Assignment 2 Report

JUNE 2

PRACTICAL DATA SCIENCE

Authored by: Sridevi Pamarthi (s3778317), Amrutha Sreevalsan (s3765847)

Contents

Abstract

Credit risk refers to the risk of default on a debt that may arise from a borrower failing to make required payments. One of the important steps in effective credit risk management for any financial institution is to gain a complete understanding of risk at the customer level. So that the financial institutions define limits for customers to avoid any undesirable consequences.

The goal of this research is to identify the risk at the customer level by analyzing the payments' trend, which helps to predict whether the customer will likely get defaulted within next six months period.

This report uses the credit card payment dataset produced by a financial institution in Taiwan and illustrates the analysis that is undertaken using the "classification technique" to achieve the goal of this research.

Moreover, the report classifies the customers into Credible and Non-Credible for ease of understanding and includes the assessment of various hypothesis made by us together with related inferences. The classification models Decision-Tree and K-Nearest Neighbor (KNN) are used to obtain the results. Overall, the results indicate that majority of the customers are Credible and will not likely get defaulted.

1. Introduction

The report summarizes the data retrieving, preparation, exploration and modelling tasks which are carried out on the default credit card payments' dataset taken from the UCI repository.

With the knowledge learnt from the previous classes and using the below provided references, the appropriate steps needed for the above-mentioned tasks have developed and implemented in the IPython Jupyter Notebook.

Below is the summary of the dataset:

- ✓ It consists of 30000 rows x 25 columns
- ✓ It includes both Categorical and Numerical variables
- ✓ It includes customers' payment information for six months
- ✓ The time limit for the customer(s) to get defaulted due to non-payment is over 6 months
- ✓ similar customers with repayment status as 0 (Timely pay) are taken for the analysis to reduce bias.

Following are 25 variables/columns available in the dataset:

Variable Name	Variable Description
ID	ID of each client
LIMIT_BAL	Amount of given credit in NT dollars (includes individual and family/supplementary credit

SEX	Gender (1=male, 2=female)
EDUCATION	(1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)
MARRIAGE	Marital status (1=married, 2=single, 3=others)
AGE	Age in years
PAY_0	Repayment status in September, 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, ... 8=payment delay for eight months, 9=payment delay for nine months and above)
PAY_2	Repayment status in August, 2005 (scale same as above)
PAY_3	Repayment status in July, 2005 (scale same as above)
PAY_4	Repayment status in June, 2005 (scale same as above)
PAY_5	Repayment status in May, 2005 (scale same as above)
PAY_6	Repayment status in April, 2005 (scale same as above)
BILL_AMT1	Amount of bill statement in September, 2005 (NT dollar)
BILL_AMT2	Amount of bill statement in August, 2005 (NT dollar)
BILL_AMT3	Amount of bill statement in July, 2005 (NT dollar)
BILL_AMT4	Amount of bill statement in June, 2005 (NT dollar)
BILL_AMT5	Amount of bill statement in May, 2005 (NT dollar)
BILL_AMT6	Amount of bill statement in April, 2005 (NT dollar)
PAY_AMT1	Amount of previous payment in September, 2005 (NT dollar)
PAY_AMT2	Amount of previous payment in August, 2005 (NT dollar)
PAY_AMT3	Amount of previous payment in July, 2005 (NT dollar)
PAY_AMT4	Amount of previous payment in June, 2005 (NT dollar)
PAY_AMT5	Amount of previous payment in May, 2005 (NT dollar)
PAY_AMT6	Amount of previous payment in April, 2005 (NT dollar)
default.payment.next.month	Default payment (1=yes, 0=no)

2. Methodology

Looking at the problem, we could see a potential use of this kind of data that includes how well can we predict, month by month, the likely default payment of our customers?

A review of related research is usually an important first step:

- Helps to determine what kind of information is needed
- Determine if similar questions already been examined

Please refer to the following sections to find more details.

2.1 Data Retrieving:

The Data retrieving process includes the following steps.

1. Importing the data into an object from the UCI repository using the urllib2 library
2. Reading the data from that object by using the function read_excel from the pandas' library.

