RMIT UNIVERSITY

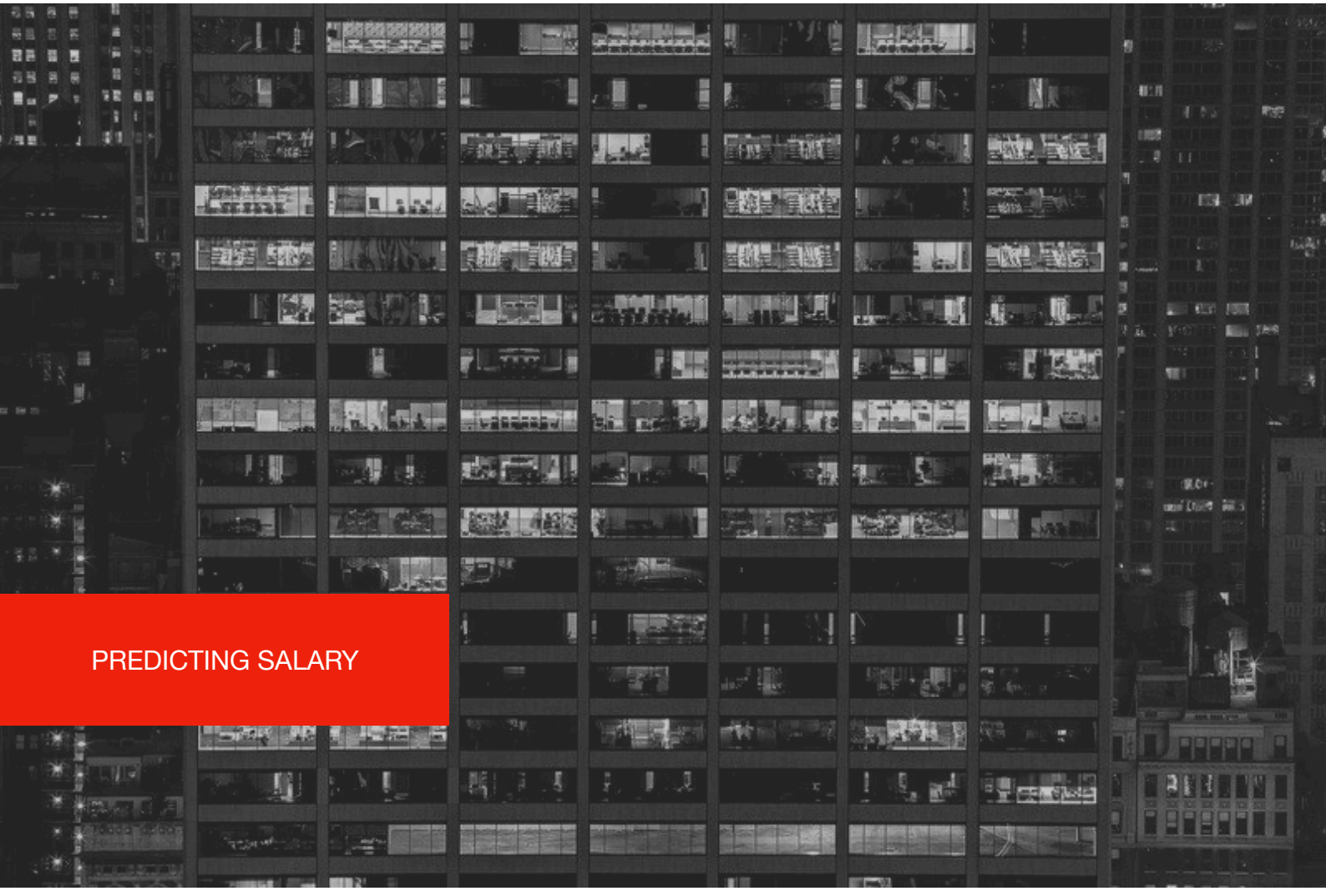
REPORT ON DATA MODELLING OF ADULT DATA SET

REPORT BY :

RAFEED SULTAN - S3763175
s3763175@student.rmit.edu.au

VISHAL BENIWAL - S3759790
s3759790@student.rmit.edu.au

Dated : 2nd June 2019



PREDICTING SALARY

Table of Content

Abstract

The aim of this report was to select a dataset and utilize it for the purpose of data modelling and presentation. The name of the dataset used is called “Adult.csv” was found in the UCI data repository provided by Ronny Kohavi and Barry Becker[1]. We cleaned the dataset for missing values and standardized the dataset so it was free from typographical and whitespace errors. Next, we explored the data according to each feature’s type to gain some insights about the dataset. As a result, we explored relationships between the features. Finally, we used the classification to model our data. The approaches for classification included comparing K-Nearest Neighbour (KNN) and Decision Tree results. In each of our algorithms, we used feature selection using the Hill Climbing approach. As a result, we were able to solve our classification problem better with Decision tree algorithm. Furthermore, we found this algorithm, faster than the KNN for producing the output.

