CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING

RICHARD (2K18/SE/106)

AND

SHIVOM (2K18/SE/119)

INTRODUCTION

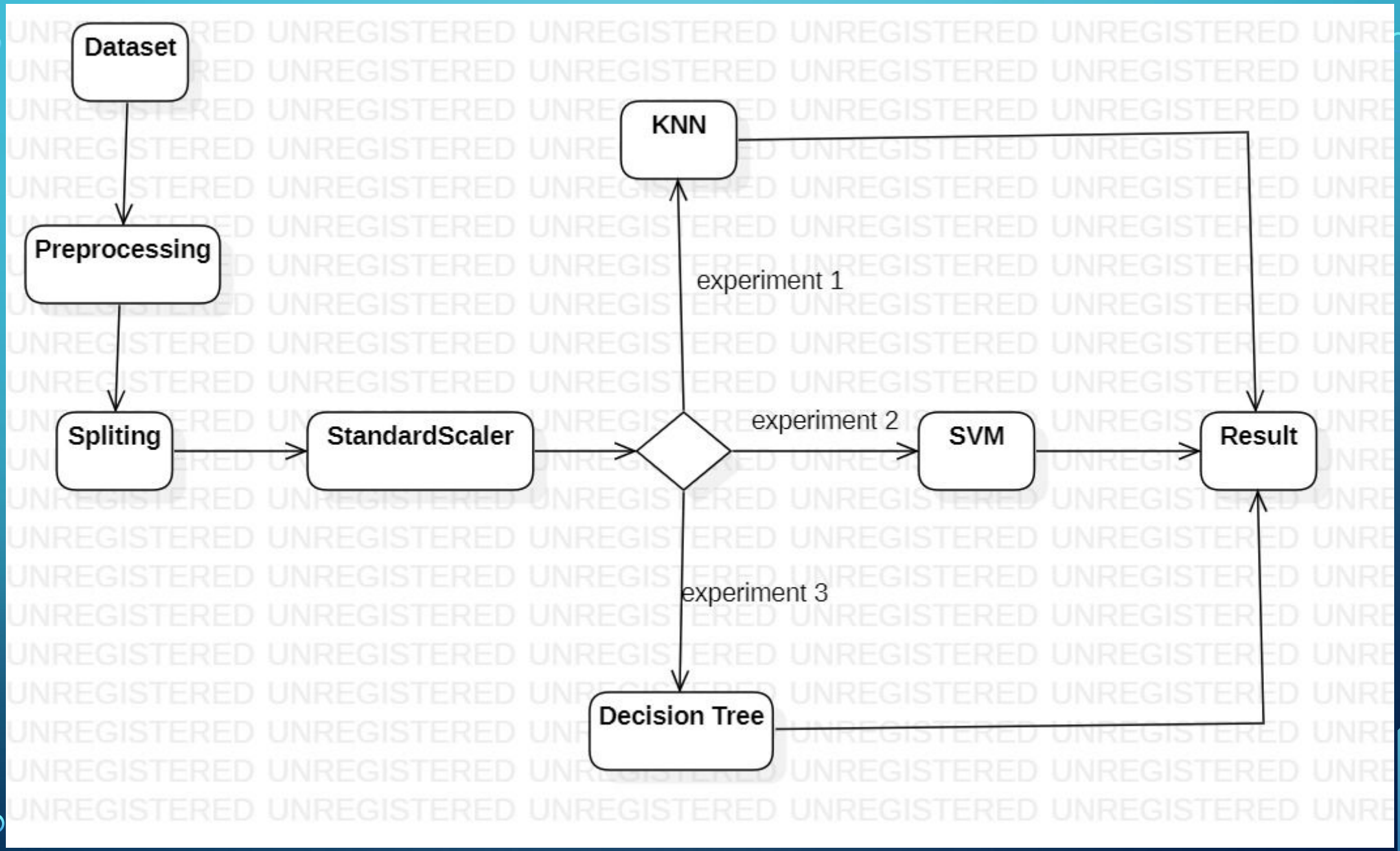
In this era of online payment the chances of fraud credit card transactions is very high due to lost or leak of credit card details .Therefore it is vital that credit card companies are able to identify fraudulent credit card transactions so that customers are not charged for items that they did not purchase. Such problems can be tackled with Data Science and its importance, along with Machine Learning. In this paper we have conducted a comparative study determining anomalies in Credit Card Fraud Detection between three different classification models. The Credit Card Fraud Detection Problem includes modelling past credit card transactions with the data of the ones that turned out to be fraud. This model is then used to recognize whether a new transaction is fraudulent or not. Our objective here is to detect 100% of the fraudulent transactions while minimizing the incorrect fraud classifications. Credit Card Fraud Detection is a typical sample of classification. In this process, we have focused on analysing and pre-processing of data sets as well as the deployment of multiple anomaly detection algorithms.

DATASET

The dataset we have used is from kaggle with 143.48 mbs of size.
The datasets contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.
It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

EXPERIMENT DESIGN

- We have used the dataset mentioned previously and applied machine learning techniques on it.
- We checked the data for null values and missing values.
- We have divided the dataset into test and train samples using test/train split method .
- 80% transactions of the above dataset are as the training dataset, and the rest as the testing dataset.
- We have applied Standard Scaler on test and train samples of the dataset.


Dataset
Preprocessing
Spliting
StandardScaler
KNN
experiment 1
experiment 2
SVM
Result
experiment 3
Decision Tree

MACHINE LEARNING MODELS

We have applied three different machine learning classification model in this study.