Using the **shape** function, we have observed that the dataset consists of 30000 rows and 25 columns.

2.2 Data Preparation:

The Data received in the Data Retrieval phase is likely to be “A Diamond in The Rough”. Data preparation is tremendously important because our models will perform better, and we’ll lose less time trying to fix strange output.

This task includes the following steps used for preparing the data which is retrieved from the UCI repository in the above step.

The following pandas’ functions are used to validate the structure, data types and underlying data.

- **dtypes:** to find datatypes of data given in dataset
- **columns:** to find column names in given dataset
- **shape:** to find number of rows and columns presented in dataset
- **head:** to view data present in dataset

Following are the observations:

- Data type of all given columns is integer
- 30000 rows and 25 columns exist

2.1.1 Checking missing (NaN) values

The function "info" is used to examine the existence of nulls and observed that there are no missing values exist in the given dataset.

2.1.2 Checking duplicate rows

The function drop_duplicates() is used for checking the duplicate rows and observed that there are no duplicate rows exist in the given dataset.

2.1.2 Dealing with extra white spaces

A custom function "remove_whitespace" has incorporated into the code to identify and remove the extra white spaces in given dataset.

2.1.3 Dealing with unknown Values

By reviewing the data, we have observed that there are few anomalous things in the following columns:

- EDUCATION has categories of 5 and 6 which are unlabeled, moreover the category 0 is undocumented.
- MARRIAGE has a label 0 that is undocumented
- PAY_N columns have two unlabeled values which are assumed as follows
    - 0 signifies "the customer paid partial bill amount"
    - -2 signifies "the customer doesn’t have credit card".

All these are labeled as "others" appropriately, the category 4 represents "others" in EDUCATION while 3 represents "others" in MARRIAGE.

2.3 Data Exploration

Hypothesis 1

"Males generally have a more tendency to be credible as opposed to Females."

default payment next month	Non-Credible (perc1)	Credible (perc0)
SEX		
Male (1)	0.204628	0.795372
Female (2)	0.177383	0.822617

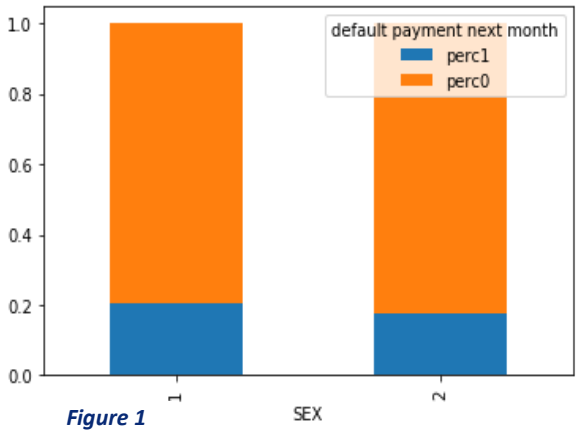


Figure 1

Inference: The hypothesis cannot be rejected as the proportion of defaulted males is SLIGHTLY more than defaulted Females

Hypothesis 2

"More educated individuals generally have a lesser tendency to be credible as opposed to lesser educated individuals."

default payment next month	Non-Credible (perc1)	Credible (perc0)
EDUCATION		
Graduate school (1)	0.154450	0.845550
University (2)	0.202939	0.797061
High school (3)	0.214634	0.785366
Others (4)	0.074074	0.925926

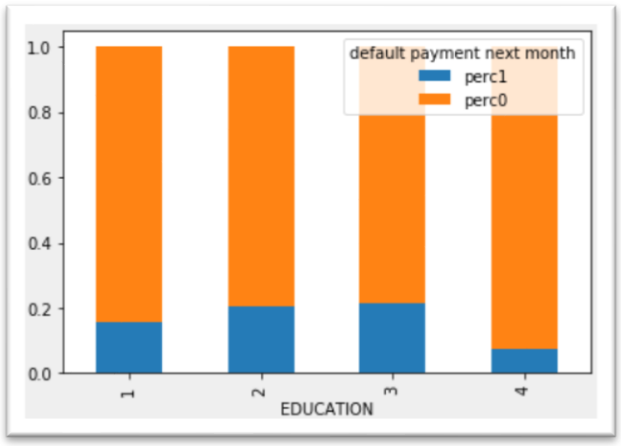


Figure 2

Inference: Hypothesis seems to be correct, as education increases, % of default also increases.