The report concludes that we achieved a locally optimal score of 83.251% when taking a random adult to predict his annual income per year. It is recommended that we should explore other classification algorithms to train our model. By doing so, we might be able to achieve higher classification rates. Additionally, we could combine K-cross validation technique in our feature selection and tune the values of k to achieve a higher accuracy model for classification if possible.

It is recommended that resampling of the data must be done to overcome the existing problem of bias data and the unbalanced ratio of the class labels. The resampling should focus on the people from different continents, race and education level and it should also focus more on the adults with a salary greater than 50k per year to gain the balance in the proportion of class labels.

Introduction

The goal of most human beings is to live a comfortable lifestyle. However, the most essential component to maintaining such a lifestyle is money. The earnings of an adult dictate what kinds of luxuries and necessities he or she can afford. Even if we believe all kinds of human beings are equal in moral value, however not all human beings are equal in terms of monetary valuation. The distinction between rich and poor all stems from their monetary valuation.

Recently, there has been an increasing need to find what are the contributing factors that affect the annual salary of an average adult. The highlighted factors that contribute to the annual income of an adult includes age, race, working sectors, nationality, and other factors.

This report will discuss research into an adult's annual income depending on the defining attributes of an adult. Moreover, the report will showcase the analysis on how the different classification models predict the average per annum salary of a random adult depending on his or her features.

Methodology

The dataset used by us is called "Adult Data Set". It is also known as the "Census Data Set" created by Ronny Kohavi and Barry Becker [1]. We collected this open dataset from the UCI data repository for the purpose of this assignment. The dataset contains 14 features and 48842 observations. The features contained the following features: age, workclass, education, edu-num, marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, and native country. A target variable is also available with the two labels: "Greater than 50k" & "Lesser or equal to 50K". Basically, the target variable is dividing the adults into two above mentioned groups based on their annual income. Therefore, we set the main goal of the project is to predict whether the adult income is greater than $50K/year or not, based on a given adult's features. The methodology is divided into 3 sections: Data Cleaning, Data Exploration, Data Modelling.

To start with the data cleaning process, we loaded two sets of dataset "adult.csv" and "adult_test.csv" using the pandas libraries on python as UCI repository have a pre-divided dataset for train and test. Therefore, we had to combine those separated datasets together to have a larger dataset for our project. To combine two datasets we used the pd.concat() function. To validate if the loaded data is in a proper state, we used .head() and .tail() function on the loaded data and checked the sample of the complete dataset.

During the validation, it was found that the first row of the "adult_test.csv" data frame had an inappropriate value which had to be removed, so, we skipped the first row while loading the "adult_test.csv". For the rest of the cleaning, we created a process. Anything that is not a part of the process will be treated as a special kind of case and will be handled accordingly.

Following the below process, the attributes were inspected. The different types of issues are categorized below :

[1] **Whitespaces:** We used the str.strip() function of pandas to remove all the leading and trailing whitespaces. All the attributes containing categorical values like "Education", "MartialStatus", "Relationship", "Race", "Sex", "Salary" had an issue to whitespaces which were removed using str.strip() function.

