STAT 5703 – Final Project
Analysis of Audit Data

Jiaming Han 100631020

Yuhua Cong 100980213

Jingyi Chen 101108268

Dec.21 2018

Catalogue

1 Introduction and Data preliminary processing

1.1 Introduction

The dataset of Audit Data has 2000 observations and 12 columns, with ID, 9 explanation variables: Age, Employment, Education, Marital, Occupation, Income, Gender, Deductions and Hours, 2 response variables: RISK_Adjusement and TARGET_Adjusted. The goal of this analysis is to show the features of 2000 individuals and to make productive audits decision by the variables above in the future.

1.2 Variable Redefine

1. **Employment:** Original variable has 8 levels and 100 missing records. There is only one observation in levels of "volunteer" and "unemployed". Redefined "Employment" variable has 4 levels: "Private", "Consultant", "Government" and "Self-employed/others". The new "Government" level bundles old levels of "PSLocal", "PSState" and "PSLocal" together. The new "Self-employed/others" level has old levels of "SelfEmp", "Unemployed", "Volunteer" and 100 missing observations.
2. **Education:** Original variable has 16 levels and new "Education" has 4 levels: "Pre-High School", "High School Graduates", "College", "Bachelor", All "Year-12" and prior levels of Education in original variable are bundled in "Pre-high School" level. Old level of "Professional", "Associate", "Vocational" and "College" are grouped into new level of "College". Old levels of "Doctorate", "Master" and are grouped in new level of "Bachelor"
3. **Marital:** Original variable has 6 levels and new "Marital" variable has 5 levels which combined "Married-spouse-absent" level into "Married" level.
4. **Occupation:** Original variable has 14 levels and 101 missing values. New "Occupation" has 6 levels including "White-Collar", "Blue-Collar", "Service", "Professional" and "Other/Unknown". New "White-Collar" includes old levels of "Clerical", "Executive". New "Blue-Collar" includes old levels of "Cleaner"," Farming"," Machinist" and "Repair". New "Service" level includes old levels of "Sales", "Support"," Transport" and "Protective". All other old levels are grouped into new level of "Other/Unknown".

1.3 Scale data

In this data set, variables are not in the same scale. So we scale and center the data, with formula: center = min, scale = max-min to make them in the same scale.

2 Data visualization

Data visualization is usually the first step to preliminarily explore the relationships between data. In this section, we use packages, such as ggplot2, plyr, ROCR and corrgram to present the data.

2.1 Relationship between response and explanatory variables

First we consider variable “TARGET_Adjusted” as the response variable and create plots to explore the relationship between response variable and explanatory variables.

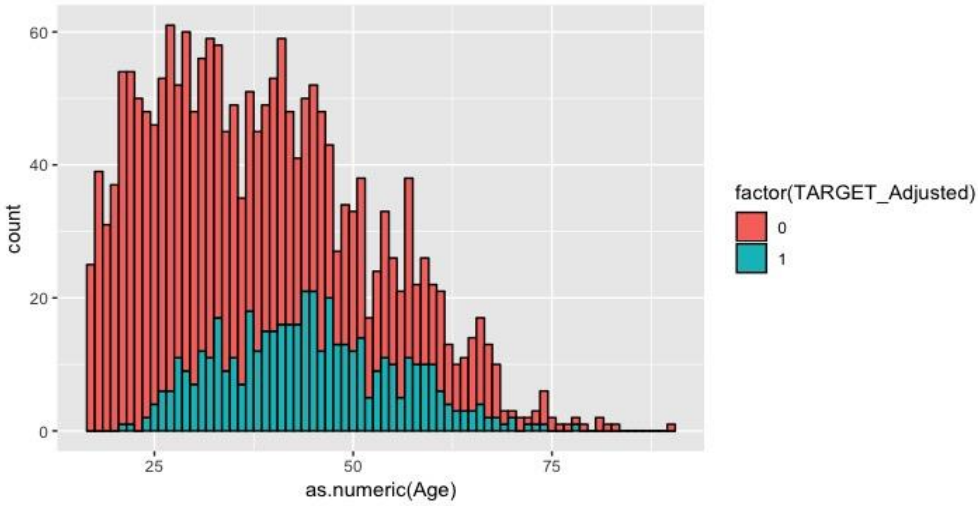
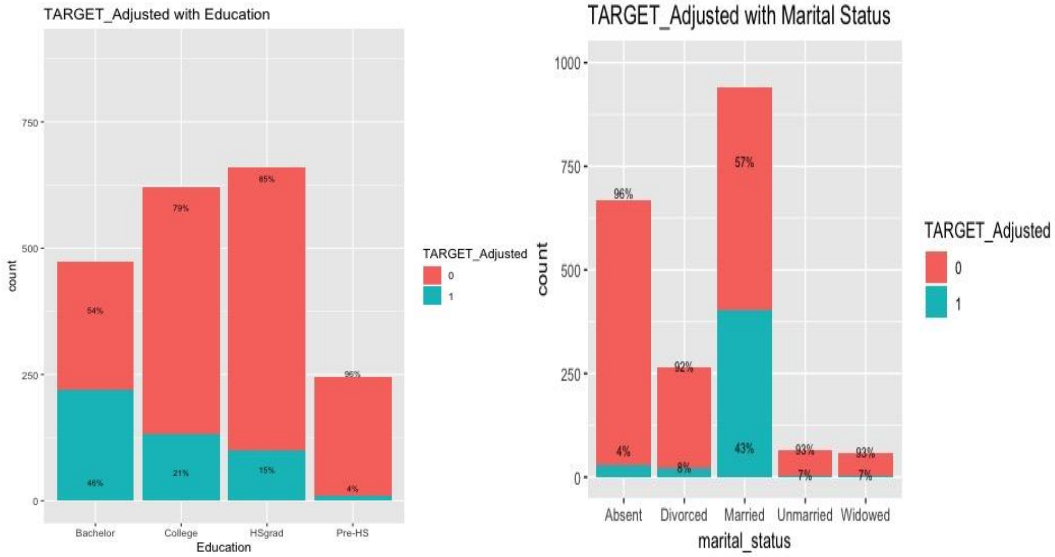


Figure 1 Histogram of age

Looking at the histogram of age by TARGET_Adjusted group above, it is skewed right. It is noticed that majority of the observations have nonproductive audit(TARGET_Adjusted=0). There are only few observations after sixty years old.

Bar plots shown below provide more straightforward and useful information.



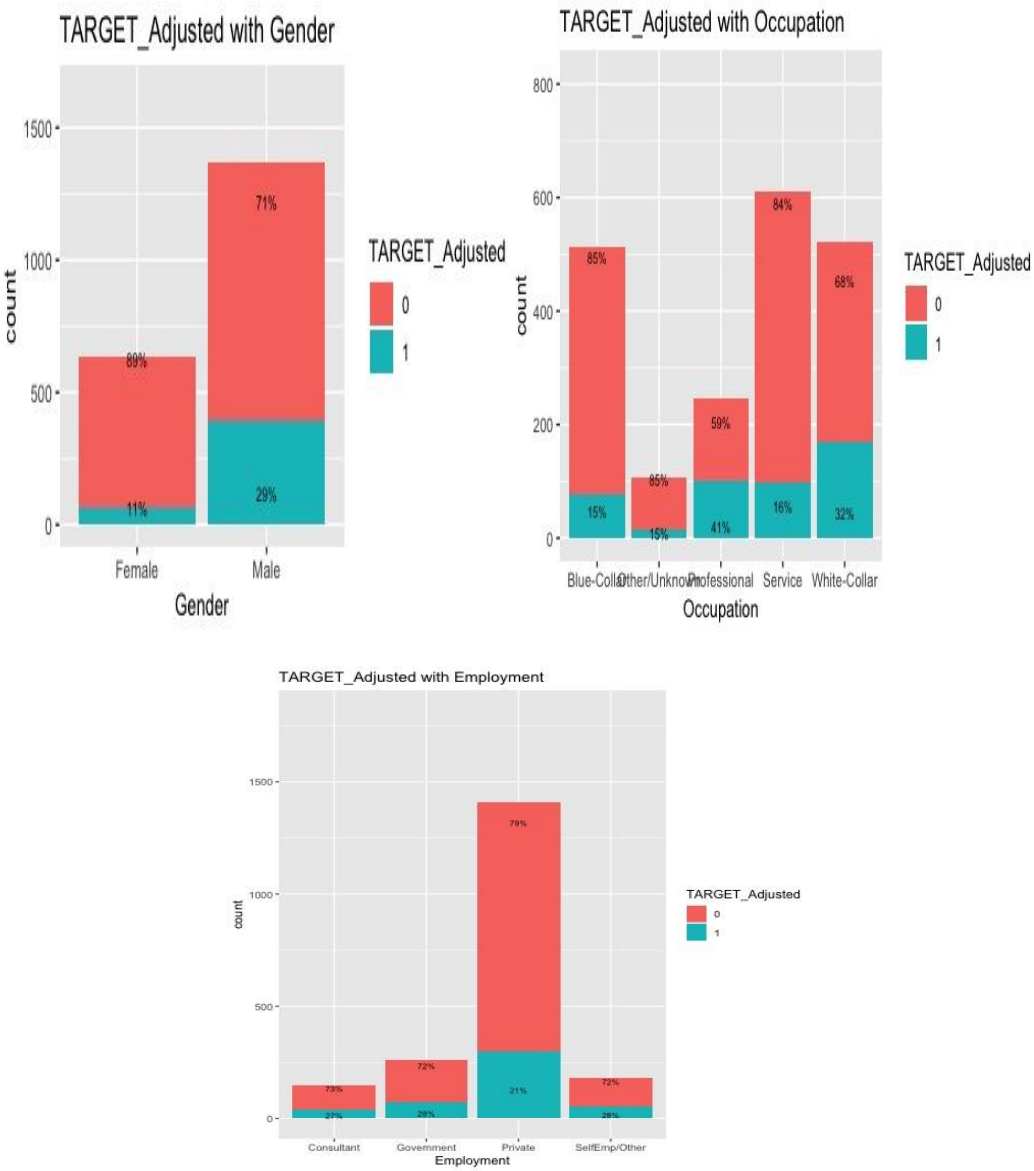


Figure 2-6 Bar plots between categorical variables

It is not hard to notice that the in group proportion of having productive audit increase as the level of education increases. For those who don't have any forms of college education, less than 15% have productive audit. While for those with doctorate, master and bachelor's degree, nearly 50% has productive audit. For those who are married, nearly half of them have productive audit. While for those who are not married, less than 8% has productive audit. Percent of male have productive audit is two times greater than the percent of female, 29% and 11% respectively. It is noticed that having productive audit or not varies greatly across different occupations. 41% of professional occupation have productive audit, while percentage is only 15% for blue-collar and service occupation. Having productive audit or not doesn't seem vary across employment, the percentage who has productive audit is around 20% for each employment type.