Hypothesis 3

"More MARRIED individuals generally have a lesser tendency to be credible as opposed to UNMARRIED."

default payment next month	Non-Credible perc1	perc0
MARRIAGE		
married (1)	0.203622	0.796378
single (2)	0.175359	0.824641
others (3)	0.234513	0.765487

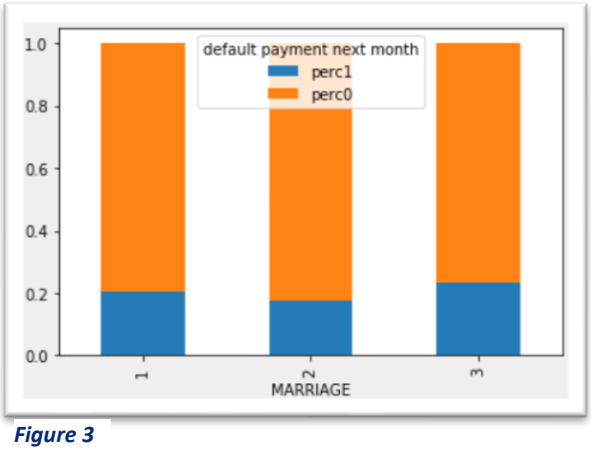


Figure 3

Inference: Hypothesis is rejected

Hypothesis 4

"Limit Balance generally have a more tendency to be higher for more educated individuals."

(1=graduate school, 2=university, 3=high school, 4=others)

Inference: Hypothesis seems to be correct

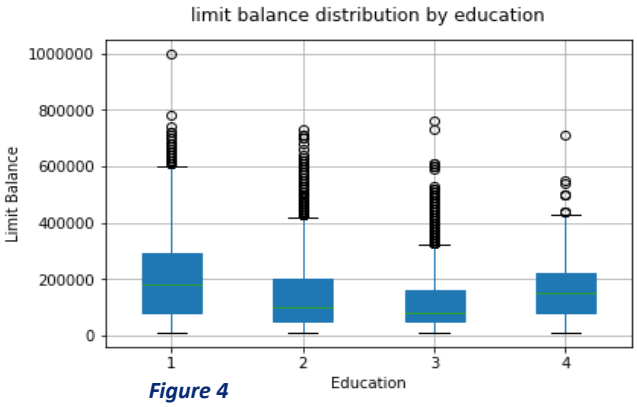
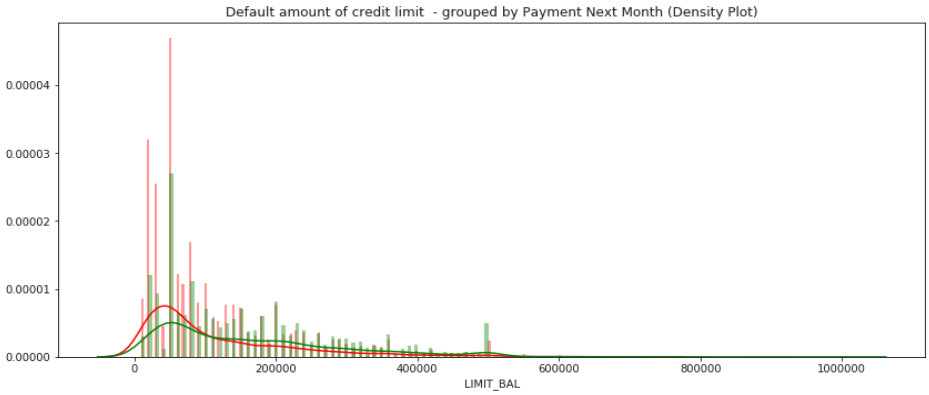


Figure 4

Hypothesis 5

"The chances of being default is higher for lower credit limits."



