Table of Contents

[1 Introduction 2](#_Toc141279195)

[2 Dataset 3](#_Toc141279196)

[2.1 Description of Dataset: 3](#_Toc141279197)

[2.2 Null values: 3](#_Toc141279198)

[2.3 Total Columns: 3](#_Toc141279199)

[2.4 Plotting continuous data: 4](#_Toc141279200)

[2.5 Data Distribution: 5](#_Toc141279201)

# Introduction

This file contains the results and process we operated a skewed Dataset. By skewed dataset we mean the data corresponding different classes were very skewed. So, the dataset selected for this purpose was ‘credit\_card.csv’. Where based on features we have to predict whether a transaction was Fraud or not.

The columns of the dataset are:

A screenshot of a computer

Description automatically generated

A screenshot of a calculator

Description automatically generated

Where, ‘Time’ represents the time it took for the transaction. ‘Amount’ represents the amount of transaction happened and ‘Class’ represents whether the transaction was ‘Fraud:1’ or not ‘Not Fraud: 1’.

# Dataset

## Description of Dataset:

A table with numbers and symbols

Description automatically generated

A table with numbers and symbols

Description automatically generated

## Null values:

A close-up of a computer code

Description automatically generated

We do not need to do anything as there are no missing values.

## Total Columns:

A group of numbers and symbols

Description automatically generated

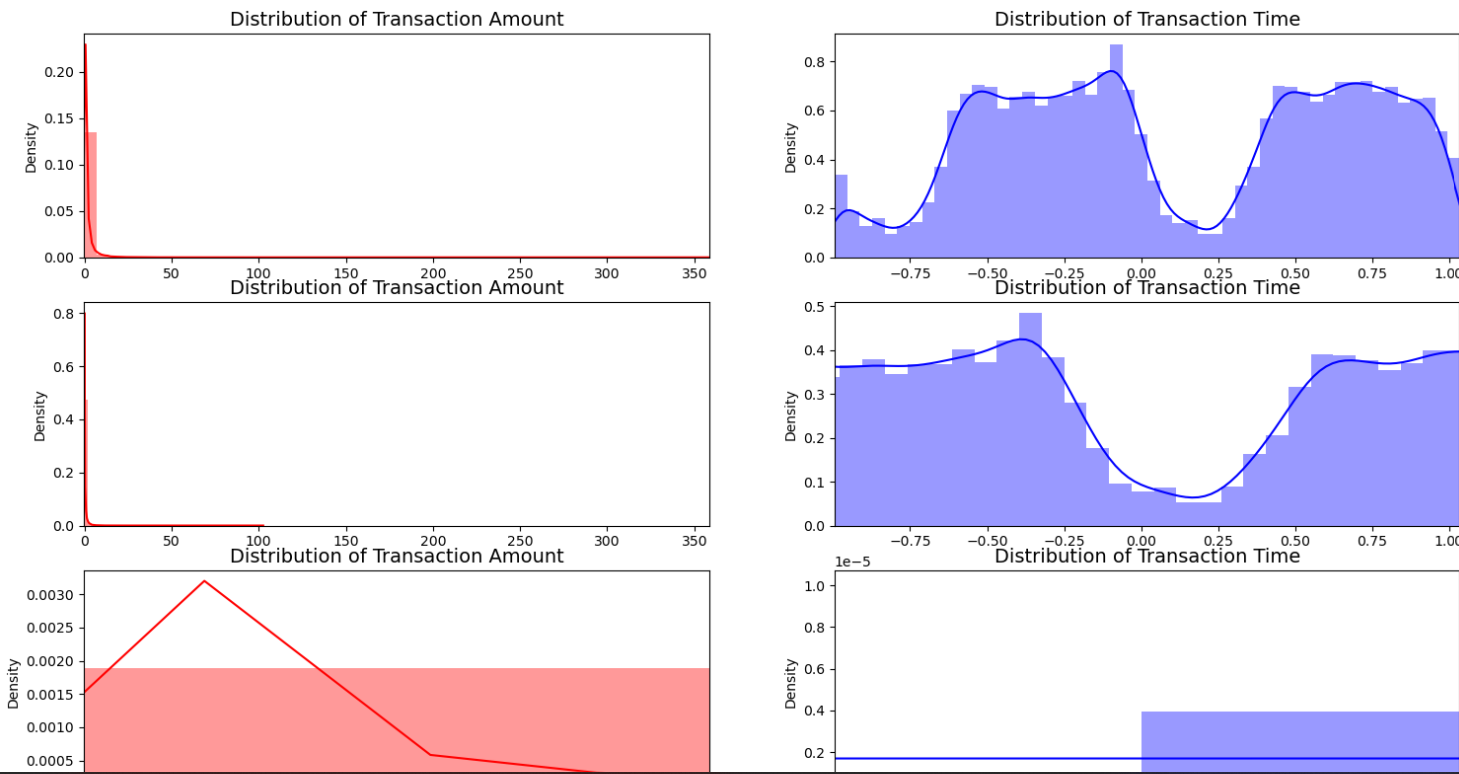
## A graph of a graph Description automatically generatedPlotting continuous data:

As could be seen that the distribution of continuous features is not normal hence, we need to transform those to more continuous form. We would use transformations for that.

The transformations we used for solving above problem are:

1. Robust Scaler
2. Standard Scaler
3. Logarithmic transformation

The task is to make the distribution of continuous data more normally distributed data.



It is wise to use Robust scaler as Standard scaler is not showing good performance for transaction amount and logarithmic transformation is not giving good performance for transaction time.

## Data Distribution:

The dataset looks like:

A screenshot of a computer

Description automatically generated

Total number of data associated with different classes:

A computer screen shot of a number

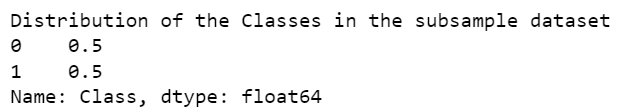
Description automatically generated

As could be seen here that the data is highly skewed as the percentage of distribution between classes is very different. What problem it brings is if our proposed algorithm predicts class 0 (No Frauds) every time then also it is giving an accuracy of 99.83 % here. So, we need to sort that as well.

A black rectangular object with white text

Description automatically generated

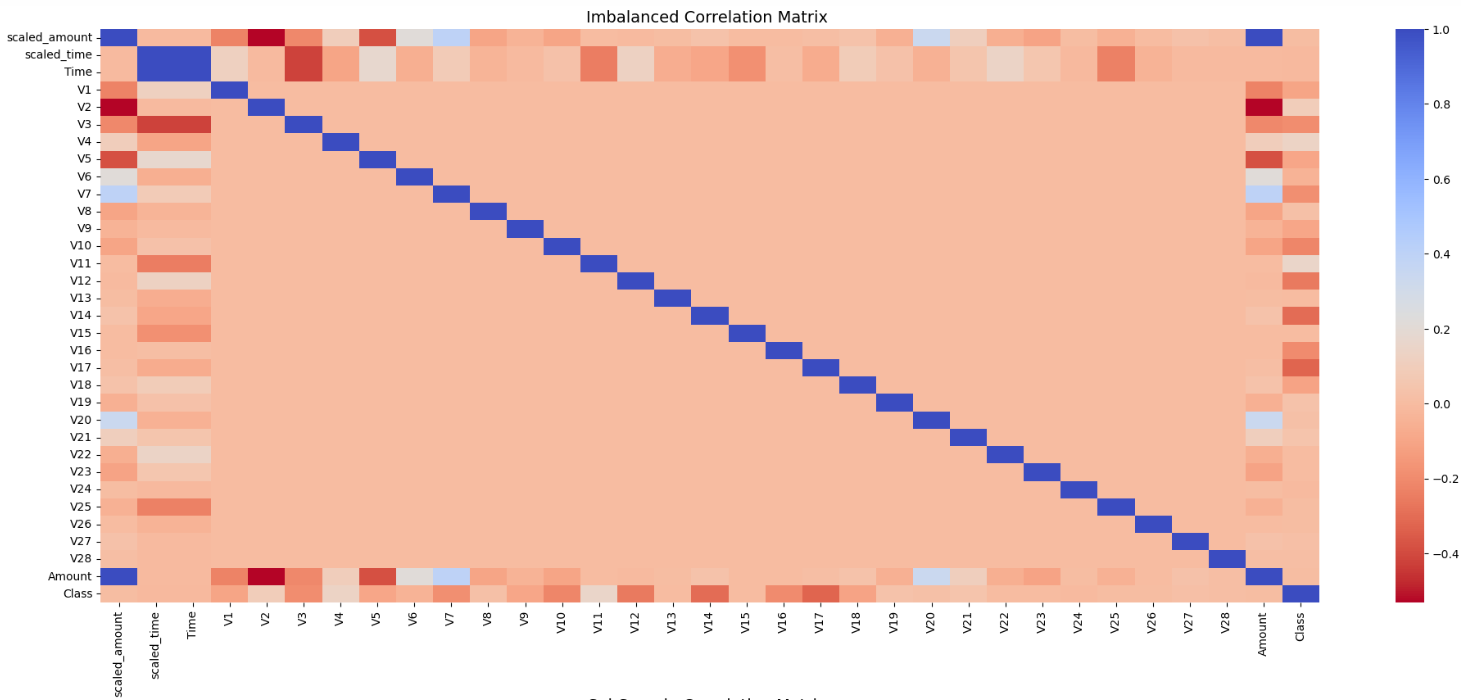
We took 492 rows of dataset ‘No Fraud’ so as the number of rows become same for both classes.



A graph of a number of classes

Description automatically generated

Correlation heat map with full dataset:

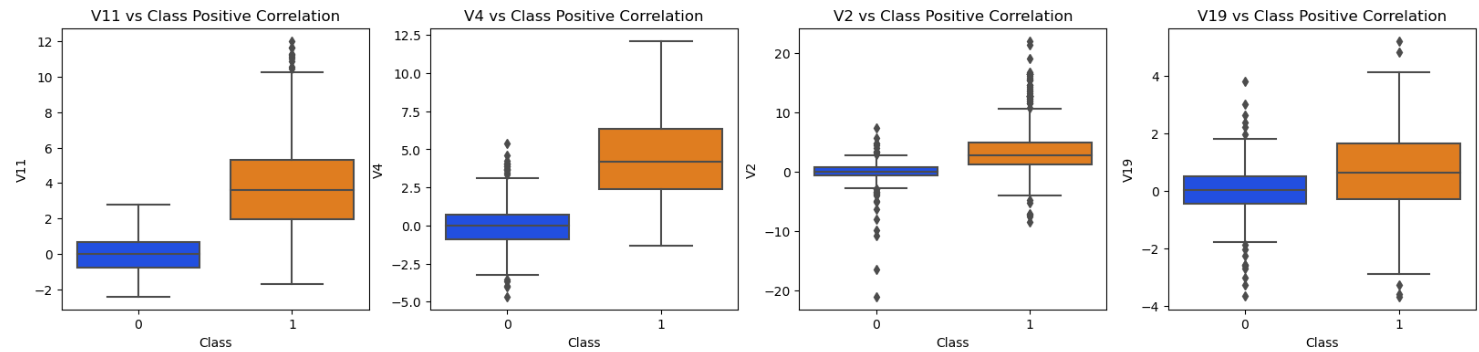
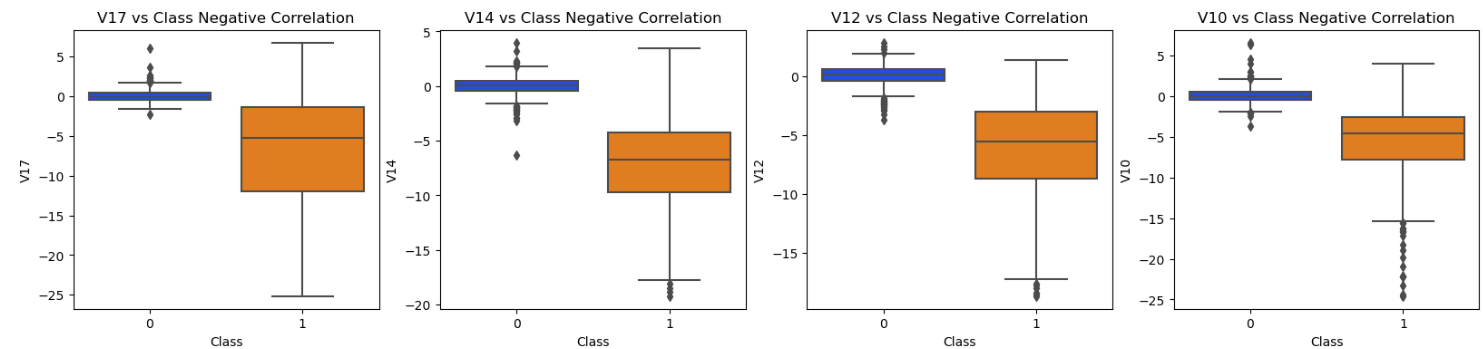


Correlation heat map with equally distributed dataset:

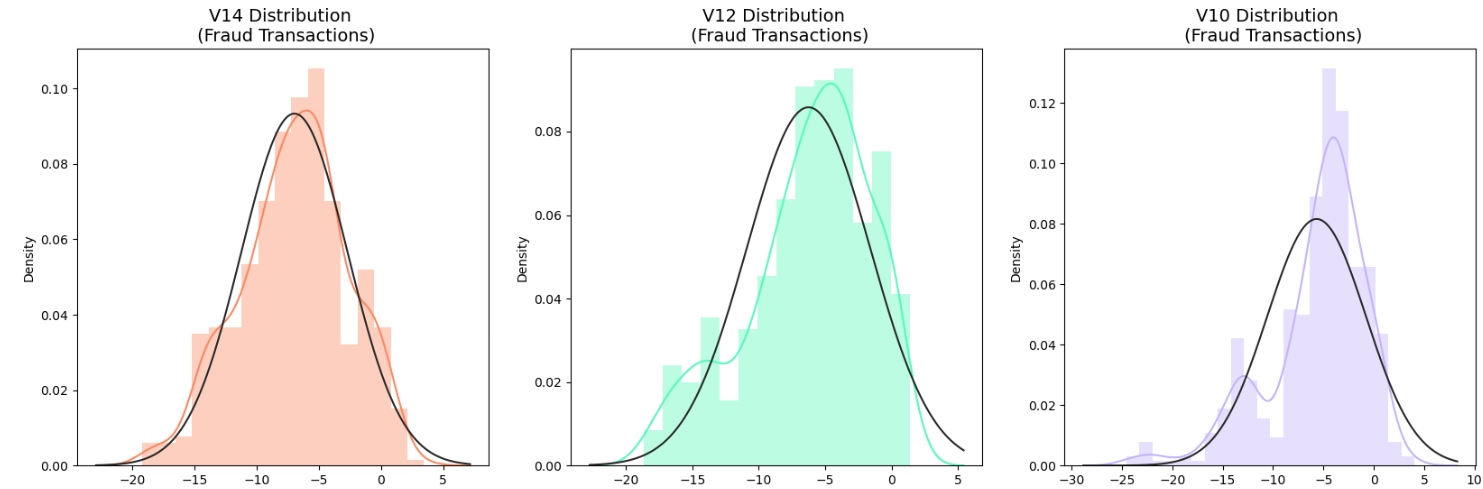
A graph of a graph

Description automatically generated

Box plot with respect to positive and negative correlated features:



The Distribution of features for fraud transactions:



Determining outliers and dropping them for class ‘Fraud’:

A white page with black text

Description automatically generatedA graph with a green and pink line

Description automatically generated