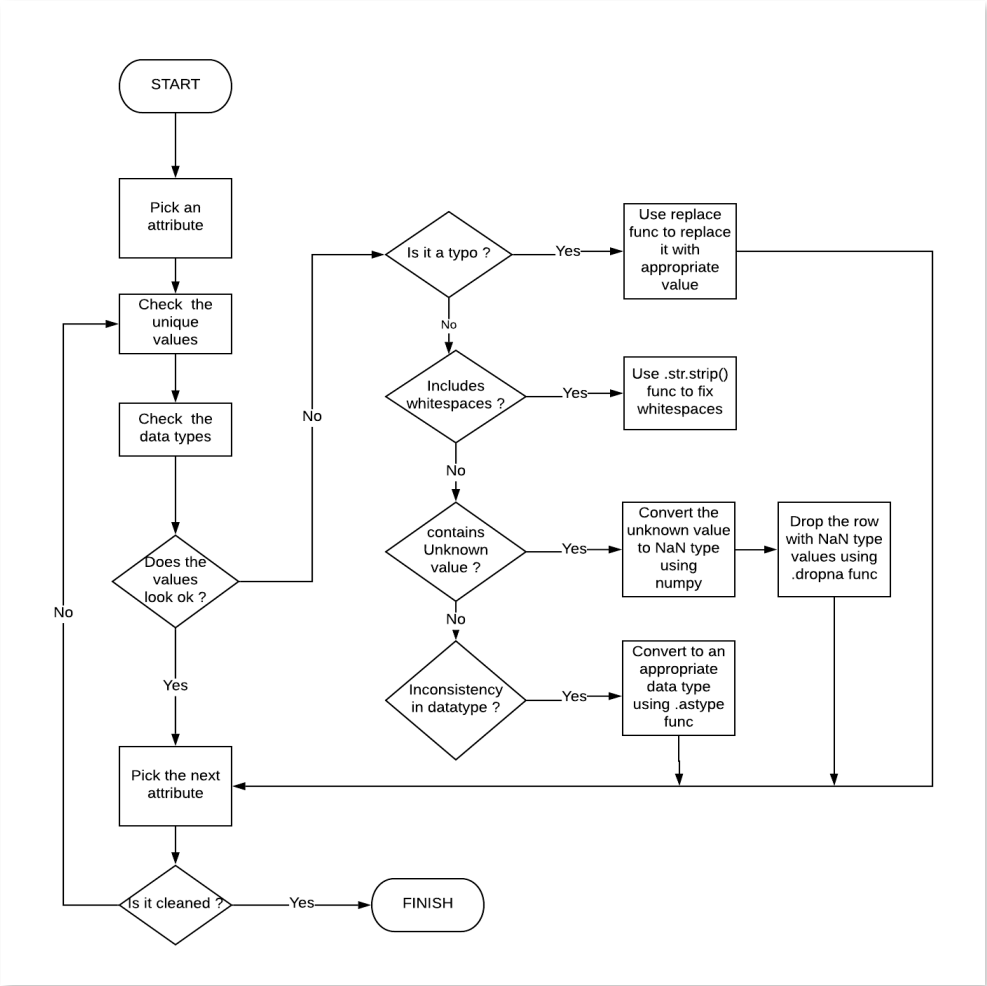
[2] **Case Sensitivity:** Only WorkClass attribute had values with which had the issue with case sensitivity. To fix the issue, str.lower( ) function was used to convert each value to lower case which ensured consistency.

[3] **Typos:** Attribute "NativeCountry" had a value "South" which doesn't indicate any real-world country and can't be derived by any of the existing values. Therefore, all the rows with this typo were dropped. Another case of typo was found in the attribute with the target values itself. "Salary" feature had values "<=50K." , so we chose to replace them with a standard value "<=50K" using the replace function.

[4] **Unknown/Missing Values:** We also encountered the case where some of the features contained '?' values which were unknown to us and was inappropriate to keep them in the data. Below is the count of such values :

Feature Name	Data Type	Unknown Value (?)
WorkClass	Categorical (str)	2799
Occupation	Categorical (str)	2809
NativeCountry	Categorical (str)	857

Therefore, in order to remove them in one go, we decided to convert them into NaN type value using numpy and replace function. After we replaced the values with the NaN type, it was straightforward to drop the observation with Null values using. dropna() function.



Data Exploration

Figure [1]

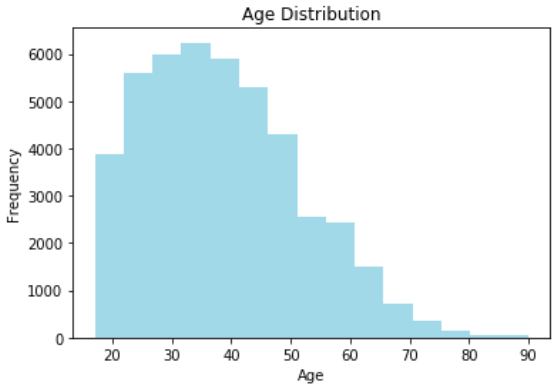


Figure [2]

According to the **figure 1**, we have found that out that 75.19 % of the adults earn less than or equal to 50K per year, whereas we can see that 24.81% of the adults earn greater than 50k per year. This pie chart can be interpreted as the 3/4th majority of the adults in the world don't earn as much as 1/4th of the adults of the world, with 50K being a benchmark. This was very surprising since there are a lot of developing countries in the world with a large population, for example, Bangladesh, where the average income of an adult is around $1K year. From this, we can see that one of our class label(<=50K) has more support in this dataset when compared to the other one(>50K).

Figure 2 is a histogram of an adult's age in the dataset. Since the right tail of the histogram is greater than the left of the histogram, we can conclude the histogram of the dataset is positively skewed. The mode of the histogram is around 35 years. The majority of the sample lies in the range of 25 years to 50 years. Therefore, we can see that teenagers and retired individuals are a minority in our dataset which is a positive sign for the purpose of predicting salary.