2.2 Relationship between each variable

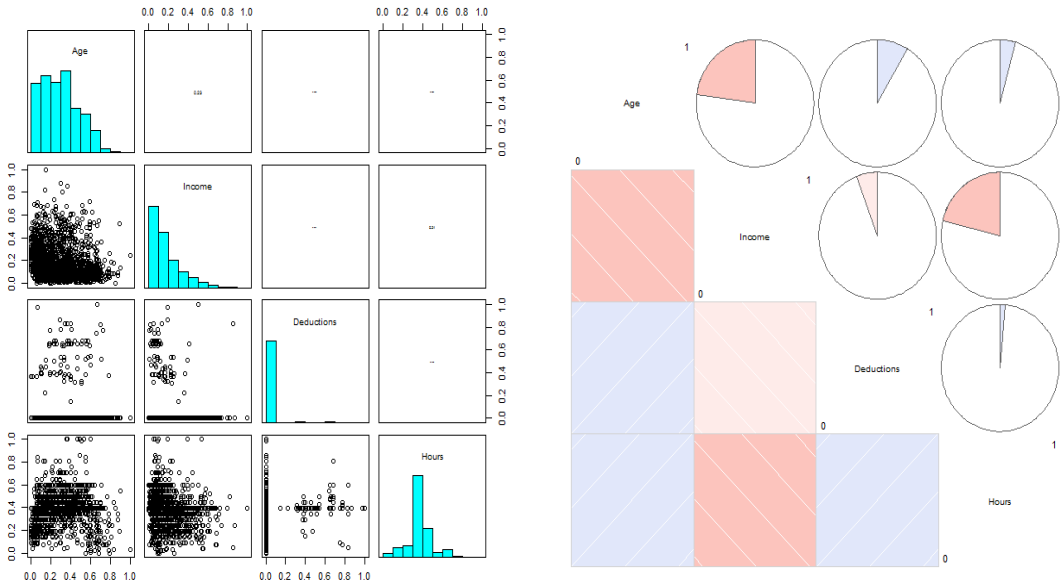


Figure 7-8 Histogram and correlogram of continuous variables

The above two plots show the distribution and correlation among four continuous variables. First, variables age, income and deductions are all skewed right, and histogram of hour is approximately symmetry. Then, from the second plot, we can see color pale blue indicates weak positive correlation and pale red indicates weak negative correlation. That means, all variables have weak correlation with each other, and detailed coefficients are also shown in Table 1.

Table 1 Coefficient among continuous variables

	Age	Income	Deductions	Hours
Age	1.00	-0.23	0.08	0.04
Income	-0.23	1.00	-0.06	-0.21
Deductions	0.08	-0.06	1.00	0.01
Hours	0.04	-0.21	0.01	1.00

Data Visualization Using Ggobi

1. Ggobi is used to perform the scatter plot matrix (“SactterPlot_Varibles_PC1_PC9.jpg”) using “Sphering(PCA)” function without two target variables (“RISK_Adjustment” and “TARGET_Adjusted”). The matrix also includes principle components PC1 to PC9. From the scatter plot matrix, it seems the variables do not have any strong correlations with each other since no scatter plot shows an obvious trend. However, based on scatter plot matrix, “Age” and “Income” are slightly correlated with PC1. “Hours” is somehow related with PC2 and PC3. “Deduction” seems related with PC4 and PC5. “Hours” is also slightly related with PC6. “Income” seems highly related with PC8. “Age” seems also slightly related with PC9.

2. From the "ScatterPlot_Matrix_Correlation_Audit.pdf", it also proves that the correlations between each variable are weak since the biggest correlation coefficients is 0.52. The four highest correlation coefficients are “Age” with “Marital” (0.52), “Income” with “Gender” (1: Female, 2: Male) (-0.42), “Income” with “Marital” (-0.27) and “Gender” with “Hours” (0.24). Since “Marital” is a categorical variable and the values are not meaningful with order, the correlation coefficients between “Marital” and other variables are misleading. In terms of the correlation between “Gender” and “Income”, it can be easily explained that man is likely to have higher income comparing to women. The positive correlation between “Gender” and “Hours” is understandable since male employee may slightly work longer than female employee.

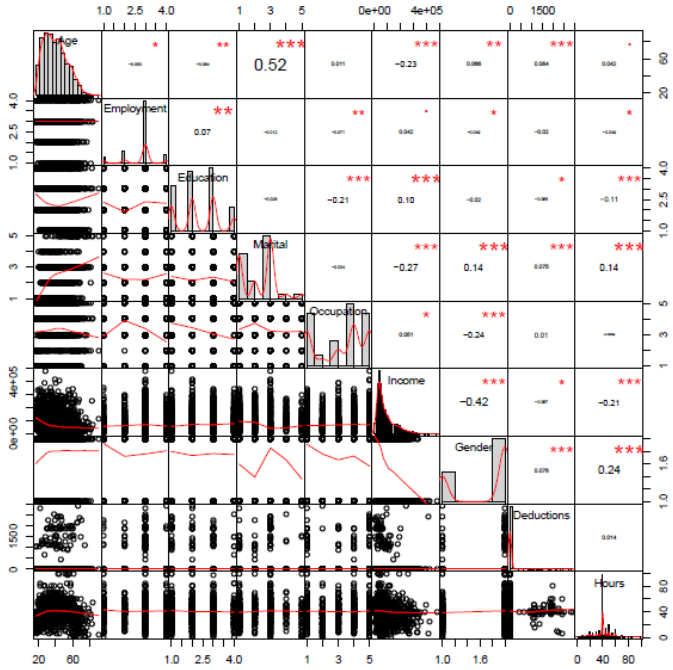


Figure 9 Scatterplot of each variable

3 Dimension Reduction

Preparation:

Following packages are installed:

1. “data.table”: To read the data file.
2. “factoextra”: To perform principal component analysis and data visualization.
3. “FactoMineR”: To perform data visualization of principal component analysis.
4. “dplyr”, “tidyverse”: To perform data manipulation.
5. “corrplot”, “ggplot2”, “PerformanceAnalytics”: To perform data visualization.
6. “lattice”, “caret”, “rpart”, “rpart.plot”: To perform decision tree analysis and its data visualization.

3.1 Factor Analysis

1. Since the data includes both continuous variables and categorical variables, Factor analysis of mixed data (FAMD) is used for analysis.
2. The FAMD analysis shows that PC1 can explain 11.9% of the total variance, PC2 can explain 9.8% of the total variance and so on. First 6 PCs can nearly explain 50% of the total variance in total. (FAMD_Summary.csv, Percentage of Explained Variance for FAMD.pdf).
3. Both contribution and coordinate tables show that "Income", "Marital", "Age" and "Gender" are highly correlated to PC1. "Occupation" and "Employment" are more related to PC2. "Occupation", "Employment" and "Education" are highly related to PC3, and so on. (FAMD_Contributions_of_variables.csv, Contribution to the first dimension.pdf, Contribution to the second dimension.pdf)
4. The quality of representation shows a slightly different result comparing to contribution table and coordinates table. It indicates that "Income" and "Gender" has a good representation of PC1. "Employment" and "Occupation" has a good representation of PC2. "Occupation" is also a good presentation of PC3. (FAMD_Quality_of_Representation_of_variables.csv)
5. From the "Factor Analysis Plot of Variables.pdf", it seems that "Deduction" and "Education" contribute least to first two principal components.
6. Due to that only around 20% variance explained by first two components, I feel factor analysis may not be a good option for this analysis regarding dimension reduction. So I decided to take another approach.

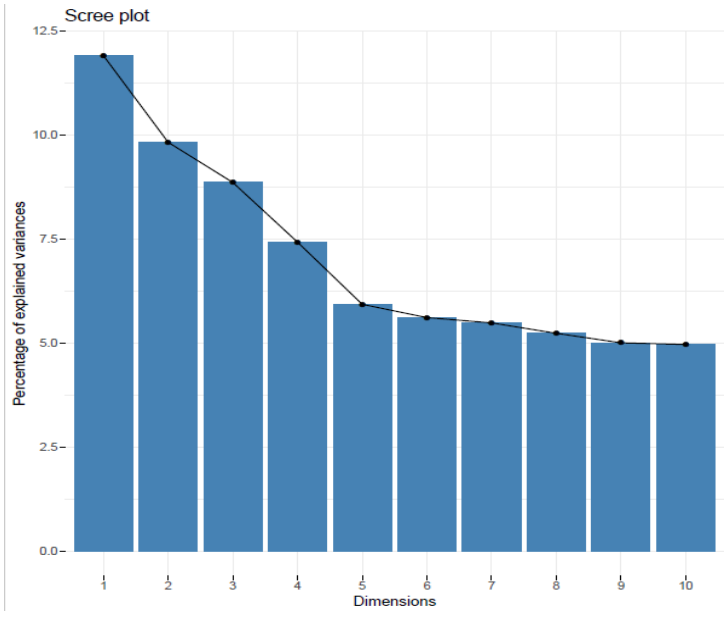


Figure 10 Scree plot of PCA analysis

3.2 Decision Tree

Part1: Use continuous target variable with Anova Regression Tree

1. Since the target continuous variable "Risk_Adjustment" suppose to have more information, first use it as a dependant variable and include all independent

variables in the model.

2. Unpruned regression tree (Unpruned Regression Tree for RISK_Adjustment.pdf) shows that “Marital” is the first node, “Education” and “Occupation” become a second node, and “Hours”, “Income”, “Employment” and “Age” become next nodes, and so on. There are two values in each small node: predicted value and percentage of observations in the node. Apparently, the tree has too many nodes and it needs to be pruned.
3. Complexity Parameter and cross validated error are mainly used to prune the tree. The best tree should have the value of cp the least, so that the cross-validated error rate (xerror) is minimum. Based on the table of “Unpruned Regression Tree CP value for RISK_Adjustment.csv”, the CP value of 0.01543213 is the best for this tree.
4. Based on the best CP, we prune the tree again and it comes with only “Marital”. (Pruned Regression Tree for RISK_Adjustment.pdf)

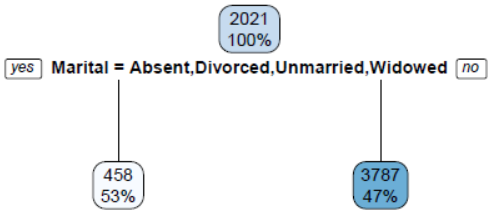


Figure 11 Regression Tree for RISK_Adjustment

Part2: Use continuous target variable with Conditional Inference Tree

1. The result of the regression tree is not satisfying, so the Conditional Inference Tree is used for another approach.
2. The conditional inference tree uses significance test methods to select and split recursively the most related predictor variables to the outcome. This can limit overfitting compared to the classical algorithm.
3. In the Conditional Inference Tree (Conditional inference tree for RISK_Adjustment.pdf), the p-value indicates the association between a given predictor variable and the outcome variable. The first decision node at the top shows that “Marital” is the variable that is most strongly associated with Risk_Adjustement with p value < 0.001, and thus is selected as the first node. Second decision nodes show that “Occupation - Professional” and “Employment - Private” are other variables that are associated with the target variable, with P-value of 0.005 and 0.015 respectively. The third node is the “Deduction” with P-value of 0.005.

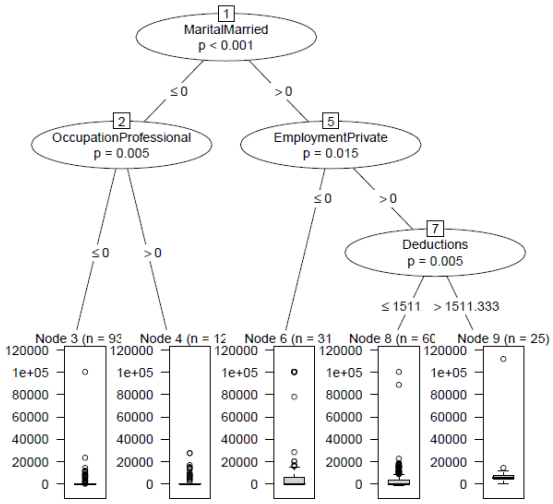


Figure 12 Conditional inference tree for RISK_Adjustment

Part3: Use categorical target variable with Classification Regression tree

1. The Anova regression tree does not give us a good result. Since we have a binary target variable, classification regression tree is another option.
2. The unpruned classification tree shows “Marital” is the first node, and then “Occupation” and “Deduction” become the second nods, “Education” becomes the third node and so on.
3. In each node, the top value is the predicted response, the value below is the predicted probability of 1, the last is the percentage of observations in the node.
4. Based on the table of “Unpruned Classification Tree CP value for TARGET_Adjusted.csv” with same rule, the CP value of 0.006479 is the best for this tree.
5. The pruned tree shows “Marital” is the first node, and then “Deduction” and “Occupation” become second nodes, and then the next node is “Education”, and so on. The variable “Gender” is dismissed from the tree.
6. The importance value of the variables shows “Marital” and “Income” has highest values. “Age”, “Occupation” and “Education” have values over 50. “Gender”, “Hours” and “Deduction” has value between 20 - 50. “Employment” has the lowest values less than 10.
7. The confusion matrix using pruned tree shows that the accuracy is 0.86 and the sensitivity is 0.94. The Specificity is a bit low, which is 0.62.

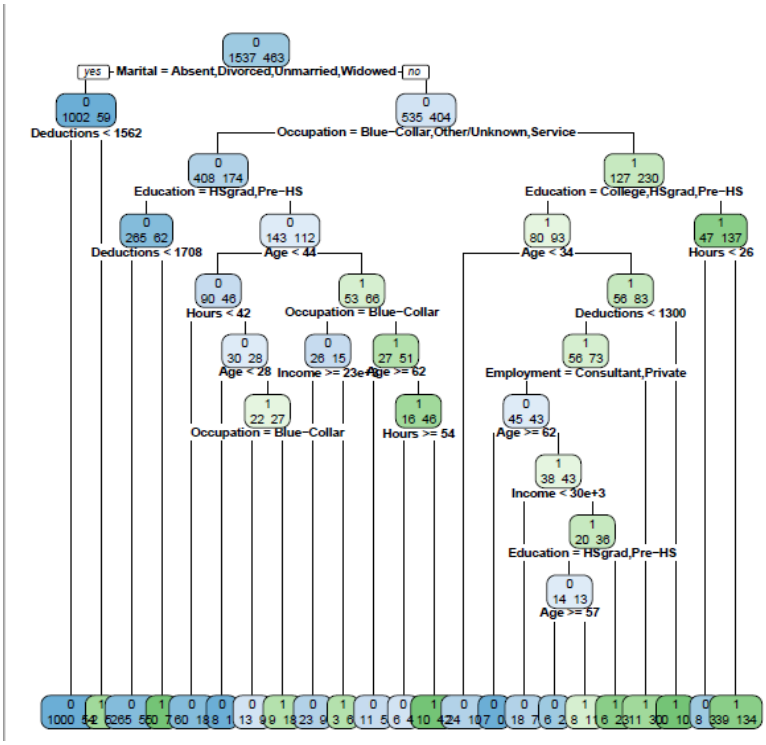


Figure 13 Regression tree for TARGET_Adjusted

Part4: Use categorical target variable with Conditional Inference Tree

1. The conditional inference tree based on binary variable TARGET_Adjusted shows that first node is "Marital", and the second nodes are two values of "Occupation" and the third nodes are "Deduction" and 1 "Occupation" value. (Conditional inference tree for TARGET_Adjusted.pdf)

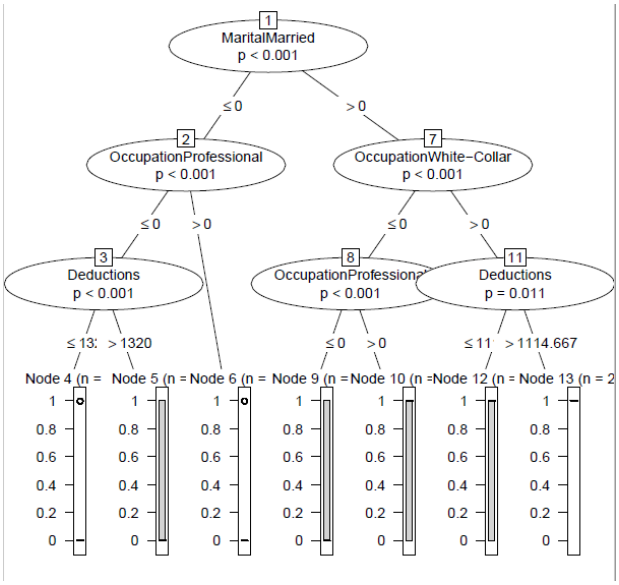


Figure 14 Conditional Inference tree for TARGET_Adjusted

Conclusion: Since the variables in "Audit" data does not have strong correlation with each other, which make factor analysis less attractive to dimension deduction. The regression decision tree based on continuous variable of "RISK_Adjustment" does not

work as well as classification decision tree based on binary variable "TARGET_Adjusted". Generally, "Marital", and "Education", "Occupation", "Deduction" shows their contribution and importance for predicting target variables. Considering "Age", "Gender", "Income" and "Hours" are important social-economic information, so I suggest to keep them in the analysis, "Employment" does not show strong evidence that they are valuable enough and since it has least importance value, it can be dismissed as an independent variable.

4 Data Reduction and Clustering

Preparation:

Based on the analysis on dimension reduction, I decided to exclude "Employment" since it has least influence based on value of importance with regards to classification tree.

Following packages are installed:

1. "tidyr", "dplyr", "descr" and "tables": To manipulate and summarize the data.
2. "NbClust", "cluster": To perform cluster analysis and evaluate.
3. "ggplot2" and "factoextra": To perform data Visualization of cluster analysis.
4. "reshape2" and "stringi": To perform Hopkins statistic and its Visualization.
5. "clValid","pvclust" and "kohonen": To compare and evaluate clustering algorithms.

Clustering Analysis:

1. "Updated Audit" data excludes variable of "Employment". Since all categorical variables have ordinal levels presented by character, so it is necessary to use numeric value to replace the characteristic levels in all categorical variables. After the transformation, the program also using "daisy" function to calculate Gower distance metric. In order, for a yet-to-be-chosen algorithm to group observations together, we first need to define some notion of (dis)similarity between observations. A popular choice for clustering is Euclidean distance. However, Euclidean distance is only valid for continuous variables, and thus is not applicable here. In order to yield sensible results for a clustering algorithm, we need to use a distance metric that can handle mixed data types. In our case, we will use Gower distance.
2. Different distance methods of Agglomerative Hierarchical Clustering (Res_Median.pdf, Res_Mcquitty.pdf, Res_Average.pdf, Res_Complete.pdf, Res_Single.pdf, Res_Wdd2.pdf, Res_Wdd.pdf) and Divisive Hierarchical Clustering (Divisive.pdf) are performed to generate dendrograms. Based on those plots, Hierarchical Clustering seems a good choice for this data. Also, from all above dendrograms, 4 - 10 clusters can all be a good number for further analysis.
3. In order to find optimal number of clusters, multiple methods including "Elbow method", "Silhouette method" and "Gap statistic" are performed. "Elbow method"

indicates 4 clusters is the optimal number and "Silhouette method" suggests 3 clusters is the optimal number. Whereas "Gap statistic" shows 10 clusters is the best option. (Elbow_Method.pdf, Silhouette_Method.pdf, Gap statistic.pdf).

4. In order to keep the simplicity, Hierarchical Clustering with "Ward.D2" method at 4 cluster is performed for clustering analysis. The Cluster Dendrogram (Final Hierarchical Clustering with 4 cluster.pdf) clearly shows the chosen 4 clusters with different colors. Indeed, from this dendrogram, we can tell that if the cut point moves a bit lower, we can have either 5, 7 or 10 clusters, which may increase the accuracy, but may bring challenges for explanation.