Inference: Most of defaults are for credit limits 0 - 100,000 (and density for this interval is larger for defaults than for non-defaults).

Figure 5: *Density plot for amount of credit limit (LIMIT_BAL), grouped by default payment next month.*

Hypothesis 6

"Males who got defaulted generally have more credit limit than Females who are defaulted."

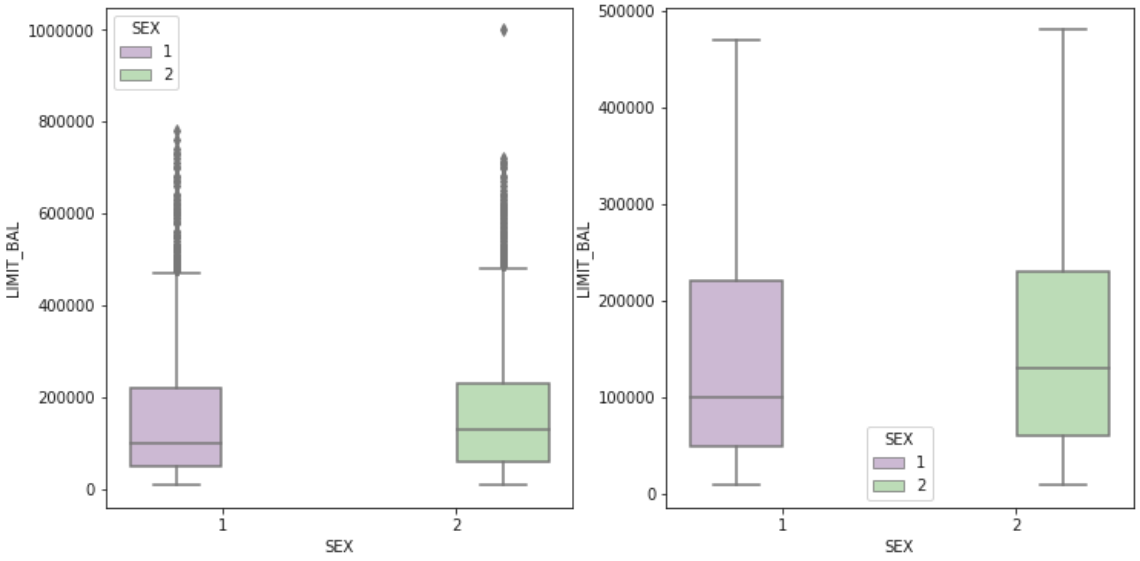


Figure 6: Credit limit distribution vs SEX (1 stands for male and 2 for female)

Inference: Hypothesis Rejected since both Males and Females have similar credit limit whether they got defaulted or not.

Hypothesis 7

"As payment delay increases, the possibility of getting default also increases which makes a customer credible."

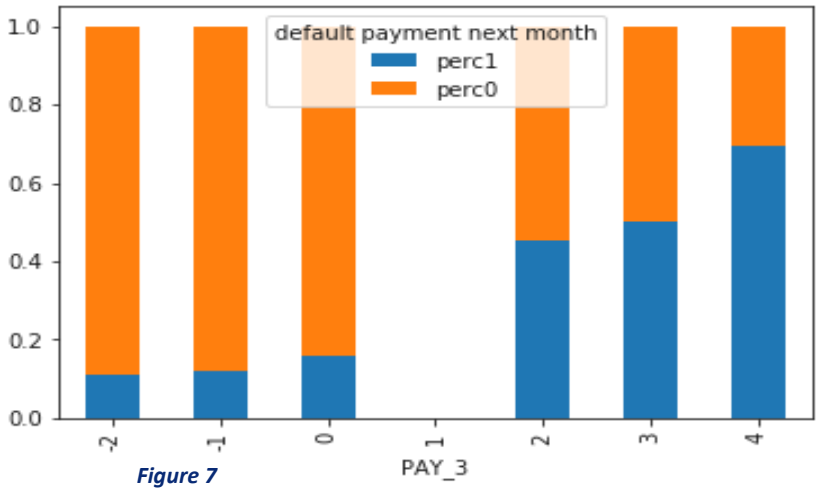


Figure 7

Final Inferences

- Proportion of defaulted males is SLIGHTLY more than Defaulted Females
- As education increases, % of default also increases
- Limit Balance do have a more tendency to be higher for more educated individuals
- Most of defaults are for credit limits 0-100,000 (and density for this interval is larger for defaults than for non-defaults.

