COSC2670: Practical Data Science

Assignment2 Project Report

A classification model for finding default on credit card payment

By

Neeraj Sehrawat (s3711712)

And

Ayaz Mujawar(s3751555)

Contents

1. Abstract

The aim of this report is to find a suitable classification model to predict the whether a person would default on the credit card payment, based on certain socio-economic conditions, with accuracy and efficiency. Two classification models namely KNearestNeighbors (abbreviated as KNN) and Decision Tree were adopted and trained by 50%, 60% and 80% of the 30,000 candidate samples. Overall, the results indicate that KNN model gives the highest precision score at 60:40 split. The report concludes that KNN is the best performing model between the two. It is recommended to increase the sample size in order to mitigate the highly imbalanced distribution of the target class (Yes and NO) in order to increase the accuracy and efficiency of the model.

2. Introduction:

Banks and other financial institutions (commonly known as 'lenders') issue credit card and define the credit limit based on certain socio-economic conditions of the customer(borrowers). A default on credit is when one fails to pay an expected debt on credit card repayment. It is an indicator of a borrower's financial stress and lenders need to be proactive in credit monitoring to identify financial stress at an early stage rather wait for a borrower to default. Early identification of stress would provide enough time for lenders to put in place the required resolution plan.[1] In past few years, lenders have used a range of methods to distinguish the defaulters from non-defaulters, but the accuracy of results and productivity is low. Since the advent of new techniques and technology in data analysis, crazy big volume of client's information is captured and an on-time analysis for this big volume of data is therefore in demand.

Given this, the practical data science techniques are the best time-efficient and accurate-wise tool to tackle this important task, guaranteed a much higher accuracy of identification of default on payment. This report is based on the hypothesis that there is an ideal classification model that can greatly increase the accuracy and efficiency of the process to distinguish defaulters and non-defaulters. Two modern data science models are applied, namely the KNearestNeighbors and Decision Tree. Feature selection and data exploration is also covered in detail to display insights from the Credit Card default dataset.

3. Methodology

3.1 Data Retrieval

The dataset, used for classification approach, "default of credit card clients" is provided by UCI ML repository [2] where 23 descriptive features are provided for customers exclusive of an index number and a binary target variable for default payment (where Yes = 1, No =2). The dataset contains both the numerical and categorical features with 30,000 observations and 25 columns (including index and the target feature.)

Attribute Information:

Default.payment.next.month : Target feature with binary variable, default payment (Yes = 1, No = 0)

The following 23 variables are used as descriptive features:

LIMIT_BAL: Amount of the given credit (NT dollar): It includes both the individual consumer credit and his/her family (supplementary) credit.

SEX: Gender: 1 = male; 2 = female

EDUCATION : Education: 1 = graduate school; 2 = university; 3 = high school; 4 = others

MARRIAGE: Marital status: 1 = married; 2 = single; 3 = others

AGE: Age: in years

PAY_0. . . PAY_06 : **History of past payment**: Past monthly payment records (from April to September 2005) as follows:

X6 = the repayment status in September 2005, X7 = the repayment status in August 2005; . . .; X11 = the repayment status in April, 2005.

The measurement scale for the repayment status is: -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.

BILL_AMT1. . . BILL_AMT6 : **Amount of bill statement (NT dollar):** X12 = amount of bill statement in September 2005; X13 = amount of bill statement in August 2005; . . .; X17 = amount of bill statement in April, 2005.

PAY_AMT1. . . PAY_AMT6: **Amount of previous payment (NT dollar**) : X18 = amount paid in September, 2005; X19 = amount paid in August, 2005; . . .;X23 = amount paid in April, 2005.

In the following sections regarding data exploration, the information of the attributes will be more clearly explained.

3.2 Data Exploration

In this section, there are two goals. First, fully understand the characteristics of the data i.e. Type of values a feature can take, the ranges into which these values fall and how the values in dataset are distributed across the range. The second goal is to determine whether the data suffer from any data quality issues (missing values, outliers) or not. To achieve the first goal, a tool called data quality report is used which shows a standard measure of central tendency and variation of each numerical feature. For the convenience of understanding, the data is prepared by refining the unlabeled data and changing the name of one column. Furthermore, various graphs are used to understand how the values for a feature are distributed across the range.

3.2.1Data Quality Report:

	count	mean	std	min	25%	50%	75%	max
LIMIT_BAL	30000.0	167484.322667	129747.661567	10000.0	50000.00	140000.0	240000.00	1000000.0
PAY_1	30000.0	-0.016700	1.123802	-2.0	-1.00	0.0	0.00	8.0
PAY_2	30000.0	-0.133767	1.197186	-2.0	-1.00	0.0	0.00	8.0
PAY_3	30000.0	-0.166200	1.196868	-2.0	-1.00	0.0	0.00	8.0
PAY_4	30000.0	-0.220667	1.169139	-2.0	-1.00	0.0	0.00	8.0
PAY_5	30000.0	-0.266200	1.133187	-2.0	-1.00	0.0	0.00	8.0
PAY_6	30000.0	-0.291100	1.149988	-2.0	-1.00	0.0	0.00	8.0
BILL_AMT1	30000.0	51223.330900	73635.860576	-165580.0	3558.75	22381.5	67091.00	964511.0
BILL_AMT2	30000.0	49179.075167	71173.768783	-69777.0	2984.75	21200.0	64006.25	983931.0
BILL_AMT3	30000.0	47013.154800	69349.387427	-157264.0	2666.25	20088.5	60164.75	1664089.0
BILL_AMT4	30000.0	43262.948967	64332.856134	-170000.0	2326.75	19052.0	54506.00	891586.0
BILL_AMT5	30000.0	40311.400967	60797.155770	-81334.0	1763.00	18104.5	50190.50	927171.0
BILL_AMT6	30000.0	38871.760400	59554.107537	-339603.0	1256.00	17071.0	49198.25	961664.0
PAY_AMT1	30000.0	5663.580500	16563.280354	0.0	1000.00	2100.0	5006.00	873552.0
PAY_AMT2	30000.0	5921.163500	23040.870402	0.0	833.00	2009.0	5000.00	1684259.0
PAY_AMT3	30000.0	5225.681500	17606.961470	0.0	390.00	1800.0	4505.00	896040.0
PAY_AMT4	30000.0	4826.076867	15666.159744	0.0	296.00	1500.0	4013.25	621000.0
PAY_AMT5	30000.0	4799.387633	15278.305679	0.0	252.50	1500.0	4031.50	426529.0
PAY_AMT6	30000.0	5215.502567	17777.465775	0.0	117.75	1500.0	4000.00	528666.0

Missing values-There are no missing values in the dataset.
Outliers- Comparing the gaps between the median, min value, max value,1st quartile and 3rd quartile values to check for unusual values. For this dataset the minimum value of "BILL_AMT6" feature seems unusual and is likely to be an outlier as the gap between the 1st quartile and min. value is noticeably large than the gap between the 1st quartile and the median. similarly, for "PAY_AMT2" maximum value looks unusual as the gap between 3rd quartile and max values is noticeably large than the gap between median and 3rd quartile. For this assignment, dealing with outliers and data normalization is not covered.

3.2.2 Data Preparation

(a) **Changed Column names**: (1) default_payment_next_month → default_pay
(2) Pay_0 → Pay_1

(b) **Unlabelled data**: Marital status – 0 coverted to 3(others)
Pay_0 to Pay_6 : -2,-1 are converted to) for duly paid

3.2.3Data Visualization

Firstly, the distribution of single variable is visualized, with the numeric features in a density graph and the discrete class variable in a bar chart. Secondly, the relationship between each feature agai nst the class variable is illustrated. The heat map showing the correlation amongst all the variables is also depicted.

(a) Individual Feature exploration: In this section, the distribution of single features(both numeric(float) and categorical(int)) including the target feature is visualized using multitude of graphs.

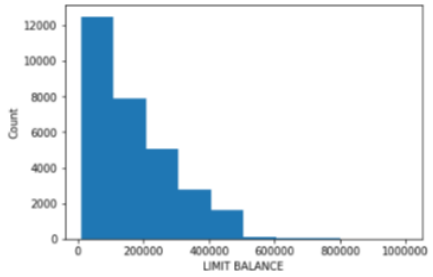

Figure 1 limit balance

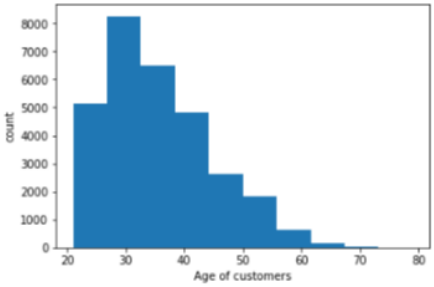

Figure 2 Age

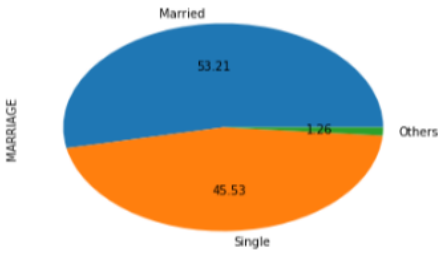

Figure 3 Marital Status

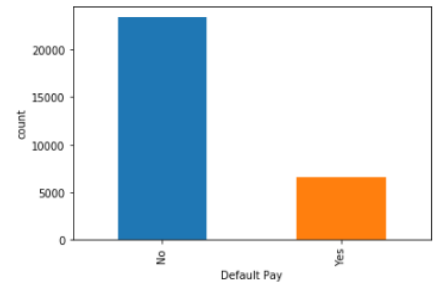

Figure 4 Default Pay

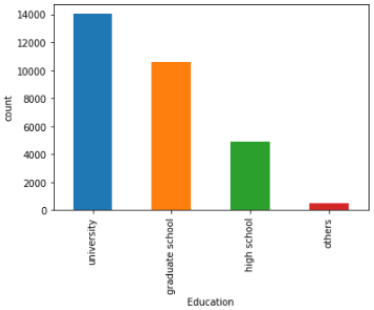

Figure 5 Education

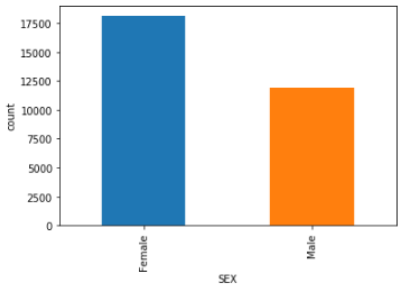

Figure 6 Sex

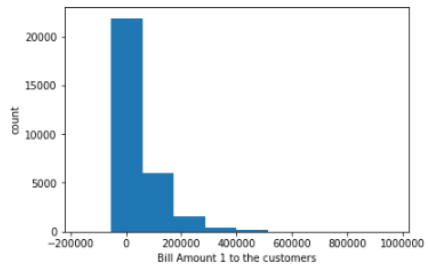

Figure 7 Bill_Amount

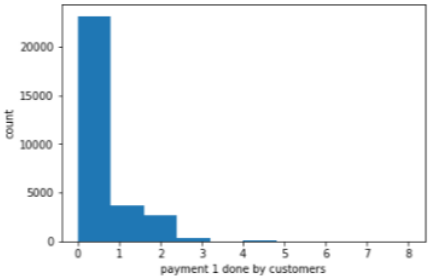

Figure 8 Pay_1

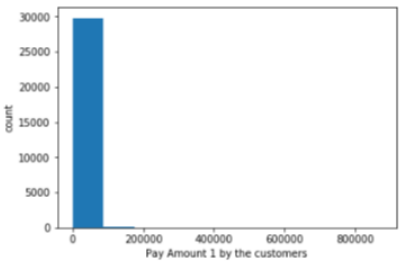

Figure 9 Pay_Amount

Fig.1 – The highest number of people have credit amount limit less than 200,000.
Fig.2- most of the customers are aged between 25-50 years of age
Fig.3- More than half of the clients are married
Fig.4- Majority of clients are not defaulters. The defaulters could be somewhere around 25%.
Fig. 5 – The highest number of clients are university graduates.
Fig. 6 – More than half of the clients are females

Fig.7- Majority of the bill amount are somewhere between 0 to 100000
Fig. 8 – Most of the people pay their dues duly on time and almost all of them within 3 months
Fig. 9 – Nothing can be deciphered from pay amount alone. This needs to seen in conjunction with other allied features.

(b) Paired feature Exploration:

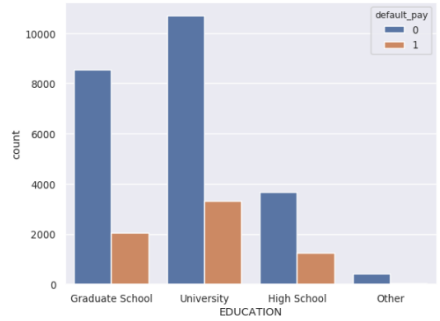

Figure 10

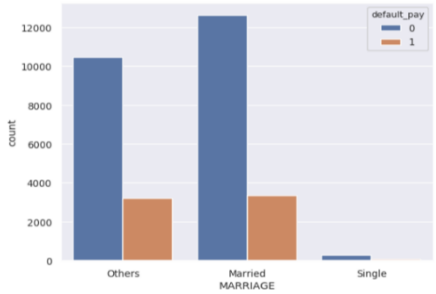

Figure 11

Figure 10: This shows that university graduate default more but considering their total number it is not that high.
Fig. 11: Married people default less

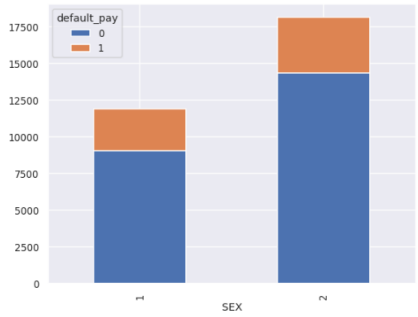

Figure 12

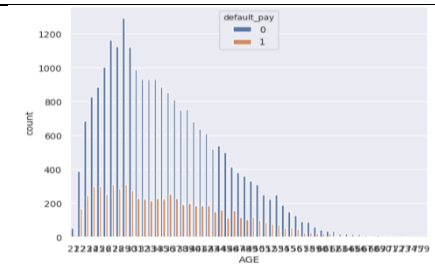

Figure 13

Fig. 12: female default less in compare to their males counterpart
Fig. 13: Young age people before 50 default more.

From Fig. 14, we can infer that the bill_amount has the highest positive correlation with each other and with the target variable followed by the history of past payment. All others are slightly correlated with the target feature and with other descriptive features.