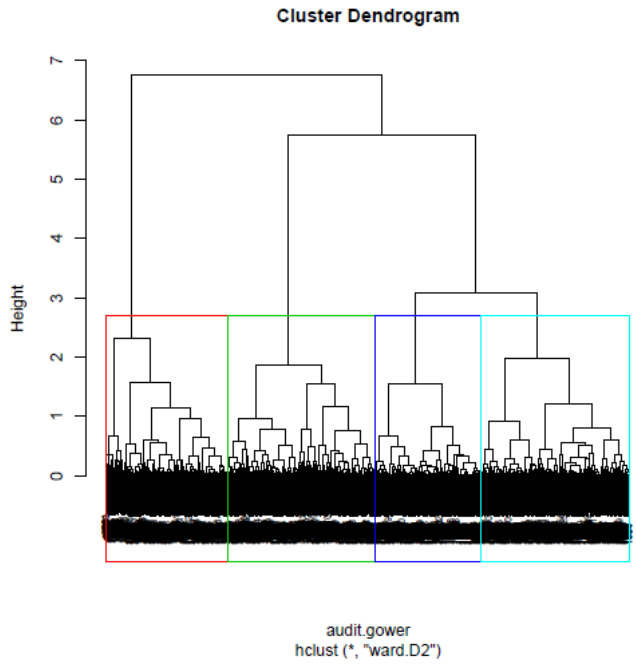


Figure 15 Hierarchical Clustering

5. The descriptive statistics for all continuous variables and cross table with row percentages of all categorical variables explain the discrepancies between clusters. There are 564 records in cluster 1, 554 records in cluster 2, 463 records in cluster 3 and 419 records in cluster 4.
   - Cluster 1: Individuals in cluster 1 are all females with average age of 37. They have highest average annual income of 123,045. They also have lowest weekly working hours of 35. They have lowest deduction claim of 23 dollars by average. Proportionally, individuals in this cluster have high proportion of working for private company (74%) or government (14%). With regards to occupation, they have higher proportion of professional (13%), white-collar (35%) and service (33%) comparing to cluster 2 and 3. Their proportion of high school (34%) is the second highest and college (34%) education is the highest comparing to other clusters. This cluster also has highest proportion of divorced (49%) and absent (27%) marital status.
   - Cluster 2: Individuals within this cluster are all male with average age of 38 and their annual income is 71,243. Their deduction claim is 32 dollars and they have average weekly working hours of 41. Regarding occupation, they have

highest proportion of working in service sector (59%). They also have the lowest proportion of self-employed/others for employment.

- Cluster 3: People in cluster 3 are most different group comparing to others. All individuals with monetary adjustment / productive audit are included in this cluster. People in cluster 3 have lowest annual income of 59,564 with oldest average age of 44. This cluster also has longest 45 average weekly working hours. They have highest claim deduction of 181 dollars and have adjustment of 8,730 dollars on average. This cluster also has highest education level since it has the highest proportion of Bachelor degree and above (47%) and lowest pre-high school proportion (2%). This is also the only cluster with both female (15%) and male (85%) population. Their professionals and White-collar occupation proportion are also the highest, 22% and 36% respectively. Most of them (87%) are married.
- Cluster 4: They are the youngest population with average age of 36 with average annual income of 78,598 with 40 average working hours weekly. Most of them are all males working in blue collar jobs (90%) with high proportion of high school diploma (45%) and pre-high school education (19%). They have highest proportion of consultant (10%) and self-employed/other(13%) comparing to people in other cluster.

6. The income and education information of individuals in cluster 3 contradicts to the common sense that high educational level should have higher income in general. This cluster has the highest educational level with lowest annual income. I calculate the average income for all original educational level and found that the average annual income with “Bachelor”, “Master” and “Doctorate” are 71,483, 60,318 and 57,357, which nearly the lowest among all educational levels. Whereas, highest income comes with “Associate”, “Yr11”, “Yr12”, “Yr7t8” and “Yr9”. That explains the “confusion” in cluster 3 between the education level and annual income. (Average income for different education levels.csv).

5 Supervised Learning

In this section, we create the models of different supervised learning algorithms to predict if one client is productive audit or not, then examine and compare their accuracy of the classification. The following supervised learning algorithms are Logistic Regression, K-Nearest Neighbor, Neural Network and Random Forest.

5.1 Logistic Regression

Preparation:

Before doing the logistics regression, we need to split our data into two sets: a training and a test set. We use the training set to build the model and test set to evaluate the model we built before. Setting the seed is paramount for reproducibility as createDataPartition() will shuffle the data randomly before splitting it, so that we can get same split in subsequent runs. As we set p=0.8, 80% of the original data is used as the training set, while the rest 20% is used as test set.

Following packages are installed:

1. “caret”: To perform createDaraPartition.
2. “stats”, “statsr”: To fit generalized linear models.

Using the logistic model:

- The R function glm() is used to compute logistic regression. We specify the option family=binomial to tell R that we want to fit logistic regression. A logistic regression using TARGET_Adjusted as the response variable, and all other 8 variables as predictors is fitted. Its parameter estimates are performed by summary() as below.

```
Call:
glm(formula = TARGET_Adjusted ~ ., family = binomial, data = train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.22308  -0.55635  -0.24473  -0.06099   2.84008

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              -4.96042    0.50787  -9.767  < 2e-16 ***
Age                       2.12592    0.49276   4.314 1.60e-05 ***
Income                    0.93863    0.70644   1.329 0.183954
Deductions                3.07772    0.61862   4.975 6.52e-07 ***
Hours                     2.90853    0.63230   4.600 4.23e-06 ***
EducationCollege         -0.75035    0.19193  -3.910 9.25e-05 ***
EducationHSgrad          -1.35001    0.20855  -6.473 9.59e-11 ***
EducationPre_HS          -2.59124    0.41867  -6.189 6.05e-10 ***
MaritalDivorced           0.06905    0.35259   0.196 0.844748
MaritalMarried            2.55288    0.25135  10.157  < 2e-16 ***
MaritalUnmarried          0.66055    0.55141   1.198 0.230943
MaritalWidowed           -0.37785    0.72665  -0.520 0.603076
OccupationOther_Unknown -0.09435    0.42383  -0.223 0.823828
OccupationProfessional    0.92609    0.26844   3.450 0.000561 ***
OccupationService         0.17745    0.21550   0.823 0.410256
OccupationWhite_Collar    1.12309    0.21829   5.145 2.67e-07 ***
GenderMale                0.39426    0.25540   1.544 0.122658
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1756.5  on 1599  degrees of freedom
Residual deviance: 1142.9  on 1583  degrees of freedom
AIC: 1176.9

Number of Fisher Scoring iterations: 6
```

Figure 16 Summary of the variables

- In the output above, age, deductions, hours, three terms for the education, one term for marital and one term for occupation are statistically significant. For the categorical variables, we have evidence that the coefficient of all other classes differs from the coefficient of reference group. The logistic regression coefficients give the change in the log odds of the outcome for a unit increase/decrease in the predictor variable.
    - The coefficient estimate of the variable Age is 2.1259, which is positive. This means that every unit increase in age is associated with increase in

the probability of having productive audit. The log odds of having productive audit increased by 2.1259.

  - The indicator variables for Education have a slightly different interpretation. EducationBachelor, MaritalAbsent, OccupationBlue_Collar and GenderFemale has been chosen to be the reference group by default. Each coefficient of categorical variables denotes the difference between the coefficient of reference group and the corresponding level. For example, the coefficient estimates of the variable EducationPre_HS is -2.59 which means having pre-high school degree versus bachelor degree, changes the log odds of having productive audit by -2.59.

- Logistics regression is modeling the probability that an individual has productive audit. In another word, a response closer to 1 indicates higher chance of having productive audit, while a response closer to 0 indicates a higher chance of having non-productive audit. Thus, a threshold of 0.5 is used to determine whether an individual is predicted to have productive audit or non-productive audit. A confusion matrix is presented to evaluate how well the model predicts TARGET_Adjusted. The prediction result has an accuracy of 83.75%, and a misclassification rate of 16.25%.

```
> print(tblog)

pred     Actual:0 Actual:1
  Pred:0      293       40
  Pred:1       25       42
```

Figure 17 Confusion matrix of logistic regression

$$\frac{293 + 42}{400} = 83.75\%$$
$$1 - 83.75\% = 16.25\%$$

5.2 K-Nearest Neighbor

Preparation:

We only can use numeric predictor variables because k-NN involves calculating distances between datapoints. The response variable can remain as a factor variable. We need to convert categorical variables to numeric variables. Use str() to check which variables are categorical variables. Call dummy.code() to convert Gender, Education, Marital and Occupation to numeric.
Split data into training and test sets by sample() and partition 80% of the data into training test and the rest into test set.

Following packages are installed:
1. “psych”: To do dummy code.
2. “caret”: To pick optimal number of neighbors(k).
3. “class”: To run k-NN classification.

4. “gmodels”: To generate a summary table of accuracy.

Using the k-NN classification:

- Do train() and the function picks the optimal number of neighbors(k). looking at the output and plot of k-NN model below, it chose k=9 at which its accuracy and kappa peaked.

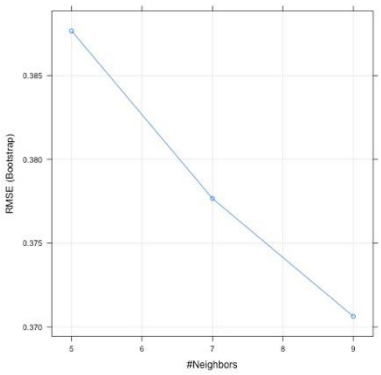
```
> ta_pred_caret
k-Nearest Neighbors

1600 samples
  19 predictor

Pre-processing: centered (18), scaled (18), ignore (1)
Resampling: Bootstrapped (25 reps)
Summary of sample sizes: 1600, 1600, 1600, 1600, 1600, 1600, ...
Resampling results across tuning parameters:

  k  RMSE       Rsquared   MAE
  5  0.3876608  0.2529803  0.2235224
  7  0.3776638  0.2674648  0.2268879
  9  0.3706312  0.2794426  0.2292235

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was k = 9.
```



- Run k-NN classification with k=9 by knn() and create a table examining model accuracy by CrossTable(). The prediction result has an accuracy of 82.25%, and a misclassification rate of 17.75%.

class_comparison$observed	class_comparison$predict: 0	1	Row Total
0	280	38	318
1	33	49	82
Column Total	313	87	400

$$\frac{280 + 49}{400} = 82.25\%$$

$$1 - 82.25\% = 17.75\%$$

5.3 Neural Networks

A neural Network is initially trained or fed large amounts of data, the output of each neuron is computed by some non-linear function of the sum of the inputs.

Preparation:

In neural networks, the response variable is usually the binary variable or has more classes. So in this dataset, we use variable “TARGRT_Adjusted” as response variable and other variables except variable “Employment” to do neural networks and factor them.

Following package is installed:

1. “neuralnet”: To train neural networks.

Neural Networks Analysis:

1. In this part, we mainly talk about the different number of neurons in one single hidden layer, we examine each model by their accuracy of classification. In general,

the number of neurons should between the number of input layer and output layer, and too many neurons will cause the problem of overfitting, so we try to fit model with neurons from 1 to 10.

2. Through the function "neuralnet", we can see with the number of nodes increasing, the error in each model is decreasing.
3. Then we use the training set, test set to fit in the model and predict. And the following graph shows the trend of percentage of classification correctly with neurons from 1 to 10. In the training set, the accuracy is increasing with the number of neurons adding, while the accuracy of test set is decreasing when the number of neurons increase.
4. So according to the result, we suggest use 4-neurons, and the accuracy of test set and training set are 89% and 90% respectively.

