Credit Card Fraud Detection by sneha kumari

PASSWORD

ABSTRACT

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, cannot be overstated. This project intends to illustrate the modelling of a data set using machine learning with Credit Card Fraud Detection. 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 data sets as well as the deployment of multiple anomaly detection algorithms such as Local Outlier Factor and Isolation Forest algorithm on the PCA transformed Credit Card Transaction data

INTRODUCTION

'Fraud' in credit card transactions is unauthorized and unwanted usage of an account by someone other than the owner of that account. Necessary prevention measures can be taken to stop this abuse and the behaviour of such fraudulent practices can be studied to minimize it and protect against similar occurrences in the future.In other words, Credit Card Fraud can be defined as a case where a person uses someone else’s credit card for personal reasons while the owner and the card issuing authorities are unaware of the fact that the card is being used. Fraud detection involves monitoring the activities of populations of users in order to estimate, perceive or avoid objectionable behaviour, which consist of fraud, intrusion, and defaulting. This is a very relevant problem that demands the attention of communities such as machine learning and data science where the solution to this problem can be automated. This problem is particularly challenging from the perspective of learning, as it is characterized by various factors such as class imbalance. The number of valid transactions far outnumber fraudulent ones. Also, the transaction patterns often change their statistical properties over the course of time.

These are not the only challenges in the implementation of a real-world fraud detection system, however. In real world examples, the massive stream of payment requests is quickly scanned by automatic tools that determine which transactions to authorize. Machine learning algorithms are employed to analyse all the authorized transactions and report the suspicious ones. These reports are investigated by professionals who contact the cardholders to confirm if the transaction was genuine or fraudulent. The investigators provide a feedback to the automated system which is used to train and update the algorithm to eventually improve the fraud-detection performance over time.

Fraud detection methods are continuously developed to defend criminals in adapting to their fraudulent strategies. These frauds are classified as:

- Credit Card Frauds: Online and Offline
- Card Theft
- Account Bankruptcy
- Device Intrusion
- Application Fraud
- Counterfeit Card
- Telecommunication Fraud

METHODOLOGY

Data Cleaning and Preprocessing:

The datasets which were collected from UCI machine learning repository and Kaggle website contain unfiltered data which must be filtered before the final data set can be used to train the model. Also, data has some categorical variables which must be modified into numerical values for which we used Pandas library of Python. In data cleaning step, first we checked whether there are any missing or junk values in the dataset for which we used the isnull() function. Then for handling categorical variables we converted them into numerical variables.

Machine learning Algorithms:

Random Forest:

Random Forest is the most famous and it is considered as the best algorithm for machine learning. It is a supervised learning algorithm. To achieve more accurate and consistent prediction, random forest creates several decision trees and combines them together. The major benefit of using it is its ability to solve both regression and classification issues. When building each individual tree, it employs bagging and feature randomness in order to produce an uncorrelated tree forest whose collective forecast has much better accuracy than any individual tree's prediction. Bagging enhances accuracy of machine learning methods by grouping them together. In this algorithm, during the splitting of nodes it takes only random subset of nodes into an account. When splitting a node, it looks for the best feature from a random group of features rather than the most significant feature. This results into getting better accuracy. It efficiently deals with the huge datasets. It also solves the issue of overfitting in datasets. It works as follows: First, it'll select random samples from the provided dataset. Next, for every selected sample it'll create a decision tree and it'll receive a forecasted result from every created decision tree. Then for each result which was predicted, it'll perform voting and through voting it will select the best predicted result.

Logistic Regression :

Logistic regression is often used a lot of times in machine learning for predicting the likelihood of response attributes when a set of explanatory independent attributes are given. It is used when the target attribute is also known as a dependent variable having categorical values like yes/no or true/false, etc. It's widely used for solving classification problems. It falls under the category of supervised machine learning. It efficiently solves linear and 12 binary classification problems. It is one of the most commonly used and easy to implement algorithms. It's a statistical technique to predict classes which are binary. When the target variable has two possible classes in that case it predicts the likelihood of occurrence of the event. In our dataset the target variable is categorical as it has only two classes-yes/no.

Naive Bayes :

It is a probabilistic machine learning algorithm which is mainly used in classification problems. 11 | Page It’s based on Bayes theorem. It is simple and easy to build. It deals with huge datasets efficiently. It can solve complicated classification problems. The existence of a specific feature in a class is assumed to be independent of the presence of any other feature according to naïve bayes theorem. It’s formula is as follows :

P(S|T) = P(T|S) * P(S) / P(T)

Here, T is the event to be predicted, S is the class value for an event. This equation. will find out the class in which the expected feature for classification.

Support Vector Machine (SVM) :

It is a powerful machine learning algorithm that falls under the category of supervised learning. Many people use SVM to solve both regression and classification problems. The primary role of SVM algorithm is that it separates two classes by creating a line of hyperplanes. Data points which are closest to the hyperplane or points of the data set that, if deleted, would change the position of dividing the hyperplane are known as support vectors. As a result, they might be regarded as essential components of the data set. The margin is the distance between hyperplane and nearest data point from either collection. The goal is to select the hyperplane with the maximum possible margin between it and any point in the training set increasing the likelihood of a new data being properly classified. SVM's main objective is to find a hyperplane in N-dimensional space which will classify all the data points. The dimension of a hyperplane is actually dependent on the quantity of input features. If input has two features in that case the hyperplane will be a line and two dimensional plane.

K Nearest Neighbor (KNN) :

KNN is a supervised machine learning algorithm. It assumes similar objects are nearer to one another. When the parameters are continuous in that case knn is preferred. In this algorithm it classifies objects by predicting their nearest neighbor. It's simple and easy to implement and also has high speed because of which it is preferred over the other algorithms when it comes to solving classification problems.

Algorithm takes following steps :-

Step 1: Select the value for K.

Step 2 : Find the Euclidean distance of K no. of neighbors.

Step 3 : Based on calculated distance, select the K nearest neighbors in the training 13 data which are nearest to unknown data points.

Step 4 : Calculate no. of data points in each category among these K neighbors.

Step 5 : Assign new data points to the category which has the maximum no. of neighbors.

Step 6 : Stop.

Implementation Steps:

As we already discussed in the methodology section about some of the implementation details. So, the language used in this project is Python programming. We're running python code in anaconda navigator's Jupyter notebook. Jupyter notebook is much faster than Python IDE tools like PyCharm or Visual studio for implementing ML algorithms. The advantage of Jupyter notebook is that while writing code, it's really helpful for Data visualization and plotting some graphs like histogram and heatmap of correlated matrices.

Let's revise implementation steps :

a) Dataset collection.

b) Importing Libraries : Numpy,

Pandas, Scikit-learn, Matplotlib and Seaborn libraries were used.

c) Exploratory data analysis : For getting more insights about data.

d) Data cleaning and preprocessing : Checked for null and junk values using isnull() and isna().sum() functions of python.In Preprocessing phase, we did feature engineering on our dataset. As we converted categorical variables into numerical variables using function of Pandas library. Both our datasets contains some categorical variables.

e) Feature Scaling : In this step, we normalize our data by applying Standardization by using StandardScalar() and fit_transform() functions of scikit-learn library.

f) Model selection : We first separated X's from y's. X's are features or input variables of our datasets and y's are dependent or target variables which are crucial for predicting disease. Then using by the importing model_selection function of the sklearn library, we splitted our X's and y's into train and test split using train_test_split() function 14 of sklearn. We splitted 80% of our data for training and 20% for testing.

g) Applied ML models and created a confusion matrix of all models.

h) Deployment of the model which gave the best accuracy.

CONCLUSIONS

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

- Accuracy of the models
- Logistic Regression :- 92 %
- Decision Tree :- 88 %
- KNN :- 72 %
- SVM :- 56 %
- Naïve Bayes Classifier :- 88 %
- Random Forest :- 93 %
- From comparing the 6 models, we can conclude that Model 6: Random Forest yields the highest accuracy, With an accuracy of 93%.

Credit Card is a great tool to pay money easily, but as with all the other monetary payment tools, reliability is a issue here too as it is subjected to breach and other frauds. To encounter this problem, a solution is needed to identify the patterns in the transactions and identify the ones which are fraud, so that finding such transactions beforehand in future will be very easy.

Machine Learning is a great tool to do this work since Machine Learning helps us in finding patterns in the data. Machine Learning can help producing great results if provided enough amount of data. Also, with further advances in the technology, Machine Learning too will advance with time, it will be easy for a person to predict if a transaction is fraud or not much more accurately with the advances.