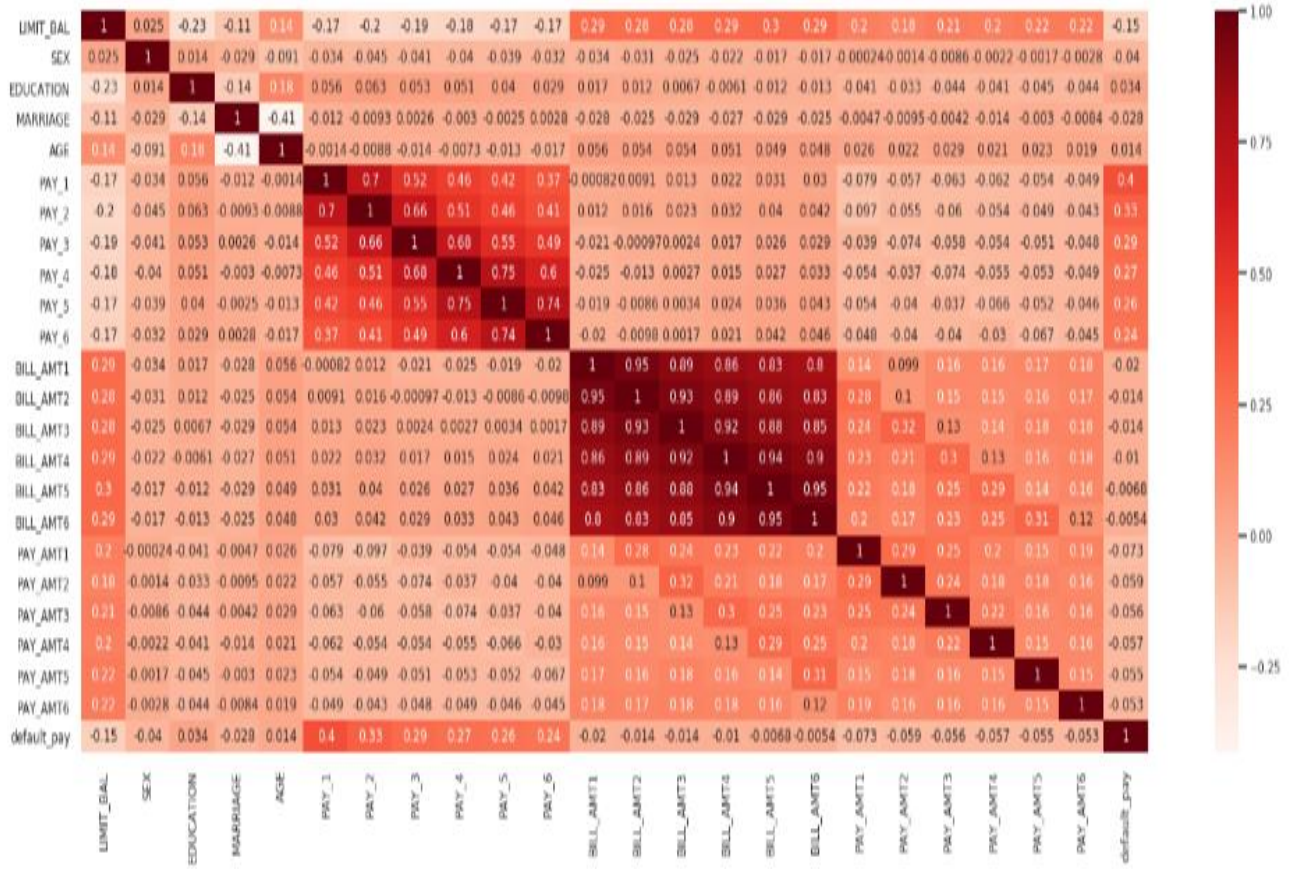

Fig 14

3.3 Data Modelling

3.3.1 Modelling flow:

The classification approach for modelling is used and the dataset is split into training set and test set at the following ratio's: 50:50 , 60:40 and 80:20. For each split, two classifiers, namely- KNN and decision tree are used. For each classifier, first the classifier is performed with the default parameters and then the parameters are tuned. One parameter is tuned at a time and three parameters were tuned in total. While tuning the parameters, precaution was taken to not lead to overfitting. Once the classifiers are performed with tuned parameters, the hill-climbing method of feature selection is performed for the same classifier at three splits ratios. The performance and accuracy of all the iterations is recorded with the respective confusion matrix and other performance assessing statistics like Precision, Recall and F-1. A comparison is made iteratively to choose the best performing classifier **(see fig.15 below**). First, the best performing model from the respective classifier type is chosen from the specific split. This gave us a total of 6 model, 3 KNN (1 best each split) and 3 Decision Tree (1 best from each split). These three are again sorted to settle for one best in each category. In the end, just one end model is selected.

3.3.2 K Nearest Neighbor (KNN) Modelling:

The theory behind the KNN is to find the number of data points closest in distance to the new data point and perform the prediction of label from these. KNN is established on feature similarity i.e. Classifying the given data point based on the feature resemblance with training data samples. The "KNeighborsClassifier" can be imported from "sklearn.neighbours" package.

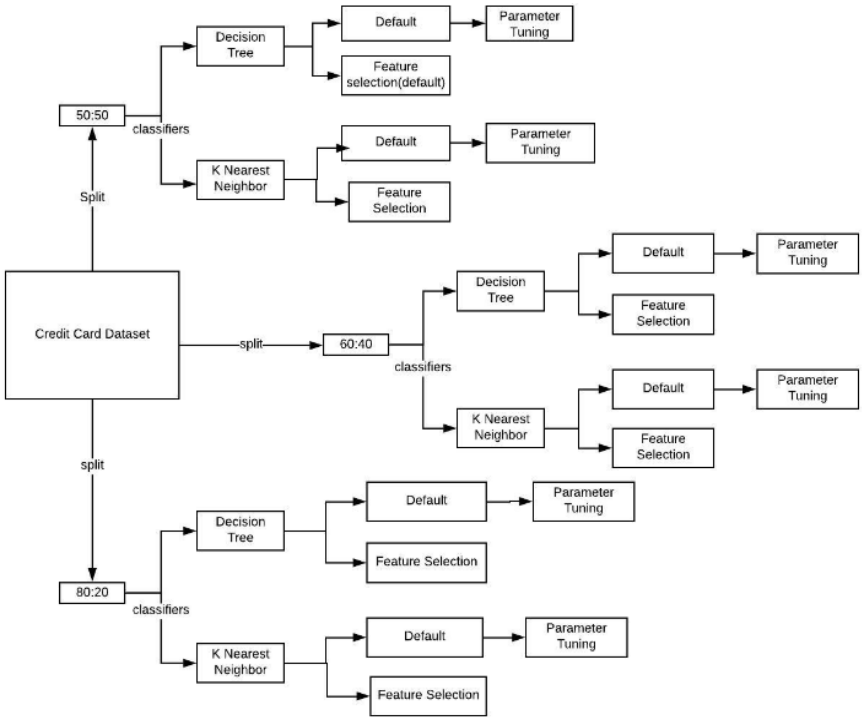

Fig. 15

Models: (a) **Default-** Algorithm with default parameters (n_neighbours=5, weights=uniform, p=2, metric=minkowski)