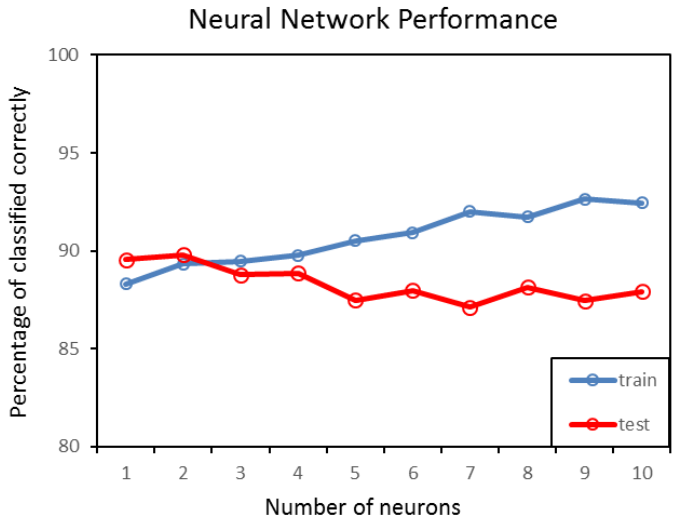


Figure 19 Accuracy performance of neural networks

5.4 Random Forest

In Section 3, we have talked about the classification tree. Random Forest is also a popular ensemble method which can be used to build predictive models for classification. It aims to reduce the correlation issue by choosing only a subsample of the feature space at each split.

Preparation:

Based on the previous analysis, in this part, variable "Employment" is excluded, and using binary variable "TARGRT_Adjusted" and other variables to do random forest.

Following packages are installed:

1. "randomForest": To classify with Random Forest.
2. "randomForestExplainer": To explain the distribution of variables.
3. "ROCR": To predict the variables.

Random Forest Analysis:

1. We use training data set to create the random forest model and get the result. The number of trees is 500, and the number of variables tried at each split is 2. The OOB estimate of error rate is 16.61%.
2. Feature importance is one of the key aspects of model. Understanding which variable is contributing the most is very important to interpret the results. The following variable importance plots show how important each 8 variable is when classifying the data and the detailed proportion. And we can see the first three features which contribute the most are Income, Marital and Age, with about 87.60, 79.20 and 76.92 respectively. Age, Education, Hours, Occupation and Deductions are between 20 to 60. And Gender is the least, with less than 10.

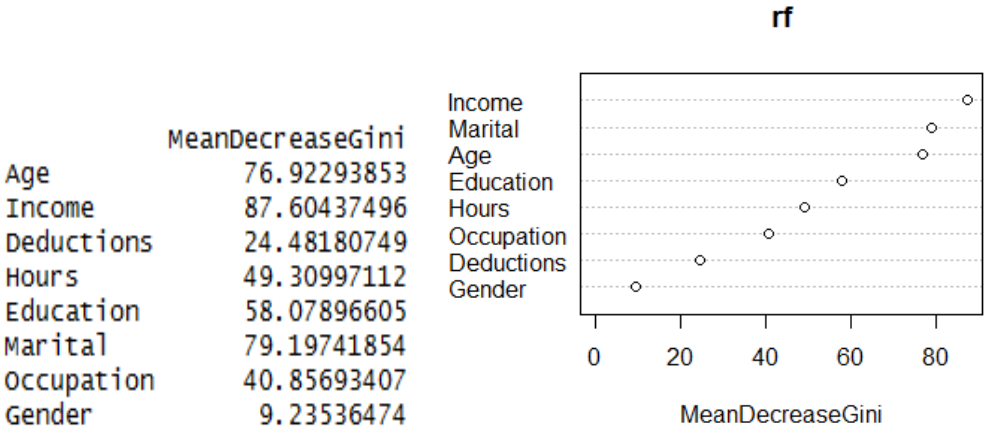


Figure 20 MeanDecreaseGini of random forest

3. To obtain the distribution of minimal depth, we use the function "randomForestExplainer". The following plot shows the distribution of minimal depth for the 8 variables according to mean minimal depth calculated using top trees.

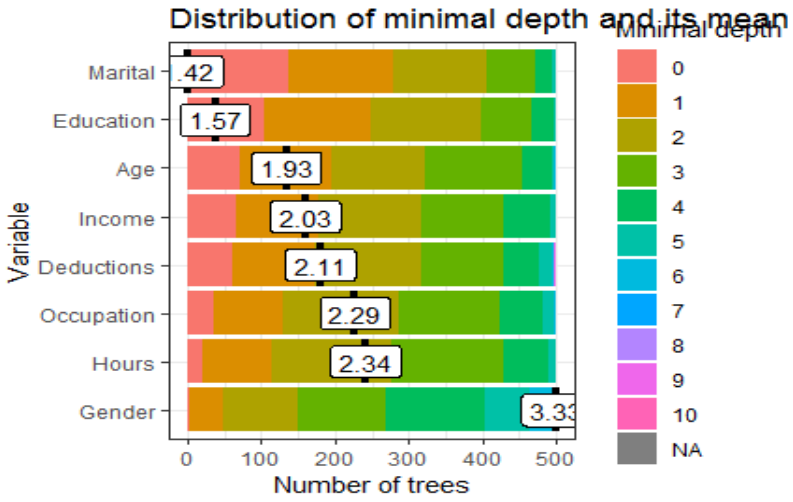


Figure 21 Distribution of minimal depth and mean

4. Then we use the forest model to predict the test data set. The confusion matrix shows that the accuracy is 0.85, the sensitivity is 0.94 and the specificity is 0.54, which is approximately equal to the result of decision tree.

6 Conclusion

This report we analyze the audit data by several steps which include preprocessing the raw data, visualization, dimension reduction, data reduction, unsupervised learning(clustering) and supervised learning (logistic regression, k-nearest neighbor, neural network).

The first part of visualization, we create plots to explore the relationship between "TARGET_Adjusted" (the response variable which we are interested in) and other explanatory variables roughly. Then from the scatter plot matrix, it seems that there's no strong correlation between variables since there's no obvious trend in the scatter plots.

For the dimension reduction, firstly we try factor analysis since the data includes both continuous variables and categorical variables. The factor analysis doesn't give us expected results, maybe because there's no strong correlation between variables. Then the regression decision tree also doesn't work well. Based on the previous conclusion of plots, having productive audit or not doesn't seem vary across employment, so employment can be dismissed.

In order to keep simplicity in clustering analysis, Hierarchical Clustering with "Ward.D2" method at 4 cluster is performed. There are 564 records in cluster 1, 554 records in cluster 2, 463 records in cluster 3 and 419 records in cluster 4.

We perform logistic regression, k-nearest neighbor, neural network and random forest under supervised learning. Accuracy of each model is calculated in order to evaluate the goodness. Although neural network has highest accuracy (about 90%), we still need to consider saving computation time and space when choosing the best model.

7 Responsibility in the team

1. Jiaming Han : 100631020
   Dimension Reduction, Data Reduction, Clustering, Data processing.
2. Yuhua Cong: 100980213
   Data visualization, Logistic Regression, K-Nearest Neighbor, Conclusion.
3. Jingyi Chen: 101108268
   Data visualization, Neural Networks, Random Forest, Conclusion

Appendix: Code for audit analysis

```
# Packages for data dimension reduction
install.packages("data.table")
install.packages("factoextra")
install.packages("FactoMineR")
install.packages("corrplot")
install.packages("ggplot2")
install.packages("plyr")
install.packages("dplyr")
install.packages("PerformanceAnalytics")

install.packages("tidyverse")
install.packages("ipred")
install.packages("caret")
install.packages("lattice")
install.packages("rpart")
install.packages("rpart.plot")



library("data.table")
library("factoextra")
library("FactoMineR")
library("corrplot")
library("ggplot2")
library(plyr)
library("dplyr")
library("PerformanceAnalytics")

library("tidyverse")
library("lattice")
library("ipred")
library("caret")
library("rpart")
library("rpart.plot")
library("RColorBrewer")

# Data cleaning
# Read Audit data in R
```

```
audit <- read.csv("~/Desktop/GRADUATE/5703/final project/Data/audit.csv",
stringsAsFactors=FALSE)
View(audit)
# Make ID as column ID
rownames(audit) <- audit$ID
audit$ID <- NULL
#Set 'None' for missing values.
audit[is.na(audit)]<-'None'

# Review variables and clean/re-organize data
summary(audit$Age)

table(audit$Employment)
# Combined unemployed, volunteer and missings with self-emplyed and others
(79+1+1+100)
audit$Employment <- gsub('^SelfEmp', 'SelfEmp_Other', audit$Employment)
audit$Employment <- gsub('^Unemployed', 'SelfEmp_Other', audit$Employment)
audit$Employment <- gsub('^Volunteer', 'SelfEmp_Other', audit$Employment)
audit$Employment <- gsub('^None', 'SelfEmp_Other', audit$Employment)
# Also combined all levels of goverment employees
audit$Employment <- gsub('^PSFederal', 'Government', audit$Employment)
audit$Employment <- gsub('^PSLocal', 'Government', audit$Employment)
audit$Employment <- gsub('^PSState', 'Government', audit$Employment)

table(audit$Education)
# Re-organize education
# Combined year12 and before to 'Pre-HS'
audit$Education <- gsub('^Yr12', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Yr11', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Yr10', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Yr9', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Yr7t8', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Yr5t6', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Yr1t4', 'Pre_HS', audit$Education)
audit$Education <- gsub('^Preschool', 'Pre_HS', audit$Education)
# Combined Associate, Vocational to 'College'
audit$Education <- gsub('^Associate', 'College', audit$Education)
audit$Education <- gsub('^Vocational', 'College', audit$Education)
audit$Education <- gsub('^Professional', 'College', audit$Education)
# Combined 'Doctorate', 'Master' to 'Bachelor'
```

```
audit$Education <- gsub('^Doctorate', 'Bachelor', audit$Education)
audit$Education <- gsub('^Master', 'Bachelor', audit$Education)

table(audit$Marital)
# Combined 'Married-spouse-absent' to 'Married'
audit$Marital <- gsub('^Married-spouse-absent', 'Married', audit$Marital)

table(audit$Occupation)
# regroup occupation to White-Collar, White-Collar,Service and Other/Unknown
audit$Occupation <- gsub('^Clerical', 'White_Collar', audit$Occupation)
audit$Occupation <- gsub('^Cleaner', 'Blue_Collar', audit$Occupation)
audit$Occupation <- gsub('^Executive', 'White_Collar', audit$Occupation)
audit$Occupation <- gsub('^Farming', 'Blue_Collar', audit$Occupation)
audit$Occupation <- gsub('^Machinist', 'Blue_Collar', audit$Occupation)
audit$Occupation <- gsub('^Protective', 'Service', audit$Occupation)
audit$Occupation <- gsub('^Sales', 'Service', audit$Occupation)
audit$Occupation <- gsub('^Support', 'Service', audit$Occupation)
audit$Occupation <- gsub('^Transport', 'Service', audit$Occupation)
audit$Occupation <- gsub('^Repair', 'Blue_Collar', audit$Occupation)
audit$Occupation <- gsub('^Military', 'Other_Unknown', audit$Occupation)
audit$Occupation <- gsub('^Home', 'Other_Unknown', audit$Occupation)
audit$Occupation <- gsub('^None', 'Other_Unknown', audit$Occupation)

summary(audit$Income)
table(audit$Gender)
summary(audit$Deductions)
summary(audit$Hours)
summary(audit$RISK_Adjustment)
table(audit$TARGET_Adjusted)


write.csv(audit,'audit_cleaned.csv')
#Also Use Ggobi generate the scatter plot for variables and PCs
# Scatter Matrix of "SactterPlot_Varibles_PC1_PC9" is generated by Ggobi and saved
# scatterplots
install.packages("corrgram")
library(corrgram)
panel.cor <- function(x, y, digits=2, prefix="", cex.cor) {
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(0, 1, 0, 1))
```

```
    r <- abs(cor(x, y))
    txt <- format(c(r, 0.123456789), digits=digits)[1]
    txt <- paste(prefix, txt, sep="")
    if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)*r
    text(0.5, 0.5, txt, cex = cex.cor)
}

panel.hist <- function(x, ...) {
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(usr[1:2], 0, 1.5) )
    h <- hist(x, plot = FALSE)
    breaks <- h$breaks; nB <- length(breaks)
    y <- h$counts; y <- y/max(y)
    rect(breaks[-nB], 0, breaks[-1], y, col="cyan", ...)
}
#scatter plots
jpeg('scatter plots.jpeg')
pairs(audit[,c("Age","Income","Deductions","Hours")],
          upper.panel = panel.cor, diag.panel = panel.hist, cex.labels = .93)
dev.off()
# correlogram and correlation coefficients
jpeg('correlogram and correlation coefficients.jpeg')
corrgram(audit[,c("Age","Income","Deductions","Hours")],
                lower.panel=panel.shade, upper.panel=panel.pie,
                diag.panel=panel.minmax, cex.labels = .95)
dev.off()
round(cor(audit[,c("Age","Income","Deductions","Hours")]), digits = 2)

####### AGE ######
# histogram of age by TARGET_adj group
jpeg('histogram of age by TARGET_adj group.jpeg')
ggplot(audit) + aes(x=as.numeric(Age), group=TARGET_Adjusted,
fill=factor(TARGET_Adjusted)) +
    geom_histogram(binwidth=1, color='black')
dev.off()
#majority of the observations have 0-target_adjusted (nonproductive audit)

####### Employment #######
table(audit$Employment)
#explore the relationship between Employment and TARGET_Adjusted
```

```
#barplot of Employment by TARGET_Adjusted group
# get the counts by employment and target_adj group
count <- table(audit[audit$Employment == 'Consultant',]$TARGET_Adjusted)["0"]
count <- c(count, table(audit[audit$Employment ==
'Consultant',]$TARGET_Adjusted)["1"])
count <- c(count, table(audit[audit$Employment ==
'Government',]$TARGET_Adjusted)["0"])
count <- c(count, table(audit[audit$Employment ==
'Government',]$TARGET_Adjusted)["1"])
count <- c(count, table(audit[audit$Employment == 'Private',]$TARGET_Adjusted)["0"])
count <- c(count, table(audit[audit$Employment == 'Private',]$TARGET_Adjusted)["1"])
count <- c(count, table(audit[audit$Employment ==
'SelfEmp_Other',]$TARGET_Adjusted)["0"])
count <- c(count, table(audit[audit$Employment ==
'SelfEmp_Other',]$TARGET_Adjusted)["1"])
count
count <- as.numeric(count)
# create a dataframe
audit$Employment<-as.factor(audit$Employment)
Employment <- rep(levels(audit$Employment), each = 2)
TARGET_Adjusted <- rep(c('0', '1'), 4)
df <- data.frame(Employment, TARGET_Adjusted, count)
df
# calculate the percentages
df <- ddply(df, .(Employment), transform, percent = count/sum(count) * 100)
# format the labels and calculate their positions
df <- ddply(df, .(Employment), transform, pos = (cumsum(count) - 0.5 * count))
df$label <- paste0(sprintf("%.0f", df$percent), "%")
# bar plot of counts by employment with in group proportions
jpeg('histogram of Employment by Target_Adjusted.jpeg')
ggplot(df, aes(x = Employment, y = count, fill =TARGET_Adjusted )) +
   geom_bar(stat = "identity") +
   geom_text(aes(y = pos, label = label), position = position_stack(vjust = 0.1), size =
2.5) +
   ggtitle('TARGET_Adjusted with Employment')
dev.off()
#there's no significant difference between employment, all around 20%


######### Education ############
```

```
# create a dataframe
audit$Education<-as.factor(audit$Education)
df1 <- data.frame(table(audit$TARGET_Adjusted, audit$Education))
names(df1) <- c('TARGET_Adjusted', 'Education', 'count')
df1
# calculate the percentages
df1 <- ddply(df1, .(Education), transform, percent = count/sum(count) * 100)
# format the labels and calculate their positions
df1 <- ddply(df1, .(Education), transform, pos = (cumsum(count) - 0.5 * count))
df1$label <- paste0(sprintf("%.0f", df1$percent), "%")
# bar plot of counts by education with in group proportions
jpeg('histogram of Education by TARGET_Adjusted.jpeg')
ggplot(df1, aes(x = Education, y = count, fill = TARGET_Adjusted)) +
   geom_bar(stat = "identity") +
   geom_text(aes(y = pos, label = label), position = position_stack(vjust = 0.1),size = 2.5)
+
   ggtitle('TARGET_Adjusted with Education')
dev.off()
#in group proportion of having productive audit increase as the level of education
increases.
#For those who don't have any forms of college education, less than 20% have a
productive audit.

######### marital ########
audit$Marital<-as.factor(audit$Marital)
table(audit$Marital)
df2 <- data.frame(table(audit$TARGET_Adjusted, audit$Marital))
names(df2) <- c('TARGET_Adjusted', 'Marital', 'count')
df2
# calculate the percentages
df2 <- ddply(df2, .(Marital), transform, percent = count/sum(count) * 100)
# format the labels and calculate their positions
df2 <- ddply(df2, .(Marital), transform, pos = (cumsum(count) - 0.5 * count))
df2$label <- paste0(sprintf("%.0f", df2$percent), "%")
# bar plot of counts by marital status with in group proportions
jpeg('histogram of Marital by TARGET_Adjusted.jpeg')
ggplot(df2, aes(x = Marital, y = count, fill = TARGET_Adjusted)) +
   geom_bar(stat = "identity") +
   geom_text(aes(y = pos, label = label), position = position_stack(vjust = 0.1),size = 2.5)
+
```

```
    ggtitle('TARGET_Adjusted with Marital Status')
dev.off()

######### occupation ########
audit$Occupation<-as.factor(audit$Occupation)
table(audit$Occupation)
df3 <- data.frame(table(audit$TARGET_Adjusted, audit$Occupation))
names(df3) <- c('TARGET_Adjusted', 'Occupation', 'count')
df3
# calculate the percentages
df3 <- ddply(df3, .(Occupation), transform, percent = count/sum(count) * 100)
# format the labels and calculate their positions
df3 <- ddply(df3, .(Occupation), transform, pos = (cumsum(count) - 0.5 * count))
df3$label <- paste0(sprintf("%.0f", df3$percent), "%")
# bar plot of counts by Occupation with in group proportions
jpeg('histogram of Occupation by TARGET_Adjusted.jpeg')
ggplot(df3, aes(x = Occupation, y = count, fill = TARGET_Adjusted)) +
    geom_bar(stat = "identity") +
    geom_text(aes(y = pos, label = label), position = position_stack(vjust = 0.1),size = 2.5)
+
    ggtitle('TARGET_Adjusted with Occupation')
dev.off()

####### GENDER ######
table(audit$Gender)
df4 <- data.frame(table(audit$TARGET_Adjusted, audit$Gender))
names(df4) <- c('TARGET_Adjusted', 'Gender', 'count')
df4
# calculate the percentages
df4 <- ddply(df4, .(Gender), transform, percent = count/sum(count) * 100)
# format the labels and calculate their positions
df4 <- ddply(df4, .(Gender), transform, pos = (cumsum(count) - 0.5 * count))
df4$label <- paste0(sprintf("%.0f", df4$percent), "%")
# bar plot of counts by Gender with in group proportions
jpeg('histogram of Gender by TARGET_Adjusted.jpeg')
ggplot(df4, aes(x = Gender, y = count, fill = TARGET_Adjusted)) +
    geom_bar(stat = "identity") +
    geom_text(aes(y = pos, label = label), position = position_stack(vjust = 0.1),size = 2.5)
+
    ggtitle('TARGET_Adjusted with Gender')
```

```
dev.off()

#########################################################
# Target variables should not be included for dimension reduction and data reduction
targetvars<- c("RISK_Adjustment", "TARGET_Adjusted")
audit.pca <- audit[, ! names(audit) %in% targetvars, drop = F]
# Transfer categorical variables to numeric variables for coorelation matrix
audit.pca.num <-data.matrix(data.frame(unclass(audit.pca)))

# Generate plot of "ScatterPlot_Matrix_Correlation_Audit.pdf"
pdf("ScatterPlot_Matrix_Correlation_Audit.pdf")
chart.Correlation(audit.pca.num, histogram=TRUE, pch=19)
dev.off()

#Perfrom    Factor analysis of mixed data (FAMD)
res.famd <- FAMD (audit.pca, ncp = 10, graph = FALSE)
eig.val <- get_eigenvalue(res.famd)
head(eig.val)
out.famd.summary <- capture.output(eig.val)
write.csv (eig.val,file = "FAMD_Summary.csv" )

pdf("Percentage of explained Variance for FAMD.pdf")
fviz_screeplot(res.famd)
dev.off()
var <- get_famd_var(res.famd)
var
# Coordinates of variables
head(var$coord,10)
write.csv (head(var$coord,10), file = "FAMD_Coordinates_of_variables.csv" )

# Cos2: quality of representation on the factore map
head(var$cos2,10)
write.csv (head(var$cos2,10),file =
"FAMD_Quality_of_Representation_of_variables.csv" )

# Contributions to the dimensions
head(var$contrib,10)
write.csv (head(var$contrib,10),file = "FAMD_Contributions_of_variables.csv" )

# Factor Analysis Plot of Variables
pdf("Factor Analysis Plot of Variables.pdf")
```

```
fviz_famd_var(res.famd, repel = TRUE)
dev.off()

# Contribution to the first dimension
pdf("Contribution to the first dimension.pdf")
fviz_contrib(res.famd, "var", axes = 1)
dev.off()

# Contribution to the second dimension
pdf("Contribution to the second dimension.pdf")
fviz_contrib(res.famd, "var", axes = 2)
dev.off()

# Use Decision Tree to perfrom dimension reduction - CART Model
excldvars<- c("TARGET_Adjusted")
audit.dt.target <- audit[, ! names(audit) %in% excldvars, drop = F]


# Fit regression tree on "RISK_Adjustment"
anova.model <- rpart(RISK_Adjustment ~
Age+Employment+Education+Marital+Occupation+Income+Gender+Deductions+Hours
  ,
                                  data=audit.dt.target, control = rpart.control(cp = 0.0001),
method="anova")

# Prediction error rate in training data = Root node error * rel error * 100%
# Prediction error rate in cross-validation = Root node error * xerror * 100%
# Hence we want the cp value (with a simpler tree) that minimizes the xerror.

printcp(anova.model)
out.anovamodel.cp <- capture.output(printcp(anova.model))
write.csv (out.anovamodel.cp,file = "Unpruned Regression Tree CP value for
RISK_Adjustment.csv" )

pdf("Unpruned Regression Tree for RISK_Adjustment.pdf")
rpart.plot(anova.model)
dev.off()

bestcp.anova <- anova.model$cptable[which.min(anova.model$cptable[,"xerror"]),"CP"]
(bestcp.anova)
pruned.anova.model <- prune(anova.model, cp = bestcp.anova)
```

```
pdf("Pruned Regression Tree for RISK_Adjustment.pdf")
rpart.plot(pruned.anova.model)
dev.off()

# Conditional inference tree on RISK_Adjustment
install.packages("party")
library(party)
set.seed(123)
ctreemodel <- train(
   RISK_Adjustment ~
Age+Employment+Education+Marital+Occupation+Income+Gender+Deductions+Hours
,
   data=audit.dt.target,
   method = "ctree2",
   trControl = trainControl("cv", number = 10),
   tuneGrid = expand.grid(maxdepth = 3, mincriterion = 0.95 )
)

pdf("Conditional inference tree for RISK_Adjustment.pdf")
plot(ctreemodel$finalModel)
dev.off()

#----Clssification regression tree on "TARGET_Adjusted"

excldvars<- c("RISK_Adjustment")
audit.dt.risk <- audit[, ! names(audit) %in% excldvars, drop = F]

# Fit Classification trees on "TARGET_Adjusted"
class.model <- rpart(TARGET_Adjusted ~
Age+Employment+Education+Marital+Occupation+Income+Gender+Deductions+Hours
  ,
                                  data=audit.dt.risk, control = rpart.control(cp = 0.0001),
method="class")

# Prediction error rate in training data = Root node error * rel error * 100%
# Prediction error rate in cross-validation = Root node error * xerror * 100%
# Hence we want the cp value (with a simpler tree) that minimizes the xerror.

printcp(class.model)
out.classmodel.cp <- capture.output(printcp(class.model))
```

```
write.csv (out.classmodel.cp,file = "Unpruned Classification Tree CP value for
TARGET_Adjusted.csv" )

pdf("Unpruned Classification Tree for TARGET_Adjusted.pdf")
rpart.plot(class.model)
dev.off()

bestcp.classificaion <-
class.model$cptable[which.min(class.model$cptable[,"xerror"]),"CP"]
(bestcp.classificaion)
pruned.class.model <- prune(class.model, cp = bestcp.classificaion)

pdf("Pruned Classification Tree for TARGET_Adjusted.pdf")
rpart.plot(pruned.class.model,faclen = 0, cex = 0.7, extra = 1)
dev.off()

pruned.class.model$variable.importance
pruned.class.model.variable.importance <-
capture.output(print(pruned.class.model$variable.importance))
write.table(pruned.class.model.variable.importance, file =    "Variable Importance Value
of Classification Tree.txt", sep = "\t",
                        row.names = TRUE, col.names = NA)

# confusion matrix (training data)
xtab <- table( predict(pruned.class.model,type="class"),
audit.dt.risk$TARGET_Adjusted )
confusionMatrix(xtab)
cMatrix <- capture.output(confusionMatrix(xtab))
write.table(capture.output(cMatrix) , file = "Confusion Matrix of Classification Tree.txt" ,
sep = "\t")

print(xtab)

rownames(xtab) <- paste("Pred", rownames(xtab), sep = ":")
colnames(xtab) <- paste("Actual", colnames(xtab), sep = ":")
print(xtab)

# Conditional inference tree on TARGET_Adjusted
library(party)
set.seed(123)
ctreemodel2 <- train(
```

```
    TARGET_Adjusted ~
Age+Employment+Education+Marital+Occupation+Income+Gender+Deductions+Hours
,
    data=audit.dt.risk,
    method = "ctree2",
    trControl = trainControl("cv", number = 10),
    tuneGrid = expand.grid(maxdepth = 3, mincriterion = 0.95 )
)

pdf("Conditional inference tree for TARGET_Adjusted.pdf")
plot(ctreemodel2$finalModel)
dev.off()

# Packages for data data reduction - clustering
install.packages("dplyr")
install.packages("plyr")
install.packages("ggplot2")
install.packages("tidyr")
install.packages("factoextra")
install.packages("NbClust")
install.packages("cluster")
install.packages("magrittr")
install.packages("reshape2")
install.packages("stringi")
install.packages("clValid")
install.packages("kohonen")
install.packages("tables")
install.packages("pvclust")
install.packages("descr")
install.packages("stats")
install.packages("statsr")

library("dplyr")
library("plyr")
library("ggplot2")
library("factoextra")
library("tidyr")
library("NbClust")
library("cluster")
library("magrittr")
```

```
library("reshape2")
library("stringi")
library("clValid")
library("kohonen")
library("tables")
library("pvclust")
library("descr")
library("stats")
library("statsr")
#Based on decision tree analysis, the clsutering analysis exludes
#    employment
cols.dont.want <- c("Employment")
audit.cluster <- audit[, ! names(audit) %in% cols.dont.want, drop = F]
# Transfer character in categorical variables to numeric values
audit.cluster.num <-data.matrix(data.frame(unclass(audit.cluster)))
# Excude NA in the data. Indeed, there is no NAs in data file
audit.cluster.num <- audit.cluster.num %>%
   na.omit()

#Data Reduction - Clustering
#Since there are categorical varibles in the data,
#Returns the distance matrix with Gower's distance:
audit.gower <- daisy(audit.cluster.num,metric="gower" )

res <- get_clust_tendency(as.matrix(audit.gower), 40, graph = TRUE)
# Hopskin statistic
res$hopkins_stat
write.table(res$hopkins_stat, file = "Hopskin statistic for Audit file.txt" , sep = "\t")

# Since the Hopskin statistic is only 0.0514603, which means it is not a
# uniform data.

# Test Agglomerative Hierarchical Clustering
# with difference methods (dendrograms)
hc.w <- hclust(audit.gower, method = "ward.D")
hc.w2 <- hclust(audit.gower, method = "ward.D2")
hc.s <- hclust(audit.gower, method = "single")
hc.c <- hclust(audit.gower, method = "complete")
hc.a <- hclust(audit.gower, method = "average")
hc.mc <- hclust(audit.gower, method = "mcquitty")
```

```
hc.me <- hclust(audit.gower, method = "median")

pdf("Res_Wdd.pdf")
fviz_dend(hc.w, cex = 0.5)
dev.off()
pdf("Res_Wdd2.pdf")
fviz_dend(hc.w2, cex = 0.5)
dev.off()
pdf("Res_Single.pdf")
fviz_dend(hc.s, cex = 0.5)
dev.off()
pdf("Res_Complete.pdf")
fviz_dend(hc.c, cex = 0.5)
dev.off()
pdf("Res_Average.pdf")
fviz_dend(hc.a, cex = 0.5)
dev.off()
pdf("Res_Mcquitty.pdf")
fviz_dend(hc.mc, cex = 0.5)
dev.off()
pdf("Res_Median.pdf")
fviz_dend(hc.me, cex = 0.5)
dev.off()

# Divisive Hierarchical Clustering
# compute divisive hierarchical clustering
divisive.clust <- diana(as.matrix(audit.gower),
                                    diss = TRUE, keep.diss = TRUE)
pdf("Divisive.pdf")
plot(divisive.clust, main = "Divisive")
dev.off()

# Find optimal number of clusters
# Elbow method
pdf("Elbow_Method.pdf")
fviz_nbclust(as.matrix(audit.gower), FUN = hcut, method = "wss") +
    geom_vline(xintercept = 4, linetype = 2)+
    labs(subtitle = "Elbow method")
dev.off()
```

```
# Silhouette method
pdf("Silhouette_Method.pdf")
fviz_nbclust(as.matrix(audit.gower), FUN = hcut, method = "silhouette")+
   labs(subtitle = "Silhouette method")
dev.off()

# Gap statistic
pdf("Gap statistic.pdf")
set.seed(123)
fviz_nbclust(as.matrix(audit.gower), FUN = hcut, nstart = 25,    method = "gap_stat",
nboot = 50)+
   labs(subtitle = "Gap statistic method")
dev.off()

# Based on a comprehensive evaluation, Hierarchical Clustering
# with Ward's method for 4 clusters will be applied
audit.final <- hclust(audit.gower, method = "ward.D2" )

pdf("Final Hierarchical Clustering with 4 cluster.pdf")
plot(audit.final, cex = 0.6)
rect.hclust(audit.final, k = 4, border = 2:5)
dev.off()

# Cut tree into 4 groups
Cluster <- cutree(audit.final, k = 4)

# Number of members in each cluster
table(Cluster)
## Cluster
##      1      2      3      4
## 564 568 463 405

# add cluster in the data
audit.original.cluster <- cbind(audit,Cluster)

#Summary of each cluster

# Select continious variables
Continious.v <-c("Age", "Income","Deductions","Hours","RISK_Adjustment","Cluster")
Categorical.v <-c("Employment","Education","Occupation",
"Marital","Gender","TARGET_Adjusted","Cluster"     )
```

```
Audit.cluster.Cont <- audit.original.cluster[, names(audit.original.cluster ) %in%
Continious.v, drop = F]
Audit.cluster.Cate <- audit.original.cluster[, names(audit.original.cluster ) %in%
Categorical.v, drop = F]


mean.audit.cluster <- Audit.cluster.Cont    %>%
   group_by(Cluster)    %>%
   summarise_all(funs(mean))

write.csv(mean.audit.cluster, file = "Average values of continious variable in each
cluster.csv")

source("http://pcwww.liv.ac.uk/~william/R/crosstab.r")

Table1 <- crosstab(Audit.cluster.Cate, row.vars = "Cluster", col.vars = "Education", type
= c("f", "r"), style = "wide",
                              addmargins = FALSE)
T1.summary <- capture.output(Table1)
cat("Cluster vs Education", T1.summary , file="Cluster vs Education.txt",    sep = "\t" , fill
= TRUE, append=TRUE)

Table2 <- crosstab(audit.original.cluster, row.vars = "Cluster", col.vars = "Marital", type
= c("f", "r"), style = "wide",
                              addmargins = FALSE)
T2.summary <- capture.output(Table2)
cat("Cluster vs Marital", T2.summary , file="Cluster vs Marital.txt",    sep = "\t" , fill =
TRUE, append=TRUE)

Table3 <- crosstab(audit.original.cluster, row.vars = "Cluster", col.vars = "Gender", type
= c("f", "r"), style = "wide",
                              addmargins = FALSE)
T3.summary <- capture.output(Table3)
cat("Cluster vs Gender", T3.summary , file="Cluster vs Gender.txt",    sep = "\t" , fill =
TRUE, append=TRUE)

Table4 <- crosstab(audit.original.cluster, row.vars = "Cluster", col.vars = "Employment",
type = c("f", "r"), style = "wide",
                              addmargins = FALSE)
T4.summary <- capture.output(Table4)
```

```
cat("Cluster vs Employment", T4.summary , file="Cluster vs Employment.txt",    sep =
"\t" , fill = TRUE, append=TRUE)

Table5 <- crosstab(audit.original.cluster, row.vars = "Cluster", col.vars = "Occupation",
type = c("f", "r"), style = "wide",
                                addmargins = FALSE)
T5.summary <- capture.output(Table5)
cat("Cluster vs Occupation", T5.summary , file="Cluster vs Occupation.txt",    sep = "\t" ,
fill = TRUE, append=TRUE)

Table6 <- crosstab(audit.original.cluster, row.vars = "Cluster", col.vars =
"TARGET_Adjusted", type = c("f", "r"), style = "wide",
                                addmargins = FALSE)
T6.summary <- capture.output(Table6)
cat("Cluster vs TARGET_Adjusted", T6.summary , file="Cluster vs
TARGET_Adjusted.txt",    sep = "\t" , fill = TRUE, append=TRUE)

# Investigation between education and income
audit.raw <- audit <- read.csv("~/Desktop/GRADUATE/5703/final project/Data/audit.csv",
stringsAsFactors=FALSE)

# Make ID as column ID
rownames(audit.raw) <- audit.raw$ID
audit.raw$ID <- NULL
table(audit.raw$Education, exclude = NULL)

selc.var <-c( "Income" , "Education")
audit.raw.edu.income <- audit.raw[, names(audit.raw) %in% selc.var, drop = F]

mean.audit.edu.income <- audit.raw.edu.income    %>%
   group_by(Education)    %>%
   summarise_all(funs(mean))
write.csv(mean.audit.edu.income, file = "Average income for different education
levels.csv")


#########################################################################
#delete employment

#scale
max <- apply(audit[,c("Age","Income","Deductions","Hours")],2,max)
```

```
min <- apply(audit[,c("Age","Income","Deductions","Hours")],2,min)
max
min
audit_sc <-
as.data.frame(scale(audit[,c("Age","Income","Deductions","Hours")],center=min,
scale=max-min))
summary(audit_sc)
setDT(audit_sc, keep.rownames = TRUE)[]
audit_c<-audit[,c(3:5,7,11)]
setDT(audit_c, keep.rownames = TRUE)[]
audit_sca<-merge(audit_sc,audit_c,by="rn")
#delete rn
audit_sca$rn<-NULL
library(dplyr)
audit_sca=audit_sca %>% mutate_if(is.character,as.factor)
install.packages("caret")
library(caret) #to use createdatapartition
#split data into training set and test set by 4:1
set.seed(123)
splitindex<-createDataPartition(audit_sca$TARGET_Adjusted,p=0.8,list=FALSE,times=
1)
train<-audit_sca[splitindex,]
test<-audit_sca[-splitindex,]
table(train$TARGET_Adjusted)
table(test$TARGET_Adjusted)
####### model fitting-logistic regression ######


#logistic
logistic<-glm(TARGET_Adjusted~.,data=train,family=binomial)
summary(logistic)
#confusion matrix
prob<-predict(logistic,test,type='response')
pred<-rep('0',length(prob))
pred[prob>=0.5]<-'1'
tblog<-table(pred,test$TARGET_Adjusted)
tblog
rownames(tblog) <- paste("Pred", rownames(tblog), sep = ":")
colnames(tblog) <- paste("Actual", colnames(tblog), sep = ":")
print(tblog)
```

```
#prediction result has an accuracy of 83.75%, and a misclassification rate of 16.25%
######## knn #######
install.packages("psych")
library(psych)
install.packages("class")
library(class)
#make a copy
audit_knn<-audit_sca
# put outcome in its own object
ta_outcome<-audit_knn %>% select(TARGET_Adjusted)
#remove it from audit_knn
audit_knn<-audit_knn %>% select(-TARGET_Adjusted)
#determine which variables are categorical
str(audit_knn)
audit_knn$Gender<-dummy.code(audit_knn$Gender)
Education<-as.data.frame(dummy.code(audit_knn$Education))
Marital<-as.data.frame(dummy.code(audit_knn$Marital))
Occupation<-as.data.frame(dummy.code(audit_knn$Occupation))
#Combine new dummy variables with original data set.
audit_knn<-cbind(audit_knn,Education,Marital,Occupation)
audit_knn <- audit_knn %>% select(-one_of(c("Education", "Marital", "Occupation")))
head(audit_knn)

set.seed(123)
smp_size<-floor(0.8*nrow(audit_knn))
splitindex1<-sample(seq_len(nrow(audit_knn)),size=smp_size)

trainknn<-audit_knn[splitindex1,]
testknn<-audit_knn[-splitindex1,]

TARGET_Adjusted_train<-ta_outcome[splitindex1,]
TARGET_Adjusted_test<-ta_outcome[-splitindex1,]
ta_pred_caret<-train(trainknn,TARGET_Adjusted_train,method="knn",preProcess =
c("center","scale"))
sink("output of knn model.csv")
ta_pred_caret
sink()
jpeg('knn plot.jpeg')
plot(ta_pred_caret)
```

```
dev.off()
#9
ta_pred_knn<-knn(train=trainknn,test=testknn,cl = TARGET_Adjusted_train, k=9)

TARGET_Adjusted_test<-data.frame(TARGET_Adjusted_test)
class_comparison<-data.frame(ta_pred_knn,TARGET_Adjusted_test)
names(class_comparison)<-c("predict","observed")
head(class_comparison)
install.packages("gmodels")
library(gmodels)
sink("confusion matrix of knn-9.csv")
CrossTable(x = class_comparison$observed, y = class_comparison$predict,
                  prop.chisq=FALSE, prop.c = FALSE, prop.r = FALSE, prop.t = FALSE)
sink()



############### random forest #########
install.packages("randomForest")
install.packages("randomForestExplainer")
library(randomForest)
library(randomForestExplainer)
# Random forest
set.seed(123)
rf <- randomForest(TARGET_Adjusted~.,data=train)
rf
prerf <- predict(rf,test,type='response')
pred.rf<-rep('0',length(prerf))
pred.rf[prerf>=0.5]<-'1'
# confusion matrix
tbrf <- table(pred.rf, test$TARGET_Adjusted)
tbrf
rownames(tbrf) <- paste("Pred", rownames(tbrf), sep = ":")
colnames(tbrf) <- paste("Actual", colnames(tbrf), sep = ":")
print(tbrf)
importance(rf)
varImpPlot(rf)
#prediction result has an accuracy of 85%, and a misclassification rate of 15%
# Distribution of minimal depth
min_depth_frame <- min_depth_distribution(rf)
```

```
plot_min_depth_distribution(min_depth_frame)

# Neural Network
install.packages("neuralnet")
library(neuralnet)
m_train <- model.matrix(~., data=train)
m_test <- model.matrix(~.,data=test)
n <- colnames(m_train)
f <- as.formula(paste("TARGET_Adjusted ~", paste(n[!n %in%
c("TARGET_Adjusted","(Intercept)")],
                                                 collapse = "+")))

# Compare the results of different hidden layers(from 1 to 10) using "nn$result.matrix" to
choose the best one
set.seed(123)
nn1 <-
neuralnet(f,data=m_train,hidden=1,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 569.38
nn1$result.matrix[1,]
set.seed(123)
nn2 <-
neuralnet(f,data=m_train,hidden=2,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 525.39
nn2$result.matrix[1,]
set.seed(123)
nn3 <-
neuralnet(f,data=m_train,hidden=3,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 531.39
nn3$result.matrix[1,]
set.seed(123)
nn4 <-
neuralnet(f,data=m_train,hidden=4,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 495.73
nn4$result.matrix[1,]
set.seed(123)
nn5 <-
neuralnet(f,data=m_train,hidden=5,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 461.53
nn5$result.matrix[1,]
set.seed(123)
```

```
nn6 <-
neuralnet(f,data=m_train,hidden=6,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 434.71
nn6$result.matrix[1,]
set.seed(123)
nn7 <-
neuralnet(f,data=m_train,hidden=7,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 398.31
nn7$result.matrix[1,]
set.seed(123)
nn8 <-
neuralnet(f,data=m_train,hidden=8,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 374.33
nn8$result.matrix[1,]
set.seed(123)
nn9 <-
neuralnet(f,data=m_train,hidden=9,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 378.91
nn9$result.matrix[1,]
set.seed(123)
nn10 <-
neuralnet(f,data=m_train,hidden=10,err.fct="ce",linear.output=FALSE,stepmax=1e7) #
error 343.61
nn10$result.matrix[1,]

# The one with the smallest error is 10 layers
plot(nn10)

# Accuracy on train set
output1 <- compute(nn10,m_train[,2:17])
p1 <- output1$net.result
head(p1)

pred1 <- ifelse(p1>0.5,1,0)
tab1 <- table(pred1,m_train[,18])
tab1
# Calculate the misclassification
MSE.nn1 <- sum((m_train[,18]-p1)^2)/nrow(m_train)
MSE.nn1
```

```
# Accuracy on test set
output2 <- compute(nn10,m_test[,2:17])
p2 <- output2$net.result
head(p2)
pred2 <- ifelse(p2>0.5,1,0)
tab2 <- table(pred2,m_test[,18])
tab2
MSE.nn2 <- sum((m_test[,18]-p2)^2)/nrow(m_test)
MSE.nn2

# According to result of the misclassification, choose 10 layers
```