- Decision Tree
- KNN(K-nearest neighbors)
- SVM(Support vector machine)

Decision Tree Model:-

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

We have implemented decision Trees algorithm with entropy as the criterion for the prediction of fraud.

$$H(s) = -P_{(+)} log_2 P_{(+)} - P_{(-)} log_2 P_{(-)}$$

$$Here\ P_{(+)} / P_{(-)} = \%\ of\ +ve\ class\ /\ \%\ of\ -ve\ class$$

KNN Model :-

It is one of the most basic yet essential classification algorithms in Machine Learning. It is a Lazy algorithm as KNN does not have a training step. All data points will be used only at the time of prediction. With no training step, prediction step is costly though.

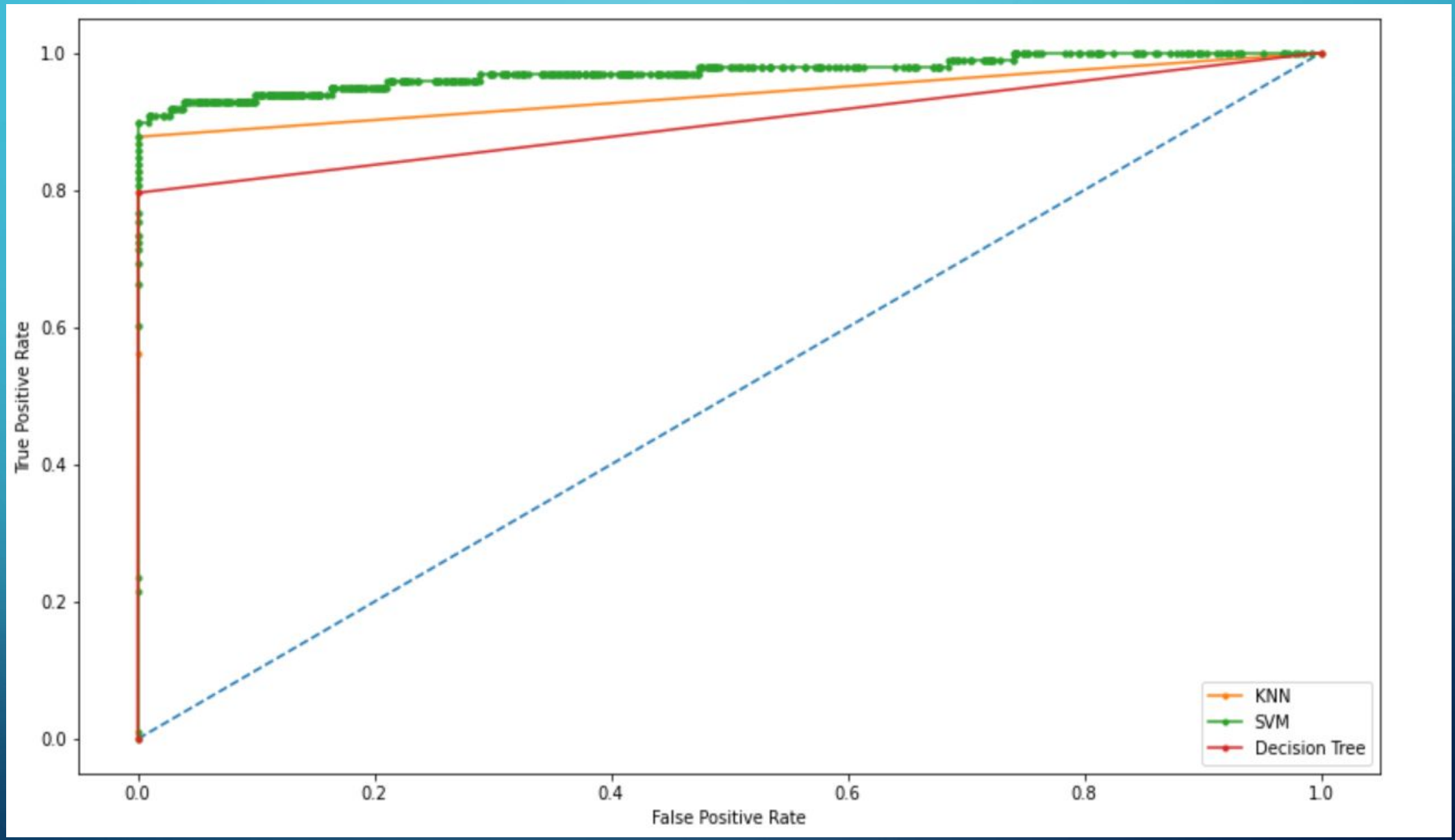
We have implemented KNN algorithm with value of k as 5 because we see a high error rate for test set when K=1. Hence we can conclude that model over fits when k=1. For a high value of K, we see that the F1 score starts to drop. The test set reaches a minimum error rate when k=5.

SVM Model :-

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n- dimensional space into classes so that we can easily put the new data point in the correct category in the future.
We have implemented Support vector machine algorithm with RBF Kernel Function.

RESULTS

Models	Accuracy (%)	Recall (%)	precision (%)	F1 score	AUC score
KNN	99.95	79.59	93.975	0.861	0.939
SVM	99.94	69.38	95.774	0.804	0.972
Decision tree	99.92	79.59	78.0	0.787	0.898


True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
KNN
SVM
Decision Tree

CONCLUSIONS

AS we can see from the results of the study that :-

- Accuracy of KNN and SVM is nearly equal but better than that of decision tree.
- KNN model has slightly more recall and f1 score and both KNN and SVM has better values than decision tree classifier.
- SVM has better auc score and precision than both KNN and decision tree.
- Hence we conclude that it is better to use SVM classifier among the all we considered in this study as svm has a high accuracy of 99.94,a high precision of 95.77 and covers a larger area under the curve on roc curve.