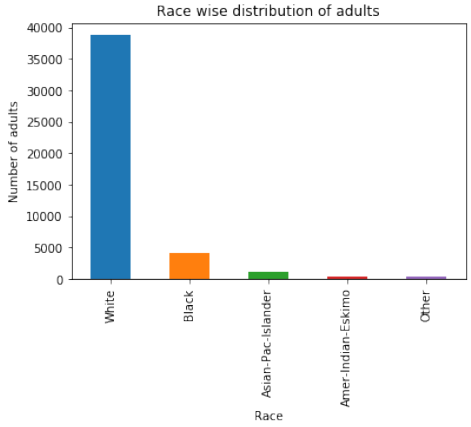


Figure [3]

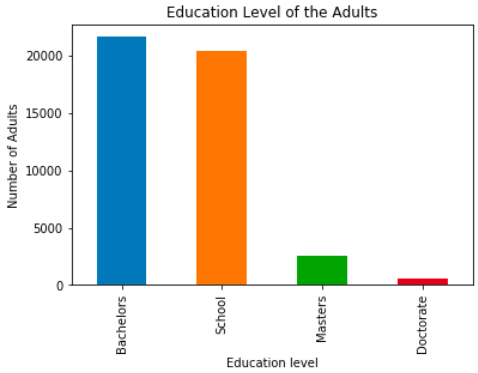


Figure [4]

Figure 3 displays the race-wise distribution of adults. We can see that the dataset is biased since the white population is 8 times greater than that of the black population. From this graph, we can see that Asian-pac-islander, Amer-Indian-, and other races are not representative of the world's population. Therefore, we can say that sampling of the data was not randomized in terms of race and was focus to the included white population.

According to the **figure 4**, we can see that most of the adults in this data set have completed their education between high School and bachelors and only a small proportion of adults went to pursue masters or doctorate degrees.

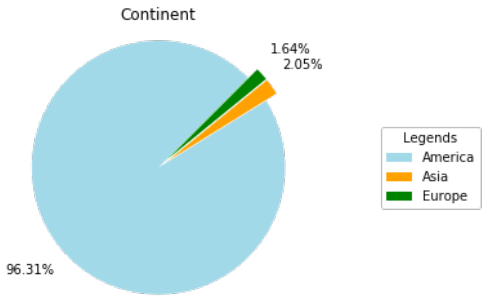


Figure [5]

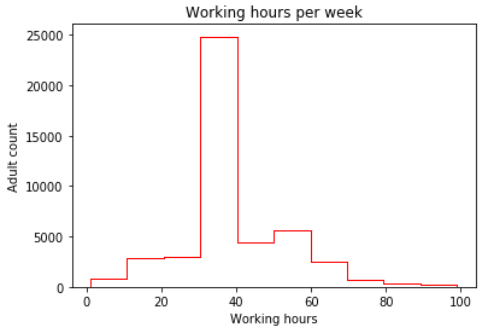


Figure [6]

According to **figure 5**, we can see that the majority of adults are from America which is around 96.3%. Asian and Europeans make up about 1.64% and 2.04% respectively. From this graph, we can see the dataset is biased towards adults from America since the rest of 3.68% of the adults are not representative of the entire world population.

According to **figure 6**, we can see that the mode working hours per week is around the range of 30-40 hours. From this graph, we can see that most full working in this dataset usually has full-time 40 hours per week jobs. There is a small proportion of adults who are working less than 30-40 hours or more than that.

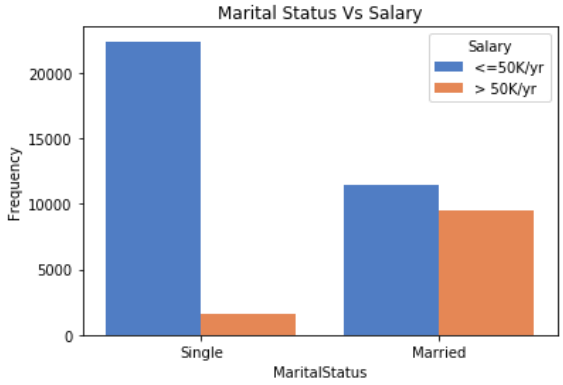


Figure [7]

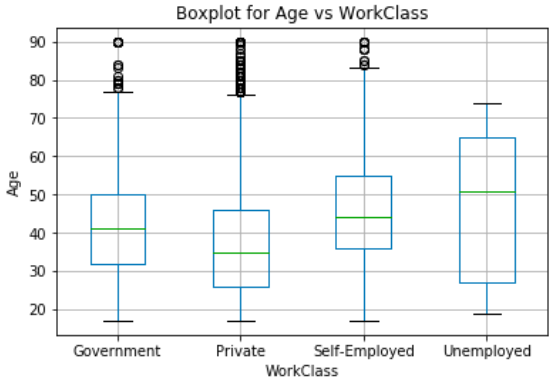


Figure [8]

The first research question was ***what was the median age of adults in each work class***. From the boxplot in **figure 8**, we can see that the unemployed work class had the highest age, whereas the private working class employed comparatively young adults. We can make a strong assumption using this graph that the private sector hires young adults since they are fitter and can work long hours. Whereas as the age of an adult increase his value in the workforce starts to decrease and therefore becomes unemployed or they retire.

