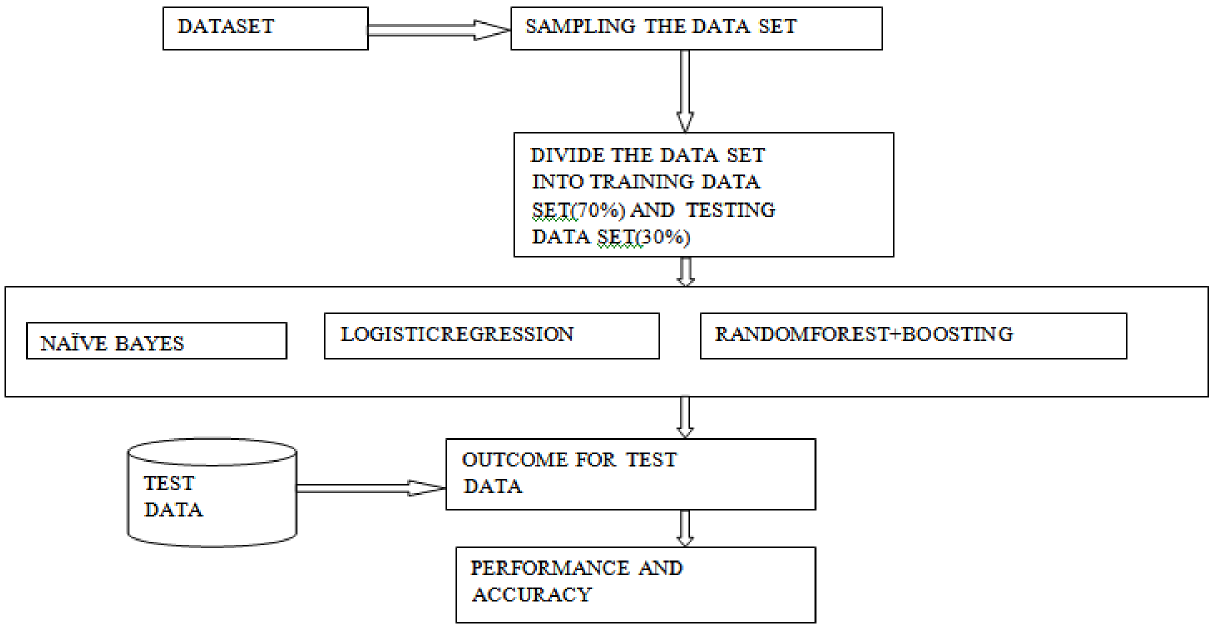
An integrated regression model for credit Credit card fraud detection coping with missing value imputation

[1]Diksha Jha and [2]Ranjan Kumar Behera
[1] [2]*Department of Mathematics, Birla Institute of Technology, Mesra, Ranchi, India,*[2]*Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Ranchi, India*
jha556626@gmail.com, ranjan.behera@bitmesra.ac.in

In this paper we have explained the concept of frauds related to credit cards. Here we implement different machine learning algorithms on an imbalanced dataset such as ***logistic regression, naïvebayes, random forest*** *with ensemble classifiers using boosting technique. An extensive review is done on the existing and proposed models for credit card fraud detection and has done a comparative study on these techniques. So Different classification models are applied to the data and the model performance is evaluated on the basis of quantitative measurements such as accuracy, precision, recall, f1 score, support, confusion matrix. The conclusion of our study explains the best classifier by training and testing using supervised techniques that provides better solution.*

Workflow of the Model-



Dataset: In this paper credit card fraud detection dataset was used,which can be downloaded from Kaggle.This dataset contains transactions,occurred in two days,made in September 2013 by European cardholders. The dataset contains 31 numerical features. Since some of the input variables contains financial information, the PCA transformation of these input variables were performed in order to keep these data anonymous. Three of the given features weren't transformed. Feature "Time" shows the time between first transaction and every other transaction in the dataset. Feature "Amount" is the amount of the transactions made by credit card. Feature "Class" represents the label and takes only 2 values: value 1 in case fraud transaction and 0 otherwise.

Sampling: Further the data set is minimized to 560 transactions.Where 228 fraud and 332 normal transactions.

Divide the dataset:The dataset is divided into trained data set and test data set. 70% of the data set is under training and the remaining 30% is under testing.Here we are using some supervised machine learning algorithms. The algorithms areNaive Bayes**,** Logistic RegressionandRandom Forest with boosting technique.

Naïve Bayes: Bayes theorem: Bayes theorem find probability of event occurring given probability of another event that has been alreadyoccurred.Naïve Bayes algorithm is easy and fast. This algorithm need less training data and highlyscalable

P (A/B) = (P (B/A) P (A)) / P (B)

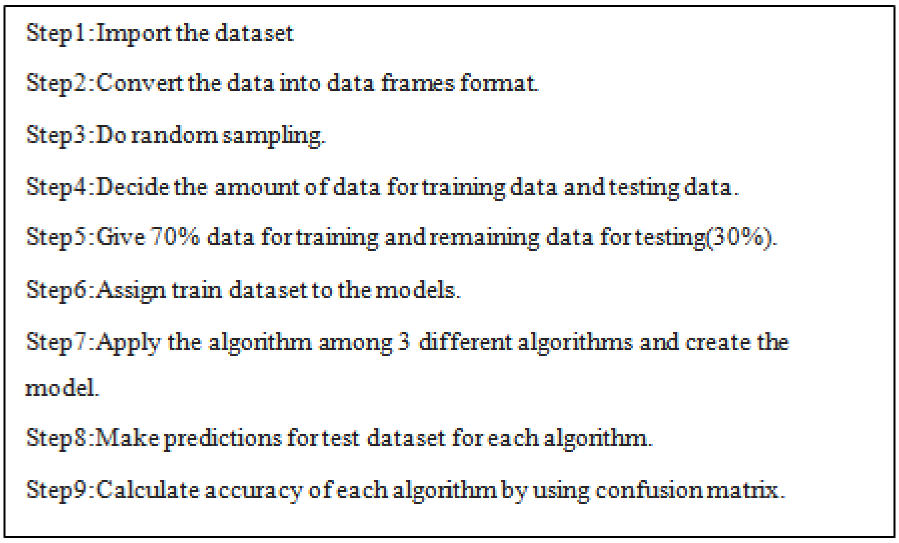
Where, P (A) – Priority of A P (B) – Priority of B P (A/B) – Posteriori priority of B

LogisticRegression: This algorithm similar to linear regression algorithm.But linear regression issued for predict / forecast values and Logistic regression is used for classificationtask.This algorithm easy for binary and multivariate classification task. Binomial is of 2 possible types (i.e. 0 or 1) only. Multinomial is of 3 or possible types and which are not ordered and Ordinal is in ordered in category (i.e. very poor, poor , good, very good).

Random Forest: First, start with the selection of random samples from a given dataset.Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree. Then voting will be performed for every predicted result. So finally, select the most voted prediction result as the final prediction result.

AdaBoost: AdaBoost is a machine learningalgorithm. Mainly developed for binary classification.For AdaBoost,Each instance in the training dataset is weighted. Initial weight is set To:

Weight (xi)= (1/n)Where, xi – i^{th} traininginstance n– Number of training instance



Step1:Import the dataset

Step2:Convert the data into data frames format.

Step3:Do random sampling.

Step4:Decide the amount of data for training data and testing data.

Step5:Give 70% data for training and remaining data for testing(30%).

Step6:Assign train dataset to the models.

Step7:Apply the algorithm among 3 different algorithms and create the model.

Step8:Make predictions for test dataset for each algorithm.

Step9:Calculate accuracy of each algorithm by using confusion matrix.

Algorithm steps for finding the Best algorithm

Test data:After training is done on the datasetthen testing process take place.

Outcome for test data: We will get the respective results for each algorithm and performance is displayed ingraphs.

Accuracy results: Finallyresults of each algorithm are shown with accuracy and the best algorithm is identified. **Evaluation:** There are a variety of measures for various algorithms and these measures have been developed toevaluate very different things .So it should be criteria for evaluation of various proposed method. False Positive(FP),False Negative(FN),True Positive(TP),True Negative(TN) and the relation between them are quantities which usually adopted by credit card fraud detection researchers to compare the accuracy of different approaches. The definitions of mentioned parameters are presented below:

- **True Positive(TP):**The true positive rate represents the portion of the fraudulent transactions correctlybeing classified as fraudulent transactions.

True positive=Tp/TP+FN

- **TrueNegative(TN):**The true negative rate represents the portion of the normal transactions correctly beingclassified as normal transactions.

True negative=TN/TN+FP

- **False Positive (FP):**The false positive rate indicates the portion of the non-fraudulent transactionswronglybeing classified as fraudulent transactions.

False positive=FP/FP+TN

- **False Negative (FN):**The false negative rate indicates the portion of the non-fraudulent transactionswrongly being classified as normal transactions.

False negative=FN/FN+TP

- **Confusion matrix:** The confusion matrix provides more insight into not only the performance of a predictive model, but also which classes are being predicted correctly, which incorrectly, and what type of errors are being made.The simplest confusion matrix is for a two-class classification problem, with negativeand positive classes. In this type of confusion matrix, each cell in the table has a specific and well- understood name

Predicted	Positive	Negative
Positive	TP	FN
Negative	FP	TN

- **Accuracy:**Accuracy is the percentage of correctly classified instances.It is one of the most widely usedclassification performance metrics.

The accuracy can be defined as:Accuracy= $\frac{TP+TN}{TP+TN+FP+FN}$

- **Precision and recall:**Precision is the number of classifiedPositive or fraudulent instances that actually are positive instances.

Precision = Tp/(Tp+Fp)

- Recall is a metric that quantifies the number of correct positive predictions made out of all positive predictions that could have been made.Unlike precision that only comments on the correct positive predictions out of all positive predictions, recall provides an indication of missed positive predictions.Recall is calculated as the number of true positives divided by the total number of true positives and false negatives.

Recall = Tp / (Tp + Fn)

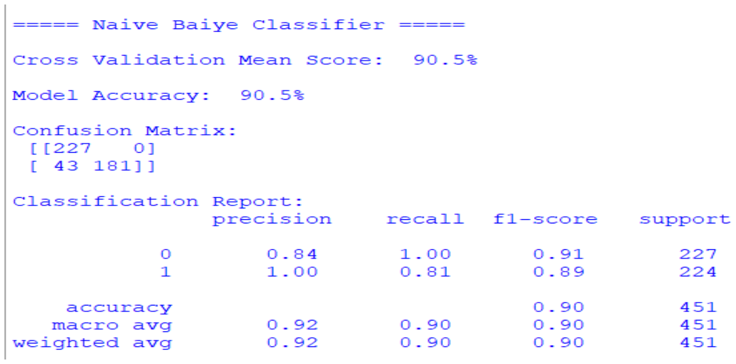
- **F1 score:** F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account.

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

- **Support:** The support is the number of samples of the true response that lie in that class.Support is the number of actual occurrences of the class in the specified dataset. Imbalanced support in the training data may indicate structural weaknesses in the reported scores of the classifier and could indicate the need for stratified sampling or rebalancing. Support doesn’t change between models but instead diagnoses the evaluation process.

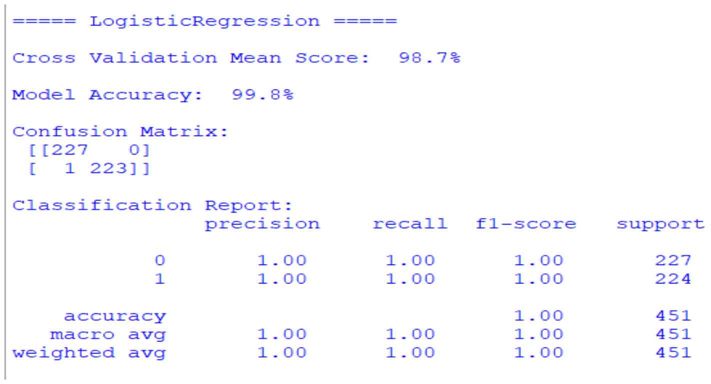
RESULTS AND DISCUSSIONS:

Model Evaluation Results



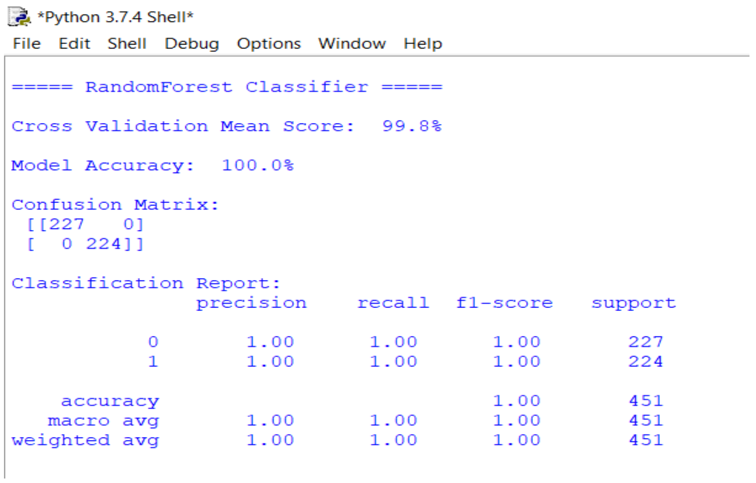
```
===== Naive Baiye Classifier =====

Cross Validation Mean Score:  90.5%

Model Accuracy:  90.5%

Confusion Matrix:
 [[227   0]
 [ 43 181]]

Classification Report:
               precision    recall  f1-score   support

           0       0.84      1.00      0.91       227
           1       1.00      0.81      0.89       224

    accuracy                           0.90       451
   macro avg       0.92      0.90      0.90       451
weighted avg       0.92      0.90      0.90       451
```

Fig -metrics for naïve bayes classifier model



```
===== LogisticRegression =====

Cross Validation Mean Score:  98.7%

Model Accuracy:  99.8%

Confusion Matrix:
 [[227   0]
 [  1 223]]

Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00       227
           1       1.00      1.00      1.00       224

    accuracy                           1.00       451
   macro avg       1.00      1.00      1.00       451
weighted avg       1.00      1.00      1.00       451
```

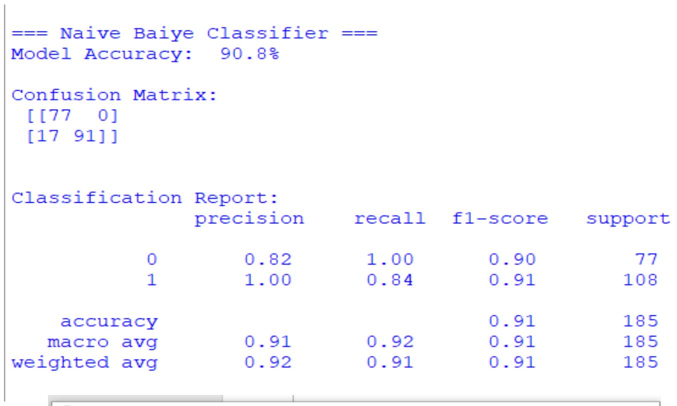
Fig: metrics for logistic regression classifier model



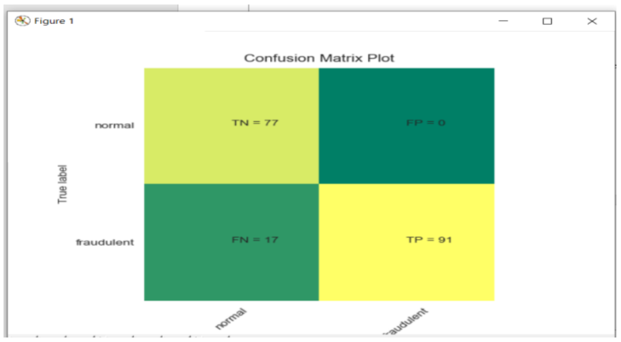
```
===== RandomForest Classifier =====

Cross Validation Mean Score:  99.8%

Model Accuracy:  100.0%

Confusion Matrix:
 [[227   0]
 [  0 224]]

Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00       227
           1       1.00      1.00      1.00       224

    accuracy                           1.00       451
   macro avg       1.00      1.00      1.00       451
weighted avg       1.00      1.00      1.00       451
```

Fig:metrics for random forest classifier model

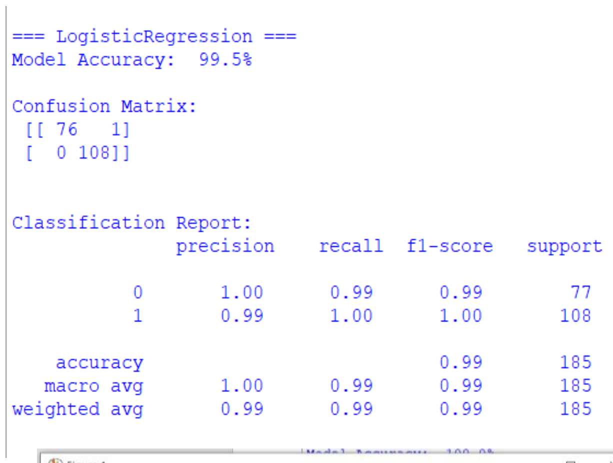
Model Test Results And Graphs



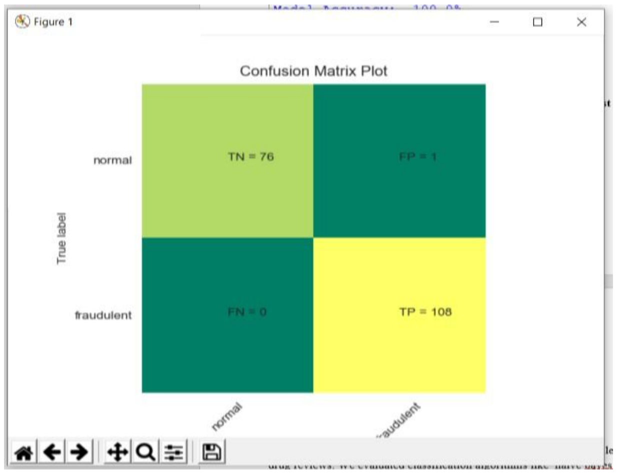
```
=== Naive Baiye Classifier ===
Model Accuracy:  90.8%

Confusion Matrix:
 [[77  0]
 [17 91]]


Classification Report:
              precision    recall  f1-score   support

           0       0.82      1.00      0.90        77
           1       1.00      0.84      0.91       108

    accuracy                           0.91       185
   macro avg       0.91      0.92      0.91       185
weighted avg       0.92      0.91      0.91       185
```



Test Result and confusion matrix plot for naive bayes classifier model

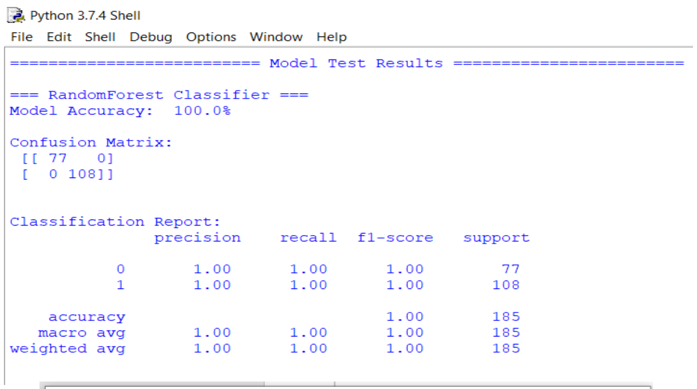


```
=== LogisticRegression ===
Model Accuracy:  99.5%

Confusion Matrix:
 [[ 76   1]
 [  0 108]]


Classification Report:
               precision    recall  f1-score   support

           0       1.00      0.99      0.99        77
           1       0.99      1.00      1.00       108

    accuracy                           0.99       185
   macro avg       1.00      0.99      0.99       185
weighted avg       0.99      0.99      0.99       185
```

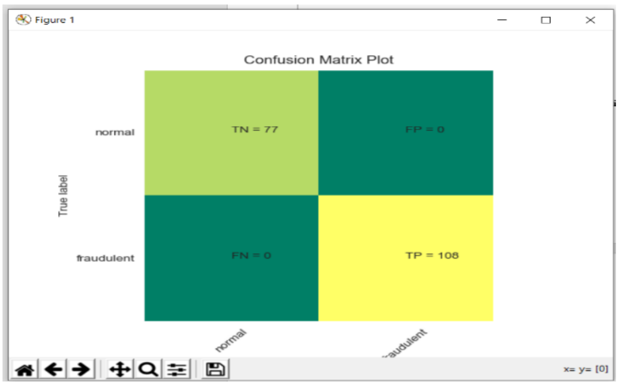


:

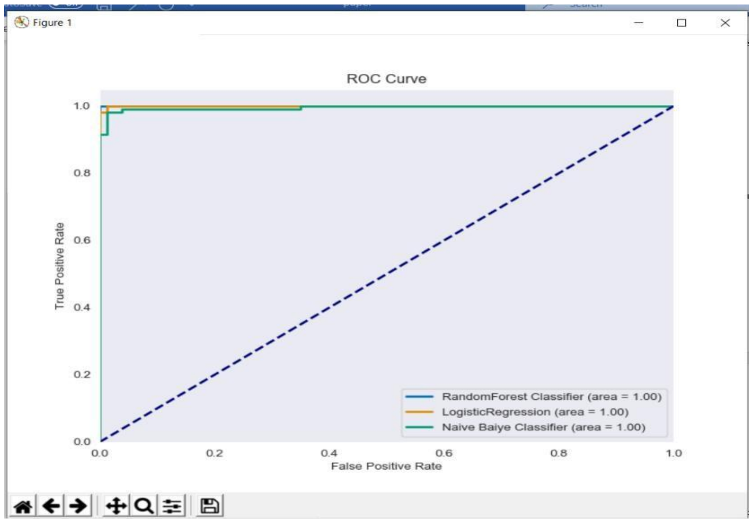
<u>Test Result and confusion matrix plot for logistic regression model</u>



```
======================== Model Test Results =======================

=== RandomForest Classifier ===
Model Accuracy:  100.0%

Confusion Matrix:
 [[ 77   0]
 [  0 108]]


Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00        77
           1       1.00      1.00      1.00       108

    accuracy                           1.00       185
   macro avg       1.00      1.00      1.00       185
weighted avg       1.00      1.00      1.00       185
```



Test Result and confusion matrix plot for random forest model



ROC Curve for all the three models

In this paper, we studied applications of machine learning like Naïve Bayes, Logistic regression, Random forest with boosting and shows that it proves accurate in deducting fraudulent transaction and minimizing the number of false alerts. Supervised learning algorithms are novel one in this literature in terms of application domain. If these algorithmsare applied into bank credit card fraud detection system, the probability of fraud transactions can be predicted soon after credit card transactions. And a series of anti-fraud strategiescan be adopted to prevent banks from great losses and reduce risks. The objective of the study was taken differently than the typical classification problems in that we had a variable misclassification cost. Percision,recall.f1-score,support and accuracy are used to evaluate the performance for the proposed system. By comparing all the three methods, we found that random forest classifier with boosting technique is better than the logistic regression and naïve bayes methods.