(b) Tuned Parameter- Performed parameter tuning for no. of neighbors. The value of K is important as the small value of k means that noise will have a higher influence on the result and a large value make it computationally expensive. As the target feature is binary (dichotomous), the algorithm is tried with the next big odd number of neighbors like 3,5,7. The even number of neighbors do not work well when the target class is even.

(c) Selected Features- The hill climbing method is used for feature selection for every split and the classifier is performed on the features with the best scores.

The quality of the models are assessed on the basis of confusion matrix, classification error rate , Precision, recall, and F1 Score which are displayed in the results later.

3.3.3 Decision Tree Modelling:

The DT algorithm repeatedly divides the data into sub-parts based on the information gain. The default values of the parameter control the size of the tree leads to unpruned trees. The default decision tree for this dataset is very large. The "DecisionTreeClassifier" can be imported from "sklearn.tree" package.[3,8,9]

Models: Default- Algorithm with default parameters (criterion =gini)

Tuned Parameter- Performed parameter tuning for parameters criterion, max_features. The default takes criterion = gini which divides the data based on Gini impurity while the criterion = entropy splits based on information gain. The max_features tunes the number of features that a classifier considers when looking for the best split. The rule of thumb is to go for the square root of the total number of features. It is generally good to check up to 30-40% of the total number of features. There are a total of 25 features hence the max_features = 5 were tuned with both gini and entropy criterion.

Selected Features-After executing the hill climbing method for feature selection, the classifier is performed on the selected features based on the best score for each split.

The quality of the models is assessed based on confusion matrix, classification error rate, Precision, recall, and F1 Score which are displayed in the results later.

3.4 Results

The table below shows that feature selection really worked very well on all the splits in all the classifier. The performance of all the classifiers are measured in terms of the precision, recall, F-1 score of both the defaulters and non-defaulters. The overall accuracy and the classification error rate are also considered. While the overall accuracy is between 70 to 80 percent, the classification error rate remains same for all the classifiers.

After a thorough analysis of the top 6 models, it can be inferred that **KNN is performing better than Decision tree in all the splits** on this dataset**.** It can also be confirmed from the confusion matrix for the top six models presented below.