The research question was ***how does the marital status affect the salary of an adult***. From **figure 7**, we can see that as the marital status changes from single to married, we can see that the number of adults earning salary >=50K per year is increasing. For this, we can make a strong assumption that as your marital status changes to married, you tend to gain more responsibilities to take care of your family and parents. Since the expenses of a married person are more, therefore the married adult faces the necessity to earn more money.

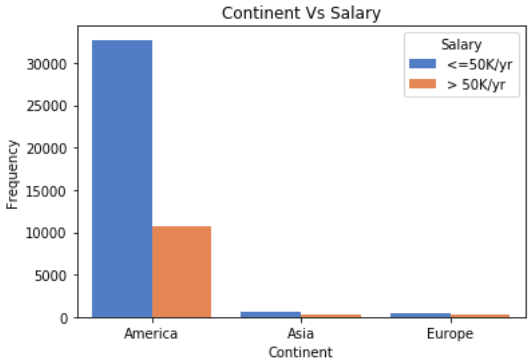
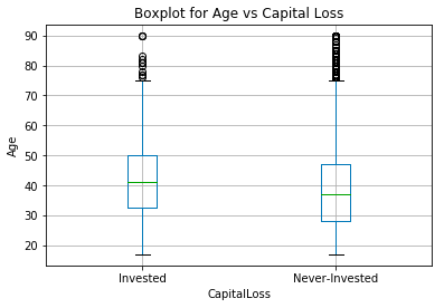


Figure [9]

Figure [10]

The research question was that ***adults of which continent was earning the most*** were addressed by **figure 9** below. In this data, America means both North American and South America. From our observation, we can see the number of American adults is very large compared to the rest of the continents. This is an indication of the fact that the data is biased and contains mostly American adults. For the purposes of fairness, we are just comparing American adults. The number of American earning <=50k per year is roughly 3 times than those of American adults earning greater than 50K per year.

The next research question was ***what gender earned the most*** which was addressed in **figure 10** below. We can see that the number of males and females are not approximately equal in this dataset. For a fair assessment, we will compare the number of male and their corresponding salaries. In male gender, we can see that the male gender earning <=50k per year is 2 times than that of the male adults earning >50k per year. In the female gender, we can see that the female adults earning less than 50k per year is roughly 7 times than that of female adults earning greater than 50k per year. Therefore we can claim that the probability of male adults earning greater than 50K per year is more than the probability of female adults.



Figure[11]

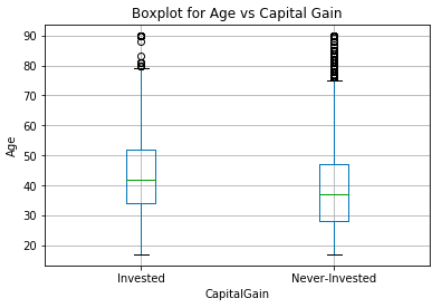


Figure [12]

The research question was that ***what was the age group that incurred a capital loss by choosing to invest or not*** addressed by the boxplot in **figure 11**. There are a few outliers above the upper fence in both box plots according to the boxplot criteria of the upper fence(i.e. upper fence = 1.5*IQR+Q3). The median age of the investment group was around 41 which was less than the median age of the never-invested group that incurred capital loss As a result, we can interpret that as 50% of the adults that invested and had capital loss had an age greater than that of those compared to the never-invested group. It may be the case that an older age you invest more and become prone to incurring a loss as compared those when you are young.

Another research question was that ***what was the age group that incurred capital gain by choosing to invest or not*** addressed by the boxplot in **figure 12** .There are a few outliers above the upper fence in both box plots according to the boxplot criteria of the upper fence(i.e. upper fence = 1.5*IQR+Q3). The median age of the investment group was around 41 which was less than the median age of the never-invested group that incurred capital gain. As a result, we can interpret that as 50% of the adults that invested and had capital gain had an age greater than that of those compared to the never-invested group. It may be the case that at an older age you also invest more to gain profit compared to those when you are young.

Figure [13]

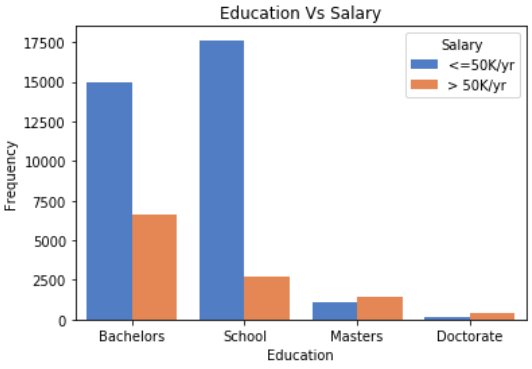


Figure [14]

The research question was that ***what Work Class provided the highest salary*** was addressed in **Figure 13**. From the graph, we can see most people work in the private sector, and this may be due to the fact that the amount of private sector jobs are more than other sector and also people are getting high-end salaries in the private sectors. We also see that the proportion of adults earning <=50K/year is roughly 3 times than that of the people earning >50K/year. Also, government and self-employed have identical proportion for <=50k and >50k.

The research question was ***what does the higher education level confirms high paying jobs*** which was addresses by **Figure 14**, we can see what proportion of adults with bachelors degree earning <=50K per year is roughly 2 times that of the proportion of adults with bachelors degree earning >50K per year. We can make a strong assumption that graduate students earning >50K per year can because of their years of experience or they are exceptionally good in their respective field. On the other hand, it can be seen that the population of adults who are earned

masters and doctorate degrees are very few and far between. As a result, there is a high probability that adults earning <50k are much higher than the adults earning >50K per year.

Data Modelling

Since the dataset has class labels, we choose to perform a supervised machine learning technique for which classification approach was picked to model our data. For classification, we have used both K-Nearest Neighbour (KNN) and Decision Tree classification models. The class label of the dataset is a binary feature where 0 means that the adult is earning less than or equal to 50k per annum and 1 indicates the earning more than 50k per year. Classification technique learns the pattern inside the dataset using the split of the dataset which we call as train data. To know whether the model's learning is accurate or not, another split of the dataset, test data, will be used for testing the accuracy of the model. Also, the model generates the confusion matrix at the end to show the count of correctly labeled and miss labeled of the class label.

To make the data modelling possible few of the changes were required in the original dataset.

- All the categorical data were converted to numerical using manual mapping. There are other available techniques also which could be used to avoid manual mapping such as oneHot [8] encoding which automatically deals with the categorical data. However, as we had only a few unique values in our categorical attributes, we decided to do it manually.
- We dropped two columns [Education & FinalWeight] from the dataset beforehand. The Education was already present in the dataset in the form of numerical values inside EduNum feature. So, due to duplicity, we dropped the education attribute. Another one was Final Weight which had continuous, random values which were trivial to the models and had no pattern or meaning inside.
- Then we split the dataset into two dummy data frames: data and target. We also used flatten() on the target data frame to make it a 1D array as it was required for the modelling.
- We loaded all the required libraries to run the models

The process of data modelling includes four major steps which needs to followed.

The first step is feature engineering and model selection :

Feature Selection

For our modelling process, we used the hill climbing approach for feature selection. We shuffled the order of the features so that the features were randomly selected based on random state. By did this exercise so that we could easily observe and identify which of the features were contributing the most in the performance of the model. We used our models with the hill climbing itself to get the important features along with the model accuracy.

Model Selection : K-Nearest Neighbour (KNN)

K-nearest neighbour algorithm is a powerful and efficient technique to solve predictive problems. KNN algorithm makes a prediction based on an idea that similar things exist near to each other. To know how two things are near or similar to each other, it uses distance to calculate the proximity [9]. To use this, we will import required libraries from sklearn available in python.

Model Selection : Decision Tree

The decision tree is another great technique to solve the classification problem. The decision tree can be visualised as a tree-like structure which works for both continuous and categorical attributes. This method is useful as the visual result of this algorithm is easy to understand which reduces the human effort to make decisions. The tree structure starts with the root node which is the topmost node of the tree which represents then the research question. Then comes the branch nodes which represents the result of that test and in the end, we get a leaf node with the outcome of the test. To use this, we will import required libraries from sklearn available in python.

The second step is to train the selected model :

Train Model : K-Nearest Neighbour (KNN)

- The dataset with the data(mentioned above) was divided into two categories: training data and testing data using sklearn. We tested our model with different split ratios. Ratios were (1) 50%

for training and 50% for testing (2) 60% for training and 40% for testing (3) 80% for training and 20% for testing. The performance of the model was later compared to find the sufficient data required for train and test to make the prediction.

- The next step was to select the right K value for the KNeighborsClassifier. K value in KNN means the number of neighbours. We started with K = 1, which resulted in low accuracy and appeared less stable. Then, we gradually increased the K value and observed an increase in the accuracy because of averaging among the data points.
- Another tuning parameter we used was the p-parameter. We found that the small values for p-parameter of the Minkowski metric gave the best results. This was because at smaller values of p, for low dimensional data, the contribution of each of the feature was more compared to higher values for p. Therefore, we kept p = 1, that is known as Manhattan distance. In our KNN model, we chose to weighted distance than uniform distance, since this decision yielded better results for this data set.
- The model was then trained on the train data with the class labels using the fit() function.
- The learned model was then applied to the test data to make the prediction of the class labels using predict() function. To check the performance of the prediction, we generated a confusion matrix.
- We did the same practice inside the hill climbing algorithm to shuffle the order of the features. The random state parameter of the hill climbing was manipulated from 0 to 10 to find the set of features highly contributed to the greatest possible weighted value for precision, recall and F1-Score and minimum possible value for error rate.

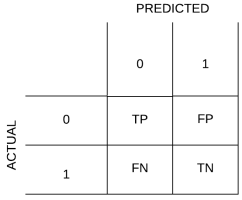
Train Model : Decision Tree

- The decision tree was also trained on the above-mentioned data splits using K-cross validation technique.
- The decision tree was defined using DecisionTreeClassfier() from sklearn. To tune the parameters, “GINI Index” was used to produce split points as it is known to be very effective for the binary split.
- We kept all the other parameters as default (max_depth=None, min_samples_split=2, min_samples_leaf=1) while tuning to avoid over-fitting or under-fitting problems.
- Again, the model was trained on the train data using the fit() function. The learned model was then applied to the test data to make the prediction of the class labels using predict() function. To check the performance of the prediction, we generated a confusion matrix.
- Though hill climbing is not that required for the decision tree, we still used it to keep the consistency in the training process for both the models.

Results

Model Evaluation :

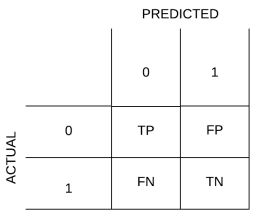
CONFUSION MATRIX



KNN 50% : 50%	PREDICTED 0	PREDICTED 1
ACTUAL 0	15635	1344
ACTUAL 1	2472	3073

KNN 60% : 40%	PREDICTED 0	PREDICTED 1
ACTUAL 0	12453	1135
ACTUAL 1	1980	2452

KNN 80% : 20%	PREDICTED 0	PREDICTED 1
ACTUAL 0	6257	561
ACTUAL 1	982	1210



Decision Tree 50% : 50%	PREDICTED 0	PREDICTED 1
ACTUAL 0	15680	1299
ACTUAL 1	2480	3065

Decision Tree 60% : 40%	PREDICTED 0	PREDICTED 1
ACTUAL 0	12544	1135
ACTUAL 1	1980	2452

Decision Tree 80% : 20%	PREDICTED 0	PREDICTED 1
ACTUAL 0	6290	528
ACTUAL 1	981	1211

In the Confusion matrix above, the basic metrics are above :

- 0 means that the adult is earning less than or equal to 50k per annum and 1 indicates the earning more than 50k per year.
- True Positive (TP) : The adults who were earning <=50K/year and correctly classified as <=50K/year.
- True Negative (TN) : The adults who were earning > 50K/year and correctly classified as > 50K/year.
- False Positive (FP) : The adults who were earning <=50K/year and miss classified as > 50K/year.
- False Negative (FN) : The adults who were earning >50K/year and miss classified as <=50K/year.

Model Evaluation & Comparison Table

Metrics		KNN			Decision tree			
Train : Test Split		50% : 50%	60% : 40%	80% : 20%	50% : 50%	60% : 40%	80% : 20%	Support
F1- Score	<=50K : 0	0.89	0.89	0.89	0.89	0.89	0.89	16979
	>50K : 1	0.62	0.61	0.61	0.62	0.62	0.62	5545
	Weighted	0.82	0.82	0.82	0.83	0.82	0.83	22524
Precision	<=50K : 0	0.86	0.86	0.86	0.86	0.86	0.87	13588
	>50K : 1	0.70	0.68	0.68	0.70	0.70	0.70	4432
	Weighted	0.82	0.82	0.82	0.82	0.82	0.82	18020
Recall	<=50K : 0	0.92	0.92	0.92	0.92	0.92	0.92	6818
	>50K : 1	0.55	0.55	0.55	0.55	0.55	0.55	2192
	Weighted	0.83	0.83	0.83	0.83	0.83	0.83	9010
Accuracy		0.83058	0.82713	0.82874	0.83222	0.83140	0.83251	
Error Rate		0.17	0.17	0.17	0.17	0.17	0.17	

Classification Accuracy : It is the percentage of correctly classified labels. Since the false positives and false negatives of the confusion metrics are not balanced. This is not particularly useful to our main hypothesis.
Classification Error : It is the percentage of the misclassified labels in the dataset. Since the false positives and false negatives of the confusion metrics are not balanced. This is not particularly useful to our main hypothesis.

We identified the lack of support for one of our class label (>50k : 1). So, to minimise the impact of the support in each class, we took the weighted f1-score, precision, recall and found that they were similar for all splits for a particular model, which showed that both models KNN and DT had similar performances regardless of the splits. Therefore, we decided to check the results class wise using f1-scores, precision, recall regardless of the support.

KNN Model Outcome :

For KNN, when we started our investigation class-wise, we found that **50:50** train and test split produced the best results, i.e. **50:50 split** had the highest F1-Score, Precision, and Recall.

- The f1-score was 0.89 for class label <=50K per year and 0.63 for class label >50K per year.

- The precision for <=50K per year was 0.86 and the precision for >50K per year was 0.76 respectively.
- In contrast, the recall for <=50K per year was 0.92 and the recall for >50K per year was 0.55 respectively.

Decision Tree Model Outcome :

Using the aforementioned Model Evaluation & Comparison Table, we found that 80:20 train and test split produced the best results for the decision tree model. 80:20 split had the highest F1-Score, Precision, and Recall.

- The f1-score from the 80:20 split which was 0.89 for F1-Score of class label <50K per year and 0.63 for class label >50K per year.
- The precision for <=50K per year was 0.87 and the precision for >50K per year was 0.76 respectively.
- In contrast, the recall for <=50K per year was 0.92 and the recall for >50K per year was 0.55 respectively.

Feature Selection Outcome :

The most important features we got while using the hill climbing algorithm with KNN and decision tree are : Sex, Relationship, Education, Capital Loss, Capital Gain, Marital Status, Work Class, Occupation. These features contributed the highest to the accuracy of our models.

Discussion

Based on the above Model Evaluation & Comparison Table, we realised that the adult dataset was imbalanced. Dataset imbalanced means one of the class labels is suffering from the lack of support and therefore accuracy for that particular class label gets affected. To overcome this situation, we had to change our priorities. F1- Score became our first priority as F1-Score is the harmonic mean of precision and recall, it captures both the false positives and the false negatives [3]. Therefore, f1-score is a better metric for predicting annual salary than the other metrics as we are not focusing on a particular class label [4]. The main objective for evaluating both models was trying to achieve the highest f1- scores as both the class labels were equally important [5].

Looking at the results in the previous section we observed that if we increase the train and test split ratio, we are getting lower values for the precision of class label >50K and therefore a lower value for the F1-Score. If we decrease the train and test split ratio, our KNN model starts suffering from under fitting since it has a trouble modelling the training dataset and it cannot generalize the new test data [6]. If we increase the split the from 50:50 to 60:40 and 80:20, we also run the risk of overfitting our data since the values for precision for class label >50K per year decreases from 0.62 to 0.61. Since the model has more train data to work with, it may start to suffer from generalization error i.e. it may start to memorize its outcomes rather than learn from the data. We can assume that one of the possible reasons for the 0.01% drop in precision for class label >50k per year maybe because we are introducing more train data that has a chance of being noisy.

Another fact that we discovered was that the general rule of thumb is that if the cost of training a model is low, then, higher split for train data is better because it takes fewer resources like computation time to train the model. Generally, more training data helps the model to learn new relationships. Since the decision tree is faster than KNN, higher split ratio is preferable.

Conclusion

After comparing, we found that the **Decision Tree** performed the best out of the both models. Since the accuracy of Decision Tree was 0.83251%, whereas for KNN the accuracy was 0.83058. There is an accuracy difference of 0.00193% between the two models. The factor with the highest priority, weighted f1-score, was found better with a Decision tree (0.83%) which was more than the weighted f1-score of KNN (0.82%). Therefore, we would recommend the decision tree as the

ideal model to solve the classification problem of predicting salary using the adult dataset from UCI repository. Also, it's not only the accuracy that was better, but even the time and speed of the computation was also faster with the Decision tree.

However, the dataset was found biased for few features and in order to overcome this problem, resampling of adult salary dataset should be done to include more observation of individuals with higher education (masters and doctorate), individuals of the Asians and Black race and more individuals from continents like Europe, Asia, Australia. Furthermore, the dataset was found unbalanced in terms of the support given to each class label. Therefore, including more instances where the adult is earning more than 50K salary per annum would help the model to learn better and in result, will provide better accuracy [7]

If we look to change anything in the technique that we used then, K-Fold Cross Validation for splitting can be used as it splits the data in a way that reduces biases towards train data. We can also use and compare with other classification models to see increase the accuracy of the data.

References

[1] Becker, B. and Kohavi,R. 2017, *Adult Data Set*, electronic dataset, UCI Machine Learning Repository, viewed 27 May 2019, <https://archive.ics.uci.edu/ml/datasets/adult

[2] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science

[3] Towards Data Science. (2019). *Accuracy, Precision, Recall or F1?*. [online] Available at: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9 [Accessed 2 Jun. 2019].

[4] Exsilio Blog. (2019). *Accuracy, Precision, Recall & F1 Score: Interpretation of Performance Measures - Exsilio Blog*. [online] Available at: https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/ [Accessed 2 Jun. 2019].

[5] Brownlee, J. (2019). *Classification Accuracy is Not Enough: More Performance Measures You Can Use*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/ [Accessed 2 Jun. 2019].

[6] Brownlee, J. (2019). *Overfitting and Underfitting With Machine Learning Algorithms*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/ [Accessed 2 Jun. 2019].

[7] Brownlee, J. (2019). *8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/ [Accessed 2 Jun. 2019].

[8] What is One Hot Encoding and How to Do It. (2019). Retrieved from https://medium.com/@michaeldelsole/what-is-one-hot-encoding-and-how-to-do-it-f0ae272f1179

[9] Machine Learning Basics with the K-Nearest Neighbors Algorithm. (2019). Retrieved from https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761