**CREDIT CARD FRAUD DETECTION**

**PHASE 2 PROJECT: INNOVATION**

**TABLE OF CONTENTS:**

\*INTRODUCTION

\*PROBLEM DEFINITION

\*INNOVATION

\*DESIGN THINKING

\*MODEL BUILDING

\*CONCLISION

**INTRODUCTION:**

The problem is to develop a machine learning-based system for real-time credit card fraud detection. The goal is to create a solution that can accurately identify fraudulent transactions while minimizing false positives. This project involves data preprocessing, feature engineering, model selection, training, and evaluation to create a robust fraud detection system.

**PROBLEM DEFINITION:**

The mission of our project is to prevent real-time credit card fraud and take measures for it to stop.

**INNOVATION**

We have used a technique called anomaly detection. There are various algorithms to implement anomaly detection. The one that we will be using is One class-SVM. One-class SVM is an unsupervised algorithm that learns a decision function for novelty detection: classifying new data as similar or different to the training set.

Let us see the process step by step in the below

**DESIGN THINKING**

**1)Data Collection:**

Dataset required for implementing the project is given below :

https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

**2) Data Preprocessing:**

\*IMPORTING LIBRARIES-

The first and foremost step is importing the libraries

import pandas as pd

import numpy as np

from sklearn import svm

import matplotlib.pyplot as plt

\*IMPORTING DATASET-

The next step is importing the dataset from the source

data = pd.read\_csv('demand\_data.csv')

**3)Data Cleaning :**

There were no duplicate and redundant data found.Moreover we didn’t have any outliers and nan values . so no data in the data set was replaced. we move forward to the next step .

In case of outliers present:

unique, counts = np.unique(fraud\_pred, return\_counts=True)

print (np.asarray((unique, counts)).T)

**MODEL BUILDING :**

**1)Model Selection :**

There are multiple algorithms available for forecasting the product prediction, The model we have selected is One class – SVM(anomaly detection). One-class SVM is an unsupervised algorithm that learns a decision function for novelty detection: classifying new data as similar or different to the training set.

**2)Model Training:**

The model is trained using the below lines of code

train\_feature = nor\_obs.loc[0:200000, :]

train\_feature = train\_feature.drop('Category', 1)

Y\_1 = nor\_obs.loc[200000:, 'Category']

Y\_2 = ano\_obs['Category']

**3)Model Testing:**

X\_test\_1 = nor\_obs.loc[200000:, :].drop('Category',1)

X\_test\_2 = ano\_obs.drop('Category',1)

X\_test = X\_test\_1.append(X\_test\_2)

**4)Setting hyperparameters:**

oneclass = svm.OneClassSVM(kernel='linear', gamma=0.001, nu=0.95)

Y\_1 = nor\_obs.loc[200000:, 'Category']

Y\_2 = ano\_obs['Category']

Y\_test= Y\_1.append(Y\_2)

**5) Training and testing using the parameters:**

oneclass.fit(train\_feature)

fraud\_pred = oneclass.predict(X\_test)

Y\_test= Y\_test.to\_frame()

Y\_test=Y\_test.reset\_index()

fraud\_pred = pd.DataFrame(fraud\_pred)

fraud\_pred= fraud\_pred.rename(columns={0: 'prediction'})

**6) Model evaluation:**

TP = FN = FP = TN = 0

for j in range(len(Y\_test)):

if Y\_test['Category'][j]== 0 and fraud\_pred['prediction'][j] == 1:

TP = TP+1

elif Y\_test['Category'][j]== 0 and fraud\_pred['prediction'][j] == -1:

FN = FN+1

elif Y\_test['Category'][j]== 1 and fraud\_pred['prediction'][j] == 1:

FP = FP+1

else:

TN = TN +1

print (TP, FN, FP, TN)

accuracy = (TP+TN)/(TP+FN+FP+TN)

print (accuracy)

sensitivity = TP/(TP+FN)

print (sensitivity)

specificity = TN/(TN+FP)

print (specificity)

**CONCLUSION:**

Thus, I have implemented my design into an innovation by using anomaly detection algorithms like one class SVM and predicted a model to detect credit card fraud.

**REFERENCE:**

\*https://www.kaggle.com/code/amarnayak/once-class-svm-to-detect-anomaly