Split	Classifier	Parameter	Prec.(yes)	Prec.(no)	Recal (yes)	Recal(no)	F-1(yes)	F-1(no)	Accuracy	C.Error Rate
50:50	K NN	Default	0.37	0.8	0.19	0.91	0.25	0.85	0.8	0.21
		K = 3	0.34	0.8	0.22	0.88	0.27	0.84	0.7	0.21
		K= 5	0.37	0.8	0.19	0.91	0.25	0.85	0.8	0.21
		K=7	0.39	0.8	0.16	0.93	0.23	0.86	0.8	0.21
		Feature Selection	**0.52**	**0.82**	**0.24**	**0.94**	**0.32**	**0.87**	**0.8**	**0.21**
	D. Tree	Default	0.38	0.83	0.42	0.81	0.4	0.82	0.7	0.21
		Criterion: Entropy	0.38	0.83	0.41	0.81	0.39	0.82	0.7	0.21
		max_features= 5	0.39	0.83	0.41	0.81	0.4	0.83	0.7	0.21
		Ent.+ max_features=	0.38	0.83	0.41	0.81	0.39	0.82	0.7	0.21
		Feature Selection	**0.51**	**0.83**	**0.32**	**0.92**	**0.39**	**0.87**	0.7	0.21
60:40	K NN	Default	0.37	0.8	0.19	0.91	0.25	0.85	0.8	0.21
		K=3	0.34	0.8	0.23	0.88	0.27	0.84	0.7	0.21
		K=5	0.37	0.8	0.19	0.91	0.25	0.85	0.8	0.21
		K=7	0.39	0.8	0.16	0.93	0.23	0.86	0.8	0.21
		Feature Selection	**0.58**	**0.83**	**0.27**	**0.89**	**0.37**	**0.88**	**0.8**	**0.21**
	D. Tree	Default	0.37	0.83	0.41	0.81	0.39	0.82	0.7	0.21
		Criterion: Entropy	0.38	0.83	0.38	0.83	0.38	0.83	0.7	0.21
		max_features = 5	0.37	0.83	0.41	0.81	0.39	0.82	0.7	0.21
		Ent.+max_features =	0.39	0.84	0.42	0.82	0.4	0.83	0.7	0.21
		Feature Selection	**0.52**	**0.83**	**0.32**	**0.92**	**0.4**	**0.87**	0.8	0.21
80:20	K NN	Default	0.4	0.81	0.19	0.92	0.26	0.86	0.8	0.21
		K=3	0.33	0.8	0.22	0.88	0.27	0.84	0.7	0.21
		K=5	0.4	0.81	0.19	0.92	0.26	0.86	0.8	0.21
		K=7	0.4	0.8	0.16	0.93	0.23	0.86	0.8	0.21
		Feature Selection	**0.58**	**0.82**	**0.26**	**0.95**	**0.36**	**0.88**	**0.8**	**0.21**
	D. Tree	Default	0.4	0.84	0.43	0.82	0.41	0.83	0.7	0.21
		Criterion: Entropy	0.4	0.84	0.41	0.83	0.41	0.83	0.7	0.21
		max_features = 5	0.4	0.84	0.44	0.82	0.42	83	0.7	0.21
		Ent.+max_features =	0.38	0.83	0.41	0.82	0.39	0.83	0.7	0.21
		Feature Selection	**0.53**	**0.83**	**0.32**	**0.92**	**0.4**	**0.87**	0.8	0.21

After a thorough analysis of performances of all the models, it can be concluded that the **KNN at 60:40 ratio/split is performing best** amongst all other classifiers**.**

Confusion matrix:

CONFUSION MATRIX : [[11016 716] [2498 770]] **50:50 KNN**	CONFUSION MATRIX : [[10739 993] [2222 1046]] **50:50 Decision tree**	CONFUSION MATRIX : [[8886 522] [1883 709]] **60:40 KNN**
CONFUSION MATRIX : [[8627 781] [1751 841]] **60:40 Decision Tree**	CONFUSION MATRIX : [[8917 491] [1907 685]] **80:20 KNN**	CONFUSION MATRIX : [[4330 373] [876 421]] **80:20 Decision Tree**

3.5 Discussion:

The Results section of the report shows that KNN outperformed the Decision tree in terms of its performance. We expected that the decision tree classifier might be appropriate considering the categorical features in the data however considering the precision rate for the default payment, KNN oat 60:40 split outperformed decision tree due to its distance method and k value.

One very important thing that was considered while modelling decision trees and specially during the parameter tuning was **Overfitting.** Overfitting occurs when a model is excessively complex, such as having too many parameters relative to the number of observations.
The overfitted model over train and do not generalize well. On the contrary, sometimes the model remains does not capture the entire relation and this leads to underfitting and such models also do not generalize well due to high variance. Hence, special care should be taken while parameter tuning (specially in case of decision trees) to avoid the underfitting or overfitting and appropriately train the model.

3.6 Conclusion:

After applying the two classification models at three split ratios', it turns out to be the KNN model gives the highest precision for the default payment. In this context, the KNN model with 7 best features is the optimal model to classify the default payment. However, the optimal model or the optimal parameter values in this research may not be applicable to other set of Credit Card analysis, due to the highly imbalanced nature of the Credit card default, more experience to train a model that can reduce the impact of imbalance to the minimum is needed, and this might be a new research question opened up.

3.7 References:

[1] https://www.thehindubusinessline.com/money-and-banking/to-prevent-default-lenders-must-be-proactive-in-monitoring-credit-rbi/article24812961.ece

[2] https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients

[3] scikit learn. (n.d.). DecisonTreeClassifier. Retrieved from scikit learn: http://scikitlearn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

[4] 4. scikit-learn. (n.d.). Nearest Neighbours. Retrieved from scikit learn: http://scikitlearn.org/stable/modules/neighbors.html

[5] Week 4 lecture slides/tutorial

[6] Week 5 lecture slides/tutorial

[7] Week 6 lecture slides/tutorial