2.4 Data Modelling

Classification task is used for modelling this dataset which includes implementing both decision tree classifier and k-nearest neighbor models. The following three different kinds of splits are taken to measure the results to find the best model.

- ✓ 50% for training and 50% for testing;
- ✓ 60% for training and 40% for testing;
- ✓ 80% for training and 20% for testing;

To train and test the models for above splits, “default payment next month” is chosen as the target column while remaining columns (excluding ID) are chosen as features. Apart from above, the feature engineering technique is also used to train & test the models for above splits. This includes

- Decision Tree: Feature Reduction after getting the feature importance scores considering only features having atleast 1% importance.
- KNN : Feature Selection using Hill climbing Technique.

Selecting appropriate values:

Explanation of parameter values chosen for the Decision Tree:

Parameter Name	Value	Explanation
class_weight	balanced	to have proper class weights for both 0s and 1s, since there are a greater number of 0s than 1s.
criterion	gini	for the Gini impurity (default)
max_depth	100	considering the dataset, a moderate value is chosen to avoid any overfitting.
max_leaf_nodes	20	best nodes are defined as relative reduction in impurity.
min_samples_leaf	10	a moderate value is chosen by considering the dataset
min_samples_split	20	this is taken in relative to the min_samples_leaf
splitter	best	default value is chosen.

Explanation of parameter values chosen for the KNN:

Parameter Name	Value	Explanation
k	5	default value is chosen.
weights	distance	to have proper class weights for both 0s and 1s, since there are a greater number of 0s than 1s.
p	1	Considering dataset Manhattan distance is chosen

3. Result

Test the accuracy of the model:

Confusion Matrix of Decision Tree:

Test:Train	*50:50*	*40:60*	*20:80*
Before Feature Engineering	[[8841 2891] [1170 2098]]	[[7898 1510] [1094 1498]]	[[3872 831] [535 762]]
After Feature Engineering	[[8734 2998] [1132 2136]]	[[7898 1510] [1094 1498]]	[[3872 831] [535 762]]

Classification report for Decision Tree:

			Before Feature Engineering				After Feature Engineering			
Test	**Train**		**Precision**	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
50	**50**	0	0.88	0.75	0.81	11732	0.89	0.74	0.81	11732
		1	0.42	0.64	0.51	3268	0.42	0.65	0.51	3268
		micro avg	0.73	0.73	0.73	15000	0.72	0.72	0.72	15000
		macro avg	0.65	0.7	0.66	15000	0.65	0.7	0.66	15000
		weighted avg	0.78	0.73	0.75	15000	0.78	0.72	0.74	15000
40	**60**	0	0.88	0.84	0.86	9408	0.88	0.84	0.86	9408
		1	0.5	0.58	0.53	2592	0.5	0.58	0.53	2592
		micro avg	0.78	0.78	0.78	12000	0.78	0.78	0.78	12000
		macro avg	0.69	0.71	0.7	12000	0.69	0.71	0.7	12000
		weighted avg	0.8	0.78	0.79	12000	0.8	0.78	0.79	12000
20	**80**	0	0.88	0.82	0.85	4703	0.88	0.82	0.85	4703
		1	0.48	0.59	0.53	1297	0.48	0.59	0.53	1297

		micro avg	0.77	0.77	0.77	6000	0.77	0.77	0.77	6000
		macro avg	0.68	0.71	0.69	6000	0.68	0.71	0.69	6000
		weighted avg	0.79	0.77	0.78	6000	0.79	0.77	0.78	6000

Inferences from Decision-Tree result:

- Model accuracy is same before and after the feature engineering, since Decision Tree does the feature selection by itself and the same is proven from the above results.
- Model with Test 40, Train 60 split is the best one to achieve the research goal, among all splits since its accuracy is 78% and predicts a greater number of 1s (Non-Credible customers) with greater precision 0.5, recall 0.58 and f1-score 0.53.
- Confusion matrix for the above-mentioned best model is as follows:

  [[7898 1510]
  [1094 1498]]
- Classification Error Rate: (1510+1094)/(7898+1590+1094+1498) = 0.22

Confusion Matrix of KNN:

Test:Train	*50:50*	*40:60*	*20:80*
Before Feature Engineering	[[10665 1067] [2645 623]]	[[12696 1370] [3151 783]]	[[4291 412] [1037 260]]
After Feature Engineering	[[10419 1313] [2792 476]]	[[8989 419] [1928 664]]	[[4467 236] [948 349]]

Classification report for KNN:

			Before Feature Engineering				After Feature Engineering			
Test	**Train**		**Precision**	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
50	**50**	0	0.8	0.91	0.85	11732	0.79	0.89	0.84	11732
		1	0.37	0.19	0.25	3268	0.27	0.15	0.19	3268
		micro avg	0.75	0.75	0.75	15000	0.73	0.73	0.73	15000
		macro avg	0.58	0.55	0.55	15000	0.53	0.52	0.51	15000
		weighted avg	0.71	0.75	0.72	15000	0.67	0.73	0.69	15000
40	**60**	0	0.8	0.9	0.85	14066	0.82	0.96	0.88	9408
		1	0.36	0.2	0.26	3934	0.61	0.26	0.36	2592
		micro avg	0.75	0.75	0.75	18000	0.8	0.8	0.8	12000
		macro avg	0.58	0.55	0.55	18000	0.72	0.61	0.62	12000
		weighted avg	0.71	0.75	0.72	18000	0.78	0.8	0.77	12000
20	**80**	0	0.81	0.91	0.86	4703	0.82	0.95	0.88	4703
		1	0.39	0.2	0.26	1297	0.6	0.27	0.37	1297
		micro avg	0.76	0.76	0.76	6000	0.8	0.8	0.8	6000
		macro avg	0.6	0.56	0.56	6000	0.71	0.61	0.63	6000
		weighted avg	0.71	0.76	0.73	6000	0.78	0.8	0.77	6000

Inferences from KNN result:

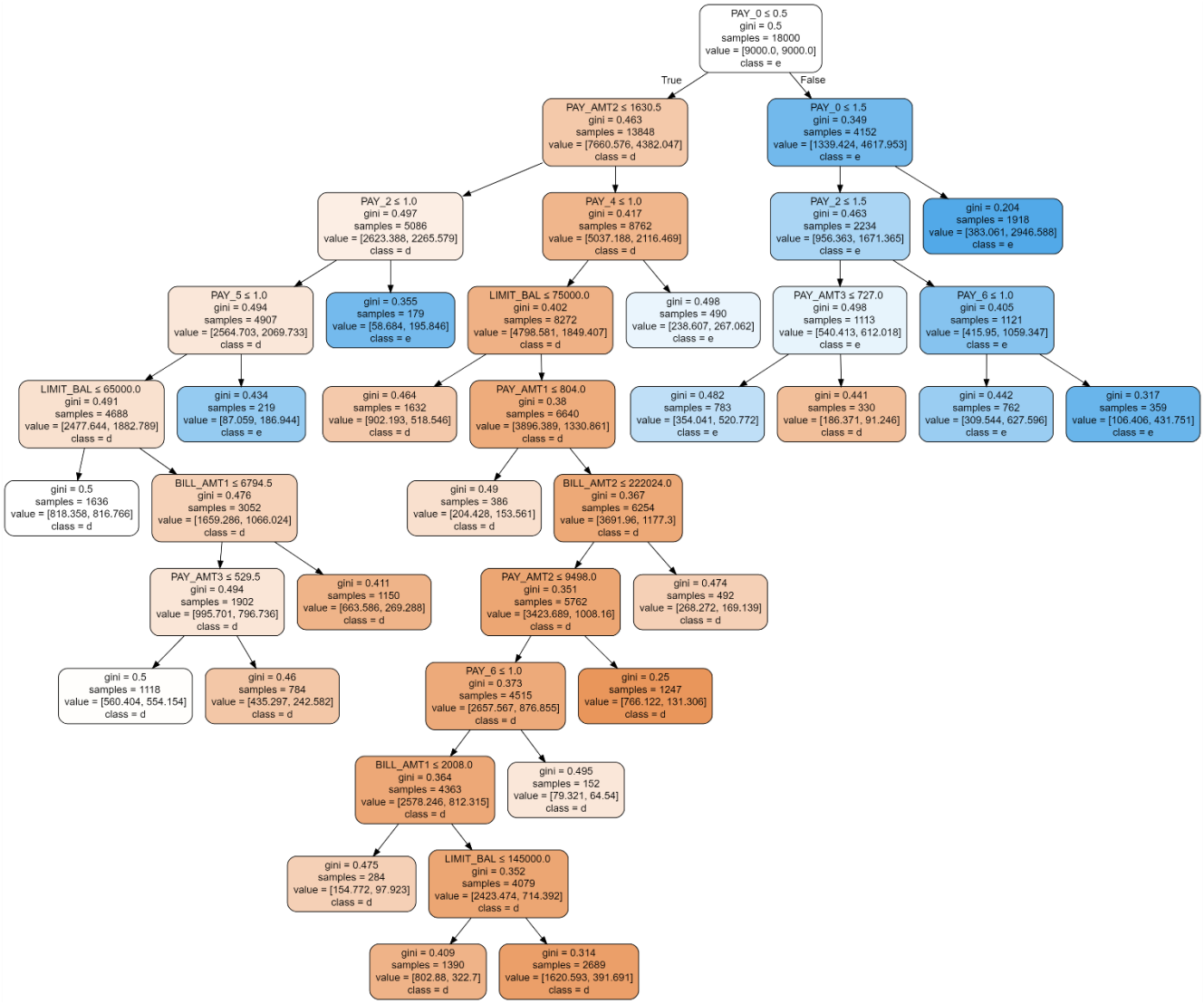
- Model accuracy is better after the feature engineering.

- Model with Test 40, Train 60 split is the best one among all splits since its accuracy is 80.44% with greater precision 0.61, recall 0.26 and f1-score 0.36.

- Confusion matrix for the above-mentioned best model is as follows:

  [[8989 419]
  [1928 664]]

- Classification Error Rate: (419+1928)/(8989+419+1928+664) = 0.1955



Decision Tree

4. Discussion:

Amongst all the splits mentioned above, Decision Tree Model with Test 40, Train 60 split, is the best one to achieve the research goal, due to its higher accuracy of 78%, and it predicts a more significant number of 1s (Non-Credible customers).

Although the K-Nearest Neighbor model results have better accuracy than the Decision Tree model, a greater number of 1s are not predicted as it is susceptible to class imbalance.

Moreover, we have re-trained the model with five months data and found that the accuracy is same as the actual six months data used in this research.

5. Conclusion:

The Decision Tree Model results are outweighed the K-Nearest Neighbor (KNN) Model results due to the class imbalance. Using the best Decision Tree Model explained above, the predictive accuracy of default is achieved with a precision of 50% for the given dataset.

Although the result is not entirely accurate, it still provides better insights to financial institutions to identify risky customers. As there is always a room for the improvement, it is worthwhile continuing this type of research with any other better techniques as we learn in the future.

6. References

1. Credit Risk Wikipedia available at https://en.wikipedia.org/wiki/Credit_risk

2. https://stackoverflow.com/questions/33788913/pythonic-efficient-way-to-strip-whitespace-from-every-pandas-data-frame-cell-tha

3. Practical data science lecture week 1
induction at https://rmit.instructure.com/courses/51550/modules/items/1647272

4. Practical data science lecture week 6
available at https://rmit.instructure.com/courses/51550/files/7073031?module_item_id=1647275

5. Practical data science tutorial week 7
available at https://rmit.instructure.com/courses/51550/modules/items/1647272

6. Archive.ics.uci.edu. (2019). UCI Machine Learning Repository: Default credit card clients. [online]
available at: http://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients