consisting of 10,169 train data instances and 3,649 test data instances. This merged data file is then ready to be transformed in WEKA.

Upon uploading to WEKA, the first variable which is the row ID was removed as it only acts as identifier values for the instances and might affect the overall Classification performance. Thus, only the 263 remaining predictor variables and 1 target variable are used in the next step.

a. Summary Statistics

The data comprises 264 variables for 13,818 students' income, 2 years after their graduation. The data was compiled from a wide range of publicly available government data sources. The attribute, *income_group* is the Class label in this dataset. These 264 predictor variables can be used to evaluate 3,649 students' income group 2 years after their graduation.

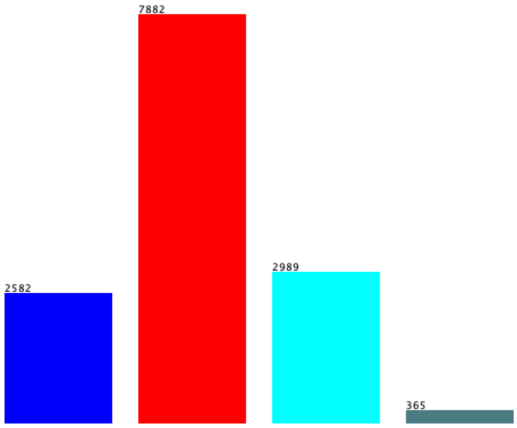


Figure 1: Summary statistics for the class attribute

The class attribute is divided into four income groups: "Below Average," "Average," "Above Average," and "High." There are 2582 instances in the category "Below Average," 7882 instances in the category "Average," 2989 instances in the category "Above Average," and 365 instances in the category "High." Based on the summary statistics presented above, we can conclude that the class attribute has an imbalanced distribution. This is due to the fact that the "High" income class has 21.5 times fewer instances than the "Average" income group. An unbalanced dataset will bias the prediction model towards the more common class. Hence, action is needed to deal with the problem.

- **Attribute 229, 230 and 235 (cost__tuition_[in/out]_state, school_faculty_salary)**
  They are all numerical variables. Their summary statistics are skewed to the left and contain some outlier values. All of them have a high proportion of missing values (>5%) as compared to other attributes that have a lower percentage of missing values. Furthermore, all of the attributes have a high standard deviation, indicating that they are not good variables to be included in the model.

- **Attribute 236 (school_ft_faculty_rate)**
  This is a numerical attribute that illustrates the full-time staff rates. It is demonstrated in percentage form ranging between 0 to 1. It consists of 4866 missing values which account for 35% of the total instances. We can conclude that it is not a good predictor and should not be added into the model.

- **Attribute 238, 244 and 258 (school_instuctional_expenditure_pre_fte, school_tuition_revenue_per_fte and student_size)**
  All of them are numerical attributes. They consist of 1% of missing value which is relatively lesser (<5%) as compared to other attributes who have a higher percentage of missing value. Their summary statistics have shown that the distribution curve is skewed to the left and have outlier values. Both of them also have extremely high standard deviations which are not good to be fitted into our model.

- **Attribute 247,252 and 253 (student_demographics_female_share, student__share_25_older and student_share_first_time_full_time)**
  All of them are numerical variables. They are shown in percentage form ranging between 0 to 1. All of them have a high proportion of missing values (>5%) as compared to other attributes that have a lower percentage of missing values. Their summary statistics have shown that the distribution curve is "abnormal". Hence, they are not a good variable to be included into our model.

- **Attribute 249 and 250 (student_demographic_married and student_demographics_veteran)**
  They are all numerical variables that show the proportion of married and veteran students. Both of them are demonstrated in percentage form ranging between 0 to 1. Their summary statistics are skewed to the left and contain some outlier values. All of them have a high proportion of missing values (>5%) as compared to other attributes that have a lower percentage of missing values. Thus, they are not good predictor variables.

- **Attribute 255 and 256 (student_share_firstgeneration_parents_highschool and student_share_firstgeneration_parents_somecollege)**
  All of them are numerical variables. They are shown in percentage form ranging between 0 to 1. All of them have a high proportion of missing values (>5%) as compared to other attributes that have a lower percentage of missing values. Their summary statistics have shown that the curves are distributed normally.

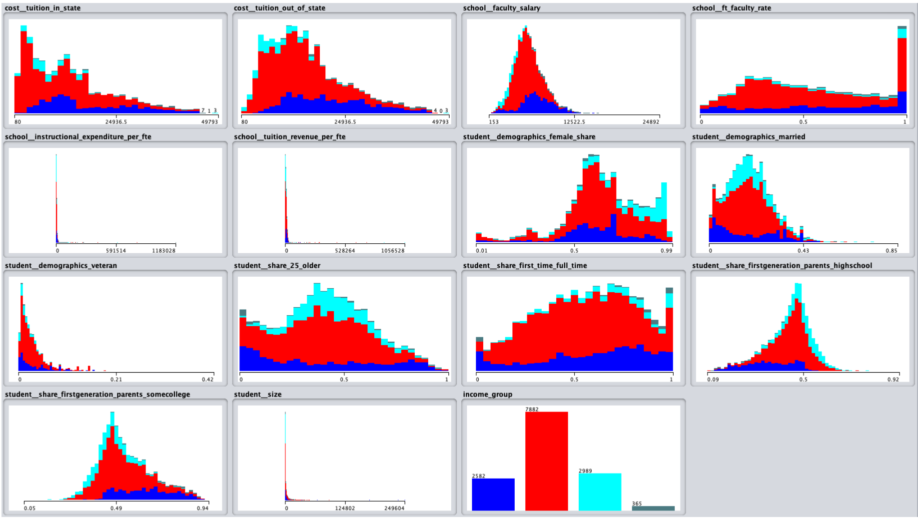
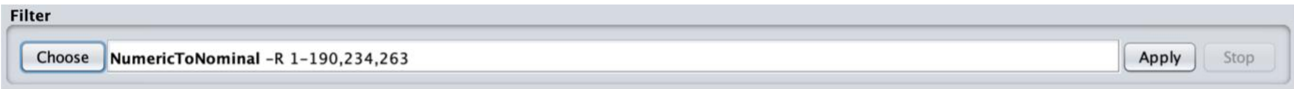


Figure 2: summary statistics for all highlighted attributes

b. Transformation

As mentioned in the above part, the *income_group* is the class attribute in this dataset. Since this variable has categorical values, it is suitable for us to use the Classification Predictive method in our modelling process. Thus, the numerical predictive attributes are required to be transformed into their nominal counterparts.

Firstly, the *NumericToNominal* filter is put on to the wrongly identified numerical variables which are supposedly measured as categorical values. These variables include *academics__program_[name]_[field]* represented by 1^{st}–190^{th} variables *school__degrees_awarded_predominant_recoded* represented by 234^{th} variable and *degrees_highearning*represented by 263^{rd} variable.



Secondly, the dataset needs to be cleaned by removing or replacing the visible missing values. Based on the summary statistics provided in WEKA, the existing missing values proportion varies between 0% and 48%. Nevertheless, the consensus threshold is set at 5% as most of the portions are below this value. The removal of attributes was done to variables with missing values of more than 5% and correlation value of less than 0.1 with *income_group* (*Appendix 1*).

Using the default settings of *CorrelationAttributeEval* and *Ranker* as shown above, attributes *school__ft_faculty_rate* (236), *school__online_only* (240), *student__demographics_married* (249), *student__demographics_veteran* (250), *student__share_25_older* (252) and *student__share_first_time_full_time*(253) are removed from the datasets. Whereas, the remaining variables with missing values are fixed using *ReplaceMissingValues* filter.

Thirdly, the remaining numerical variables are transformed into categorical counterparts by binning it using the default settings of the Discretization filter in WEKA. Moreover, using *Discretization* is more appropriate in this dataset as it would resolve the outlier and extreme values problems without actually removing it with the usual *InterquartileRange*filter. One of the reasons is that most of the independent variables consist of several extreme values which might be useful in predicting the High category in *income_group* class attribute. The variables that were discretized include *academics__program_percentage_[field]* (191–228), *cost__tuition_[in/out]_state* (229–230), *school__faculty_salary*(235), *school__instructional_expenditure_per_fte* (237), *school__tuition_revenue_per_fte* (242) and several student and degrees related variables (243–256).

Filter

Choose | **Discretize** –B 10 –M –1.0 –R 191–230,235,237,242–256 –precision 6 | Apply | Stop

c. Attribute Selection

The 257 predictors are further processed with a feature selection method where the irrelevant and redundant attributes are removed to have a better estimation performance. After doing several iterations in the Select Attributes panel with different parameter settings of Attribute Evaluator and Search Method, the *InfoGainAttributeEval* and *Ranker* is proven to project the most accurate rank selection of the variables. Based on the attribute selection output and some trial & errors, the removal is done to a total of 226 attributes that have values below the determined threshold which is 0.1. The remaining 31 significant predictors and 1 target variable are ready to be used in further modelling process.

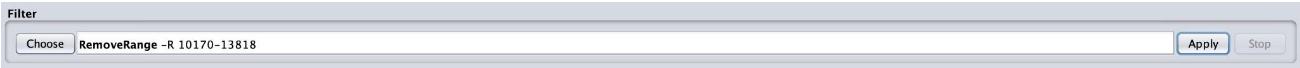
Filter

Choose | **AttributeSelection** –E "weka.attributeSelection.InfoGainAttributeEval " –S "weka.attributeSelection.Ranker –T –1.7976931348623157E308 –N –1" | Apply | Stop

```
Attributes:   32
              school__degrees_awarded_highest
              school__degrees_awarded_predominant
              school__degrees_awarded_predominant_recoded
              school__institutional_characteristics_level
              academics__program_percentage_personal_culinary
              school__ownership
              cost__tuition_out_of_state
              cost__tuition_in_state
              academics__program_certificate_lt_2_yr_personal_culinary
              school__faculty_salary
              student__share_firstgeneration
              student__demographics_first_generation
              student__share_firstgeneration_parents_somecollege
              academics__program_bachelors_business_marketing
              student__share_firstgeneration_parents_highschool
              academics__program_bachelors_computer
              student__share_independent_students
              student__demographics_dependent
              academics__program_bachelors_psychology
              academics__program_certificate_lt_1_yr_personal_culinary
              academics__program_bachelors_health
              school__state
              degrees__total_count
              academics__program_bachelors_english
              academics__program_bachelors_history
              academics__program_bachelors_social_science
              academics__program_bachelors_mathematics
              academics__program_bachelors_multidiscipline
              academics__program_percentage_business_marketing
              academics__program_bachelors_biological
              school__region_id
              income_group
```

The relationship of these attributes can be seen in the MS Excel Correlation Matrix as well as the WEKA Visualization panel (*Appendix 2)*.

According to the graphs above, it is shown that there are more strong positive correlations between the variables compared to the negative correlations. These attributes are the final variables to be included in the modelling process. However, it is not suitable to use 100% of the data since it might cause overfitting issues. Thus, the data was required to be split back into 2 dataset i.e., train data and test data in accordance with the ordering of the original dataset provided. This step is done by using the *RemoveRange* filter. Firstly, to obtain the 10169 train data set, the *instancesIndices* is set at 10170-13818. After the training dataset is saved, the step is reverted by clicking the 'Undo' button. Then, the remaining would be the 3649 test data instances which can be obtained using the same parameter but with *invertSelection* as 'True'.



Upon opening the training dataset in WEKA, the target attribute is shown to be unbalanced which might affect the overall modelling performance. Thus, income_group variable was rebalanced using SMOTE to the 4th indices in class value and it is set to add 500% instances. This final training dataset consists of 10774 instances which are then ready to be used for modelling purposes.

Modelling

a) Setting Benchmark with ZeroR

ZeroR algorithm is chosen to be the benchmark since it is the simplest algorithm that heavily relies on the target rather than the variables.

Model #1 (Base)	**Scheme**	weka.classifiers.rules.ZeroR				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	54.2324%		**Time Taken**	0.00 seconds	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		?	0.703	?	?	?
	ROC Curve	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.499	0.500	0.499	0.498	0.499

b) Train Datasets

In the training processes, we selected another 4 algorithm that are :

- NaiveBayes is an algorithm create an assumptions on the datas (both presence or absence) that is unrelated towards each other
- J48 is the algorithm to create a pruned or unpruned decision tree, and it is commonly used to examine the data categorically and continuously in the data mining process. In this modelling process, the J48 was originally set at 0.25 *confidenceFactor*. However, the tuned version with 0.5 *confidenceFactor* provides a better performance than the output of default parameter setting.
- IBk is an algorithm to classify each class with the similarity between the instances "neighbours". It uses the K-Nearest Neighbour where the gap between each instance in training data is classified based on the distances between instances similarity.
- Simple Logistic regression is an algorithm that looking and taking assumption on the relationship and it produces discrete output

IBk Training Result

No tuning required as the base model has the best output. The accuracy decreases as the changes made on the K- Value. Therefore, it is concluded that we will test using the standard settings. Moreover, IBk has the best performing model among the other 4 models.

Model	**Scheme**	weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.				
	Relation	weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 Attributes				
	Accuracy	82.5042%		**Time Taken**	0.03s	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.796	0.854	0.757	0.934	0.825
	ROC	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.925	0.884	0.912	0.985	0.904

Evaluation results using test data

ZeroR

Model #1 (Base)	**Scheme**	weka.classifiers.rules.ZeroR				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818-				
	Attributes	32 attributes				
	Accuracy	55.8783%		**Time Taken**	0.05 seconds	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		?	0.717	?	?	?
	ROC Curve	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.500	0.500	0.500	0.500	0.500

Tuned J48

Model #3 (J48)	Scheme	weka.classifiers.trees.J48 -C 0.25 -M 2				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	82.7076%		Time Taken	0.08 seconds	
	F-Measure	Below Avg	Avg	Above Avg	High	Weighted
		0.793	0.834	0.704	0.911	0.807
	ROC Area	Below Avg	Avg	Above Avg	High	Weighted
		0.927	0.858	0.895	0.973	0.887

Comparison on training and test modelling

Model	Accuracy		Time Taken (in seconds)		F-Measure Weighted Avg		ROC Area	
	Training	Test	Training	Test	Training	Test	Training	Test
ZeroR	54.2324%	55.8783%	0	0.05	?	?	0.499	?
Naive Bayes	59.5576%	55.6317%	0.664	0.06	0.809	0.555	0.809	0.794
Tuned J48	80.7964%	82.7076%	0.07	0.08	0.807	0.807	0.887	0.887
IBk	82.5042%	91.5419%	0.03	4.4s	0.825	0.914	0.904	0.961
Simple Logistic	79.0236%	75.5549%	52.96	53.99	0.788	0.751	0.917	0.9

Training Dataset

From the result shown above, The algorithm that has improved its accuracy during the test data is ZeroR, Tuned J48, and IBk, most of the model has a longer processing time during evaluation using the test data except for Naive Bayes. Moreover, the F-measure which the accuracy on the test indicators improved on the IBk, stayed the same for Tuned J48 and decreased for Naive Bayes and Simple Logistic algorithm. Lastly, Naive Bayes, Tuned J48 and IBk improved it's ROC area which also a measurement of the algorithm performance but slightly decreased on the Simple Logistic algorithm.

In general, all of the algorithms have passed the benchmark which has been set by the simplest algorithm, ZeroR with a score of 54.2324%.

Based on the output as shown above, the Naive Bayes algorithm has the least percentage of correctly classified instances, weighted F-measure and also the weighted receiver operating characteristic (ROC) as compared to other algorithms. Meanwhile, the IBk algorithm has the highest proportion of correctly classified instances, as well as the highest score for both weighted F-measure and weighted ROC scores.

Furthermore, after tuning the confidence factor of the J48 algorithm from 0.25 to 0.5, we discovered that the percentage of correctly classified instances improved by 1.6707 percent. As a result, we opted to use the tuned version to evaluate our test dataset as it has the higher capability in estimating data.

To summarise, IBk has the highest level of understanding of how the given input variables are associated with the class attribute.

Testing Dataset

On the test data evaluation, J48, IBk and SimpleLogistic have passed the benchmark which has been set by the simplest algorithm, ZeroR with a score of 55.8783%, while NaiveBayes has a lower percentage of correctly classified instances than ZeroR. Hence, we will omit NaiveBayes in our following comparison analysis.

Based on the output as shown above, the SimpleLogistic algorithm has the lowest percentage of correctly classified instances, weighted F-measure and also the weighted receiver operating characteristic (ROC) as compared to other algorithms. Meanwhile, the IBk algorithm has the highest percentage of correctly classified instances, weighted F-measure and also the weighted ROC.

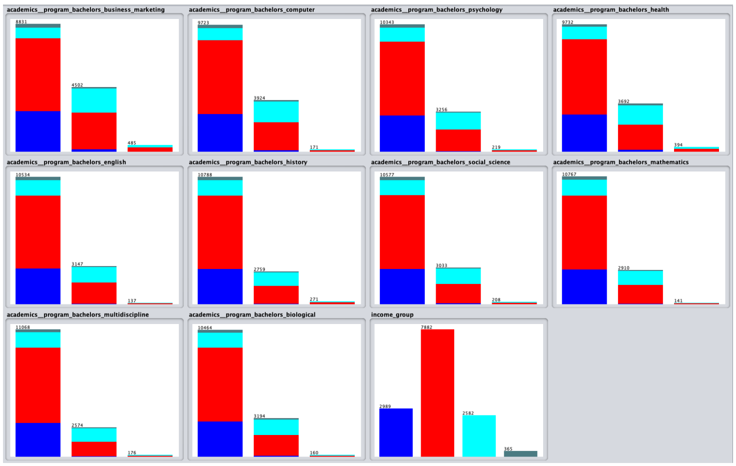
Although the IBk algorithm has the highest accuracy in predicting the model, it is inadequate to be chosen as the best classifier as it has an incredibly high percentage of correctly classified instances with a score of 91.5419%. This is due to the fact that when the score of correctly classified instances exceeds 85% and above, the model becomes overfitting. Overfitting occurs when a model learns the detail and noise in the data to the extent that it negatively impacts the performance of the model on new data. This implies that the algorithm picks up on noise or random fluctuations in the input and learns it as a concept.

As a result, the J48 (tuned) algorithm will be our most optimal model. Throughout the iteration processes, the accuracy, weighted F-measure, and also the weighted ROC of the J48 model have been increased significantly. Thus, it is suitable for us to use the **J48** as our final model.

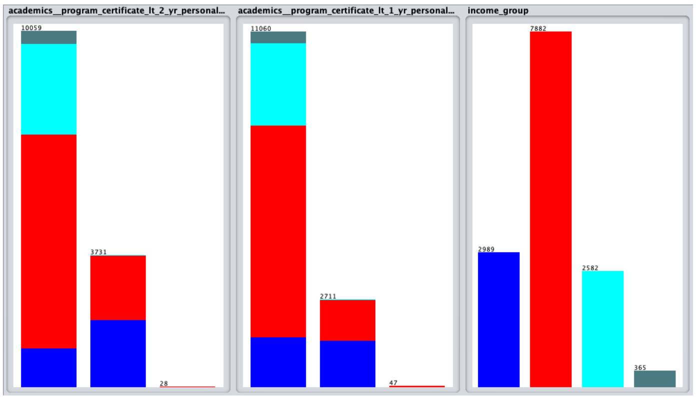
Analysis

The "academics_program_bachelor_history" illustrates that graduates with a Bachelor degree in Multidisciplinary have the lowest probability of getting "below average" income as compared to other degree holders. From the summary of this attribute, we can visually see that it has the least "below average" income group (dark blue) for the "offered" category.

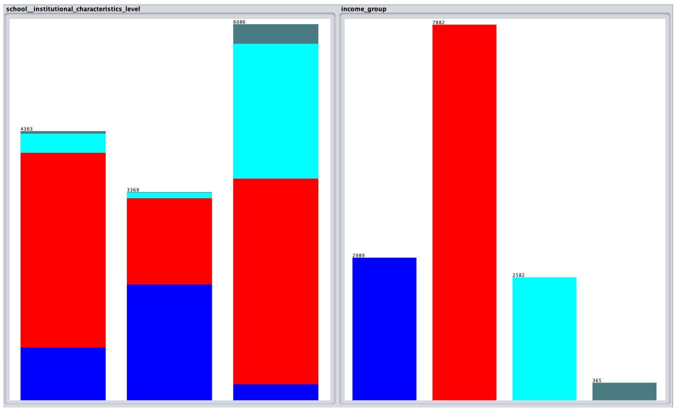
Moreover, the "acedemics_program_bachelor_health" attribute shows us that graduates with a Bachelor degree in Health have the highest probability of having a "high" income as compared to other degree holders . As we know, "health" professions normally have higher salaries. It shows that it has the highest proportion of "high" income groups under the "offered" category.



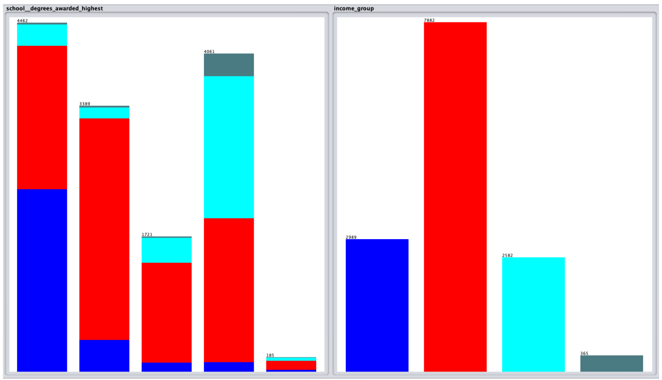
The "academics_program_cerificate_lt_2_yr_personal_culinary" attribute indicates that graduates with a Certificate Degree for 2 years in Personal culinary are more likely to earn "below average" income than graduates with a Certificate Degree for 1 year in Personal culinary. It shows that it has the highest percentage of "below average" income groups under the "offered" category.



The "school_institutional_characteristics_level" attribute shows that graduates from institutions who have studied a program which is less than 2 years have the highest percentage of getting "below average" income. On the contrary, graduates from institutions who have studied a program for 4 years have the highest probability of getting a "high" or "above average" income. As a result, we can conclude that graduates who completed the four-year program have a better chance of earning a higher salary.



When we compare the two types of academic programs, Bachelor Degree and Certificate Degree, we can see that a Bachelor Degree holder is more likely to be paid a higher salary than a graduate who only obtains a certificate Degree. Based on our common knowledge, we know that people who graduate with a Bachelor Degree tend to have a higher salary as compared to those who graduated with a Certificate Degree. This indicates that it is better to obtain a Bachelor Degree rather than taking a Certificate Degree regardless of the programs taken. This is proven by the attribute "school_degree_awarded_highest" as we can see Certificate Degree holders have the highest probability of getting "below average" income as compared to other degree programs.



Based on the analysis above, we would recommend that students should take at least a Bachelor Degree program so that they will have a higher income when starting to work. Furthermore, Certificate Degree holders should further their studies to get a higher income. Graduates seeking a higher income job are advised to obtain a "Graduate Degree," regardless of the programs they are pursuing at the institution, as obtaining a "Graduate Degree" will increase their chances of obtaining an "above average" or "high" income profession.

Moreover, it is important for students to take into consideration what type of institution they graduate from. Students who graduated from a "private for non-profit" institution tend to have the highest probability of getting an "above average" or maybe a "high" income. This may be due to the fact that "private for non-profit" institutions are mainly private prestigious universities that usually have higher tuition fees. Companies will almost certainly be willing to pay higher salaries to graduates from well-known institutions.

Conclusion and Recommendations

The most appropriate technique to handle this dataset is Classification Analysis as our objective is to see what are the factors that increase income in the future (categorical variable). In the data preparation, the training dataset is pre-processed using several filters such as NumericToNominal, ReplaceMissingValues, Discretize, and SMOTE. The cleaned and preprocessed dataset consists of a total 31 significant predictors to estimate the target classification.

As mentioned in the introduction part, the 5 modelling algorithms used in this report are *ZeroR*, *NaiveBayes*, *J48*, *IBk*, and Simple Logistic Regression. After running iterations on the algorithms and its corresponding parameter settings, the J48 (tuned) algorithm will be our most optimal model using the method Cross Validation (k=10). This is because the J48 algorithm has performed consistently throughout the training and testing dataset and it has an optimum percentage of correctly classified instances after eliminating all of the irrelevant attributes by using "InfoGainAttributeEval" and "Ranker".

Furthermore, as opposed to other algorithms, J48's confusion matrix has the optimal number of wrongly classified instances. Although the IBk algorithm's confusion matrix has the least number of wrongly classified instances, it seems to be overfitting. This is due to the fact that it has an incredibly high percentage of accuracy for all of its performance indicators.

J48 (631 wrongly classified instances)

```
=== Confusion Matrix ===

    a    b    c    d   <-- classified as
  579  150    2    0 |    a = Below Average
  109 1812  112    6 |    b = Average
    1  110  515    9 |    c = Above Average
    4   34   94  112 |    d = High
```

IBk (309 wrongly classified instances)

```
=== Confusion Matrix ===

    a    b    c    d   <-- classified as
  717   14    0    0 |    a = Below Average
  108 1862   65    4 |    b = Average
    1   11  622    1 |    c = Above Average
    3   22   80  139 |    d = High
```

Throughout the iteration processes, the accuracy, weighted F-measure, and also the weighted ROC of the J48 model have been increased significantly. Thus, it is suitable for us to use the **J48** as our final model.

We apply the principles of CRISP-DM using each of the methods described in order to provide clear and workable business intelligence models based on the dataset given. We use advanced data modeling and machine learning methods to create new meaning to help students to choose what to study but a recurring concern is future income.

There are some recommendations to obtain a better understanding and estimation of predicting the future income of higher education students. A better way of visualizing can be developed if more time is given. The exceedingly large amount of data and its features makes it difficult to immediately grasp some insights, and it is time-consuming to create the most optimal model. Our group might also have some unsupervised approaches such as trying to combine some of the similar features together and identifying the association between the variables. Thereby, we could obtain some possible alternative techniques in determining what factors affect future earnings of university students.

References

Brownlee, J. (2021). *Overfitting and Underfitting With Machine Learning Algorithms*. Machine Learning Mastery. Retrieved 30 May 2021, from https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/.v

Elmi, N. A. (2010). Earning-Education Correlation. *Research of International Horn University*.

WEKA. (2016). RemoveRange. Retrieved from https://weka.sourceforge.io/doc.stable/weka/filters/unsupervised/instance/RemoveRange.html

WEKA. (2020). CorrelationAttributeEval (weka-dev 3.9.5 API). Retrieved 2021, from https://weka.sourceforge.io/doc.dev/weka/attributeSelection/CorrelationAttributeEval.html

Appendices

Appendix 1. CorrelationAttributeEval - Remove variables that have more than 5% missing values and are not highlighted.

```
=== Run information ===

Evaluator:    weka.attributeSelection.CorrelationAttributeEval
Search:       weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:     College_Income_Train_Test_NoComma-weka.filters.unsupervised.attribute.Remove-
R1
Instances:    13818
Attributes:   264
              [list of attributes omitted]
Evaluation mode:    evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
	Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 264 income_group):
	Correlation Ranking Filter
Ranked attributes:
 0.34367    216 academics__program_percentage_personal_culinary
 0.23335    232 school__degrees_awarded_highest
 0.21246      4 academics__program_assoc_business_marketing
 0.21223     34 academics__program_assoc_security_law_enforcement
 0.21181      7 academics__program_assoc_computer
 0.21102    233 school__degrees_awarded_predominant
 0.20953    118 academics__program_certificate_lt_2_yr_business_marketing
 0.20868    259 degrees__total_count
 0.20601     11 academics__program_assoc_engineering_technology
 0.20483    260 degrees__assoc_count
 0.20403    129 academics__program_certificate_lt_2_yr_health
 0.20327    140 academics__program_certificate_lt_2_yr_personal_culinary
 0.20002    148 academics__program_certificate_lt_2_yr_security_law_enforcement
 0.19937    143 academics__program_certificate_lt_2_yr_precision_production
 0.19852    241 school__ownership
 0.19834    256 student__share_firstgeneration_parents_somecollege
 0.19661     38 academics__program_assoc_visual_performing
 0.19603     15 academics__program_assoc_health
 0.19478    136 academics__program_certificate_lt_2_yr_mechanic_repair_technology
 0.19461    235 school__faculty_salary
 0.19367    237 school__institutional_characteristics_level
 0.19171    230 cost__tuition_out_of_state
 0.1915     262 degrees_certificate_count
 0.19149    207 academics__program_percentage_humanities
 0.18992     98 academics__program_certificate_lt_1_yr_mechanic_repair_technology
 0.18892     22 academics__program_assoc_mechanic_repair_technology
 0.18795    248 student__demographics_first_generation
 0.18795    254 student__share_firstgeneration
 0.1877     125 academics__program_certificate_lt_2_yr_engineering_technology
 0.18647    131 academics__program_certificate_lt_2_yr_humanities
 0.18477    255 student__share_firstgeneration_parents_highschool
 0.1842      19 academics__program_assoc_legal
 0.18279     29 academics__program_assoc_precision_production
 0.18123    105 academics__program_certificate_lt_1_yr_precision_production
 0.18024    234 school__degrees_awarded_predominant_recoded
 0.17907     87 academics__program_certificate_lt_1_yr_engineering_technology
 0.17843     26 academics__program_assoc_personal_culinary
 0.17665     17 academics__program_assoc_humanities
 0.17612    102 academics__program_certificate_lt_1_yr_personal_culinary
 0.17434    152 academics__program_certificate_lt_2_yr_visual_performing
 0.17367    110 academics__program_certificate_lt_1_yr_security_law_enforcement
 0.16891    263 degrees__high_earning
 0.16639    229 cost__tuition_in_state
 0.1656      14 academics__program_assoc_family_consumer_science
 0.16451    121 academics__program_certificate_lt_2_yr_computer
 0.16299     80 academics__program_certificate_lt_1_yr_business_marketing
 0.16214     91 academics__program_certificate_lt_1_yr_health
 0.16028    128 academics__program_certificate_lt_2_yr_family_consumer_science
 0.1579      93 academics__program_certificate_lt_1_yr_humanities
 0.15569     48 academics__program_bachelors_engineering
 0.15492     84 academics__program_certificate_lt_1_yr_construction
 0.15377    247 student__demographics_female_share
 0.15316     90 academics__program_certificate_lt_1_yr_family_consumer_science
 0.15221     62 academics__program_bachelors_multidiscipline
 0.15051      8 academics__program_assoc_construction
 0.14972     53 academics__program_bachelors_health
 0.14928    122 academics__program_certificate_lt_2_yr_construction
 0.14871    133 academics__program_certificate_lt_2_yr_legal
 0.14848     83 academics__program_certificate_lt_1_yr_computer
 0.1462       9 academics__program_assoc_education
 0.14601      6 academics__program_assoc_communications_technology
 0.14504    114 academics__program_certificate_lt_1_yr_visual_performing
 0.14415     45 academics__program_bachelors_computer
 0.14281     51 academics__program_bachelors_ethnic_cultural_gender
 0.1412      76 academics__program_bachelors_visual_performing
 0.13797     68 academics__program_bachelors_psychology
 0.13395    200 academics__program_percentage_engineering
 0.13369     73 academics__program_bachelors_social_science
 0.13243     42 academics__program_bachelors_business_marketing
 0.13119     41 academics__program_bachelors_biological
 0.13013     95 academics__program_certificate_lt_1_yr_legal
 0.1289     120 academics__program_certificate_lt_2_yr_communications_technology
 0.12788    225 academics__program_percentage_social_science
 0.12786      1 academics__program_assoc_agriculture
 0.12723    251 student__part_time_share
 0.12683     66 academics__program_bachelors_physical_science
 0.12571     59 academics__program_bachelors_mathematics
 0.12531     65 academics__program_bachelors_philosophy_religious
 0.1246     123 academics__program_certificate_lt_2_yr_education
 0.12266     55 academics__program_bachelors_humanities
 0.1216      50 academics__program_bachelors_english
 0.12095    194 academics__program_percentage_business_marketing
 0.11806    178 academics__program_certificate_lt_4_yr_personal_culinary
 0.11786     54 academics__program_bachelors_history
 0.11725     47 academics__program_bachelors_education
 0.11686     56 academics__program_bachelors_language
 0.11617     69 academics__program_bachelors_public_administration_social_service
 0.11582    224 academics__program_percentage_security_law_enforcement
 0.11114     70 academics__program_bachelors_resources
 0.11054     43 academics__program_bachelors_communication
 0.11013     85 academics__program_certificate_lt_1_yr_education
 0.1087     151 academics__program_certificate_lt_2_yr_transportation
 0.10756    246 student__demographics_dependent
 0.10756    257 student__share_independent_students
 0.10642    115 academics__program_certificate_lt_2_yr_agriculture
 0.10611    193 academics__program_percentage_biological
 0.10492     40 academics__program_bachelors_architecture
 0.10404    113 academics__program_certificate_lt_1_yr_transportation
 0.10384    174 academics__program_certificate_lt_4_yr_mechanic_repair_technology
 0.1038      82 academics__program_certificate_lt_1_yr_communications_technology
 0.10376     77 academics__program_certificate_lt_1_yr_agriculture
 0.10278     63 academics__program_bachelors_parks_recreation_fitness
 0.1011      25 academics__program_assoc_parks_recreation_fitness
 0.101      124 academics__program_certificate_lt_2_yr_engineering
 0.10029    206 academics__program_percentage_history
```

Appendix 2. Correlation Visualization the final 32 attributes

Appendix 3. ZeroR - Training and Testing Output

Model #1 (Base)	**Scheme**	weka.classifiers.rules.ZeroR				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	54.2324%		**Time Taken**	0.00 seconds	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		?	0.703	?	?	?
	ROC Curve	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.499	0.500	0.499	0.498	0.499

Model #1 (Base)	**Scheme**	weka.classifiers.rules.ZeroR				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818-				
	Attributes	32 attributes				
	Accuracy	55.8783%		**Time Taken**	0.05 seconds	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		?	0.717	?	?	?
	ROC Curve	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.500	0.500	0.500	0.500	0.500

```
=== Classifier model (full training set) ===

ZeroR predicts class value: Average

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        5843               54.2324 %
Incorrectly Classified Instances      4931               45.7676 %
Kappa statistic                          0
Mean absolute error                      0.3124
Root mean squared error                  0.3952
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances            10774

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.000    0.000    ?          0.000    ?          ?        0.499     0.209     Below Average
                 1.000    1.000    0.542      1.000    0.703      ?        0.500     0.542     Average
                 0.000    0.000    ?          0.000    ?          ?        0.499     0.180     Above Average
                 0.000    0.000    ?          0.000    ?          ?        0.498     0.067     High
Weighted Avg.    0.542    0.542    ?          0.542    ?          ?        0.499     0.375

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
    0 2258    0    0 |    a = Below Average
    0 5843    0    0 |    b = Average
    0 1947    0    0 |    c = Above Average
    0  726    0    0 |    d = High
```

```
=== Classifier model (full training set) ===

ZeroR predicts class value: Average

Time taken to build model: 0 seconds

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.05 seconds

=== Summary ===

Correctly Classified Instances        2039               55.8783 %
Incorrectly Classified Instances      1610               44.1217 %
Kappa statistic                          0
Mean absolute error                      0.3095
Root mean squared error                  0.3916
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances             3649

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.000    0.000    ?          0.000    ?          ?        0.500     0.200     Below Average
                 1.000    1.000    0.559      1.000    0.717      ?        0.500     0.559     Average
                 0.000    0.000    ?          0.000    ?          ?        0.500     0.174     Above Average
                 0.000    0.000    ?          0.000    ?          ?        0.500     0.067     High
Weighted Avg.    0.559    0.559    ?          0.559    ?          ?        0.500     0.387

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
    0  731    0    0 |    a = Below Average
    0 2039    0    0 |    b = Average
    0  635    0    0 |    c = Above Average
    0  244    0    0 |    d = High
```

Appendix 4. NaiveBayes - Training and Testing Output

Model #2 (Naïve Bayes)	**Scheme**	weka.classifiers.bayes.NaiveBayes				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	59.5576%		**Time Taken**	0.01 seconds	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.493	0.798	0.448	0.69	0.664
	ROC Curve	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.901	0.742	0.837	0.978	0.809

Model #2 (Naives Bayes)	**Scheme**	weka.classifiers.bayes.NaiveBayes				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818-				
	Attributes	32 attributes				
	Accuracy	55.6317%		**Time Taken**	0.06 seconds	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.602	0.556	0.503	0.55	0.555
	ROC Curve	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.892	0.734	0.820	0.935	0.794

```
Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        6309               58.5576 %
Incorrectly Classified Instances      4465               41.4424 %
Kappa statistic                          0.417
Mean absolute error                      0.2078
Root mean squared error                  0.4401
Relative absolute error                 66.5138 %
Root relative squared error            111.3571 %
Total Number of Instances            10774

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.851    0.232    0.493      0.851    0.625      0.525    0.901     0.757     Below Average
                 0.433    0.130    0.798      0.433    0.562      0.332    0.742     0.780     Average
                 0.669    0.182    0.448      0.669    0.537      0.422    0.837     0.479     Above Average
                 0.762    0.025    0.690      0.762    0.724      0.704    0.978     0.778     High
Weighted Avg.    0.586    0.153    0.664      0.586    0.581      0.414    0.809     0.721

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 1922  275   55    6 |    a = Below Average
 1736 2532 1475  100 |    b = Average
  206  296 1302  143 |    c = Above Average
   31   69   73  553 |    d = High
```

```
=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.06 seconds

=== Summary ===

Correctly Classified Instances        2030               55.6317 %
Incorrectly Classified Instances      1619               44.3683 %
Kappa statistic                          0.3773
Mean absolute error                      0.2217
Root mean squared error                  0.4558
Relative absolute error                 71.6081 %
Root relative squared error            116.4053 %
Total Number of Instances             3649

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.845    0.241    0.468      0.845    0.602      0.503    0.892     0.749     Below Average
                 0.421    0.120    0.816      0.421    0.556      0.330    0.734     0.780     Average
                 0.660    0.203    0.406      0.660    0.503      0.384    0.820     0.433     Above Average
                 0.549    0.032    0.551      0.549    0.550      0.518    0.935     0.595     High
Weighted Avg.    0.556    0.153    0.657      0.556    0.555      0.387    0.794     0.701

=== Confusion Matrix ===

   a   b   c   d   <-- classified as
 618  80  31   2 |   a = Below Average
 611 859 524  45 |   b = Average
  71  83 419  62 |   c = Above Average
  21  31  58 134 |   d = High
```

Appendix 5. J48 - Training Before and After Tuning

Model #3 (J48)	Scheme	weka.classifiers.trees.J48 -C 0.25 -M 2				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	79.1257%		Time Taken	0.08 seconds	
	F-Measure	Below Avg	Avg	Above Avg	High	Weighted
		0.764	0.820	0.686	0.903	0.790
	ROC Area	Below Avg	Avg	Above Avg	High	Weighted
		0.923	0.845	0.886	0.973	0.877

Model #3 (J48)	Scheme	weka.classifiers.trees.J48 -C 0.5 -M 2				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	80.7964%		Time Taken	0.07 seconds	
	F-Measure	Below Avg	Avg	Above Avg	High	Weighted
		0.793	0.834	0.704	0.911	0.807
	ROC Area	Below Avg	Avg	Above Avg	High	Weighted
		0.927	0.858	0.895	0.973	0.887

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        8525               79.1257 %
Incorrectly Classified Instances      2249               20.8743 %
Kappa statistic                          0.6641
Mean absolute error                      0.1441
Root mean squared error                  0.2884
Relative absolute error                 46.118  %
Root relative squared error             72.967  %
Total Number of Instances            10774

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.768    0.064    0.761      0.768    0.764      0.702    0.923     0.718     Below Average
                 0.829    0.227    0.812      0.829    0.820      0.603    0.845     0.837     Average
                 0.652    0.055    0.724      0.652    0.686      0.622    0.886     0.669     Above Average
                 0.938    0.010    0.870      0.938    0.903      0.896    0.973     0.860     High
Weighted Avg.    0.791    0.147    0.789      0.791    0.790      0.647    0.877     0.783

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 1734  510   11    3 |    a = Below Average
  535 4841  445   22 |    b = Average
    9  592 1269   77 |    c = Above Average
    1   16   28  681 |    d = High
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        8705               80.7964 %
Incorrectly Classified Instances      2069               19.2036 %
Kappa statistic                          0.6921
Mean absolute error                      0.12
Root mean squared error                  0.2807
Relative absolute error                 38.4239 %
Root relative squared error             71.0237 %
Total Number of Instances            10774

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.801    0.058    0.785      0.801    0.793      0.737    0.927     0.757     Below Average
                 0.837    0.202    0.831      0.837    0.834      0.635    0.858     0.843     Average
                 0.680    0.055    0.730      0.680    0.704      0.642    0.895     0.667     Above Average
                 0.942    0.009    0.883      0.942    0.911      0.905    0.973     0.897     High
Weighted Avg.    0.808    0.132    0.807      0.808    0.807      0.676    0.887     0.797

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 1808  436   11    3 |    a = Below Average
  481 4889  446   27 |    b = Average
   12  550 1324   61 |    c = Above Average
    1    9   32  684 |    d = High
```

Appendix 6. J48 - Testing Output

```
=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.05 seconds

=== Summary ===

Correctly Classified Instances        3018               82.7076 %
Incorrectly Classified Instances       631               17.2924 %
Kappa statistic                          0.7131
Mean absolute error                      0.1145
Root mean squared error                  0.2577
Relative absolute error                 36.9935 %
Root relative squared error             65.8219 %
Total Number of Instances             3649

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.792    0.039    0.835      0.792    0.813      0.768    0.967     0.856     Below Average
                 0.889    0.183    0.860      0.889    0.874      0.710    0.924     0.938     Average
                 0.811    0.069    0.712      0.811    0.758      0.706    0.954     0.763     Above Average
                 0.459    0.004    0.882      0.459    0.604      0.620    0.830     0.637     High
Weighted Avg.    0.827    0.122    0.831      0.827    0.824      0.715    0.931     0.871

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
  579  150    2    0 |    a = Below Average
  109 1812  112    6 |    b = Average
    1  110  515    9 |    c = Above Average
    4   34   94  112 |    d = High
```

Appendix 7. IBk - Training Before and Testing Output

Model	**Scheme**	weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.				
	Relation	weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 Attributes				
	Accuracy	91.5419%		**Time Taker**	4.4s	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.919	0.913	0.887	0.716	0.914
	ROC	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.992	0.967	0.977	0.78	0.961

Model	**Scheme**	weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.				
	Relation	weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 Attributes				
	Accuracy	82.5042%		**Time Taker**	0.03s	
	F-Measure	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.796	0.854	0.757	0.934	0.825
	ROC	**Below Avg**	**Avg**	**Above Avg**	**High**	**Weighted**
		0.925	0.884	0.912	0.985	0.904

```
Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        8889               82.5042 %
Incorrectly Classified Instances      1885               17.4958 %
Kappa statistic                          0.7213
Mean absolute error                      0.0933
Root mean squared error                  0.2739
Relative absolute error                 29.8692 %
Root relative squared error             69.3037 %
Total Number of Instances            10774

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.826    0.066    0.769      0.826    0.796      0.740    0.925     0.783     Below Average
                 0.838    0.172    0.852      0.838    0.845      0.665    0.884     0.875     Average
                 0.738    0.047    0.777      0.738    0.757      0.705    0.912     0.730     Above Average
                 0.950    0.006    0.919      0.950    0.934      0.930    0.985     0.949     High
Weighted Avg.    0.825    0.116    0.826      0.825    0.825      0.706    0.904     0.835

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 1864  389    5    0 |    a = Below Average
  553 4898  375   17 |    b = Average
    8  458 1437   44 |    c = Above Average
    0    3   33  690 |    d = High
```

```
Time taken to test model on supplied test set: 4.74 seconds

=== Summary ===

Correctly Classified Instances        3340               91.5319 %
Incorrectly Classified Instances       309                8.4681 %
Kappa statistic                          0.8641
Mean absolute error                      0.0473
Root mean squared error                  0.1892
Relative absolute error                 15.2723 %
Root relative squared error             48.3231 %
Total Number of Instances             3649

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.981    0.038    0.865      0.981    0.919      0.900    0.992     0.943     Below Average
                 0.913    0.029    0.975      0.913    0.943      0.879    0.967     0.971     Average
                 0.980    0.048    0.811      0.980    0.887      0.867    0.977     0.821     Above Average
                 0.570    0.001    0.965      0.570    0.716      0.729    0.780     0.618     High
Weighted Avg.    0.915    0.032    0.924      0.915    0.914      0.871    0.961     0.916

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
  717   14    0    0 |    a = Below Average
  108 1862   65    4 |    b = Average
    1   11  622    1 |    c = Above Average
    3   22   80  139 |    d = High
```

Appendix 8. SimpleLogistic - Training and Testing Output

Model #5 (SimpleLogistic)	Scheme	weka.classifiers.functions.SimpleLogistic -I 0 -M 500 -H 50 -W 0.0				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	79.0236%		Time Taken	52.96 seconds	
	F-Measure	Below Avg	Avg	Above Avg	High	Weighted
		0.772	0.827	0.667	0.853	0.788
	ROC Area	Below Avg	Avg	Above Avg	High	Weighted
		0.953	0.891	0.928	0.992	0.917

Model #5 (SimpleLogistic)	Scheme	weka.classifiers.functions.SimpleLogistic -I 0 -M 500 -H 50 -W 0.0				
	Relation	College_Income_Train_Test_NoComma- weka.filters.unsupervised.attribute.Remove-R1- weka.filters.unsupervised.attribute.NumericToNominal-R1-190,234,263- weka.filters.unsupervised.attribute.Remove-R236,240,249-250,252-253- weka.filters.unsupervised.attribute.ReplaceMissingValues- weka.filters.unsupervised.attribute.Discretize-B10-M-1.0-R191-230,235,237,242-256-precision6- weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.InfoGainAttributeEval-S weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1- weka.filters.unsupervised.attribute.Remove-R32-257- weka.filters.unsupervised.instance.RemoveRange-R10170-13818- weka.filters.supervised.instance.SMOTE-C4-K5-P500.0-S1				
	Attributes	32 attributes				
	Accuracy	75.5549%		Time Taken	53.99 seconds	
	F-Measure	Below Avg	Avg	Above Avg	High	Weighted
		0.710	0.820	0.624	0.634	0.751
	ROC Area	Below Avg	Avg	Above Avg	High	Weighted
		0.939	0.878	0.908	0.953	0.900

```
Time taken to build model: 52.96 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        8514               79.0236 %
Incorrectly Classified Instances      2260               20.9764 %
Kappa statistic                          0.6594
Mean absolute error                      0.1499
Root mean squared error                  0.2719
Relative absolute error                 47.9674 %
Root relative squared error             68.8015 %
Total Number of Instances            10774

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.763    0.056    0.782      0.763    0.772      0.713    0.953     0.846     Below Average
                 0.846    0.239    0.808      0.846    0.827      0.611    0.891     0.906     Average
                 0.622    0.054    0.718      0.622    0.667      0.602    0.928     0.735     Above Average
                 0.873    0.013    0.834      0.873    0.853      0.843    0.992     0.910     High
Weighted Avg.    0.790    0.152    0.788      0.790    0.788      0.646    0.917     0.863

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 1723  525    9    1 |    a = Below Average
  468 4946  390   39 |    b = Average
   12  638 1211   86 |    c = Above Average
    0   16   76  634 |    d = High
```

```
=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.2 seconds

=== Summary ===

Correctly Classified Instances        2757               75.5549 %
Incorrectly Classified Instances       892               24.4451 %
Kappa statistic                          0.5853
Mean absolute error                      0.1625
Root mean squared error                  0.2912
Relative absolute error                 52.511  %
Root relative squared error             74.3574 %
Total Number of Instances             3649

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.658    0.049    0.771      0.658    0.710      0.647    0.939     0.806     Below Average
                 0.862    0.304    0.782      0.862    0.820      0.569    0.878     0.899     Average
                 0.611    0.073    0.637      0.611    0.624      0.547    0.908     0.655     Above Average
                 0.537    0.011    0.775      0.537    0.634      0.625    0.953     0.701     High
Weighted Avg.    0.756    0.193    0.754      0.756    0.751      0.585    0.900     0.824
=== Confusion Matrix ===

    a    b    c    d   <-- classified as
  481  240    9    1 |    a = Below Average
  139 1757  128   15 |    b = Average
    4  221  388   22 |    c = Above Average
    0   29   84  131 |    d = High
```