Mini Project 1

Abstract

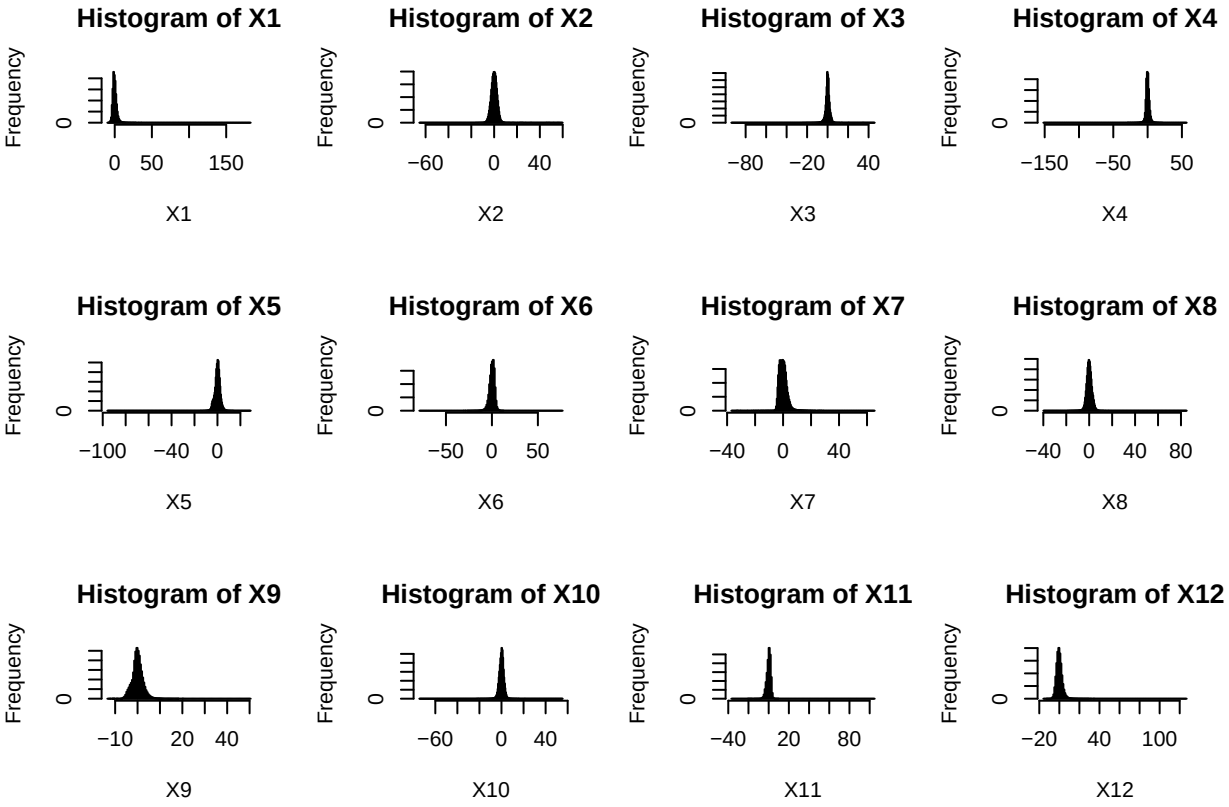
Introduction

In 2024, $12.5 billion dollars were lost to fraud in the United States alone, according to the Federal Trade Commission (FTC). This staggering figure underscores the critical need for effective fraud detection mechanisms. In this analysis, we will use a dataset containing 250,000 financial transactions over a two-day period to try and identify 250 of the <500 fraudulent transactions. To identify these transactions, we will use outlier detection methods under a multivariate normal distribution assumption.

Data Description

The dataset consists of 250,000 observations and 12 variables. Due to the sensitive nature of the data, the variables are anonymized and labeled as X1 to X12. Each observation represents a financial transaction, with the variables capturing various attributes of these transactions. This dataset is *not labeled*, meaning we do not have prior knowledge of which transactions are fraudulent. This lack of labels necessitates the use of unsupervised learning techniques for fraud detection.

Assessment of Normality



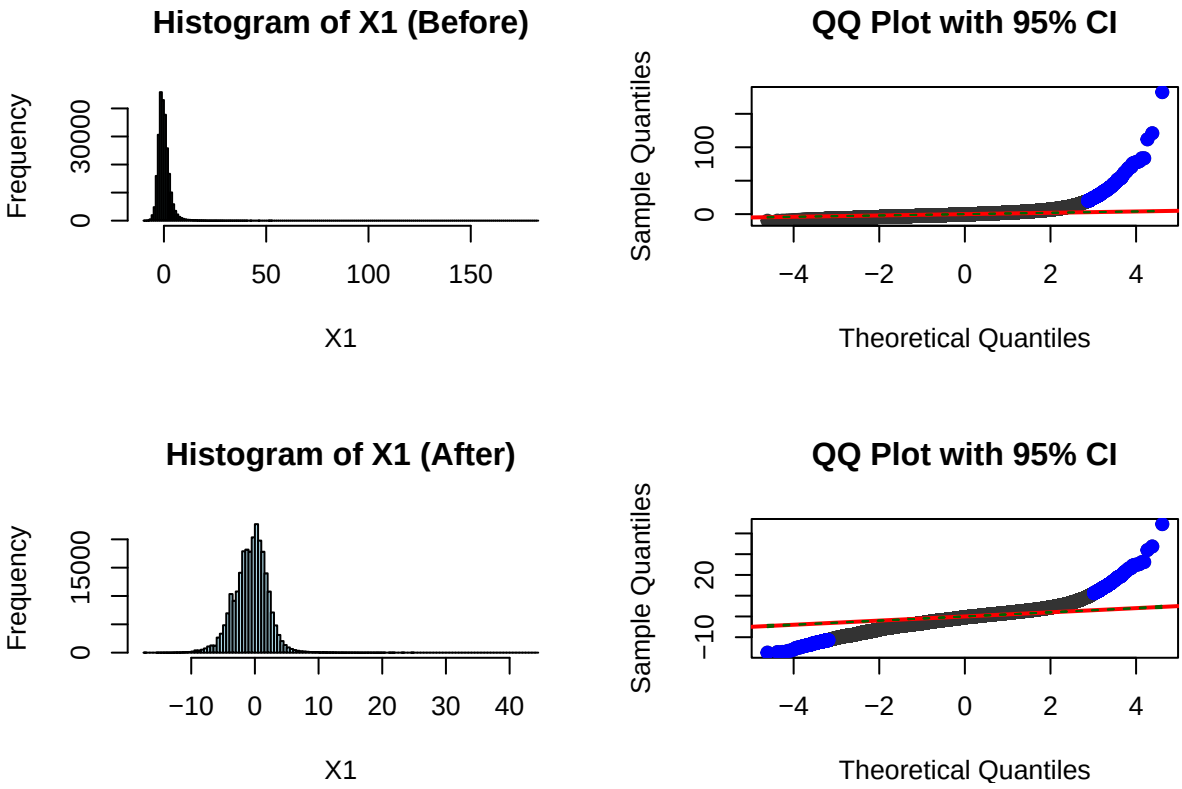
An initial assessment of the data reveals that the variables do not follow a normal distribution, as evidenced by the histograms above. In general, the distributions tend to be very heavy tailed and skewed. To improve the skewness of each variable individually, we will apply a Yeo-Johnson transformation to each variable.

The Yeo-Johnson transformation is a power transformation that can handle both positive and negative values, making it more suitable for our dataset than a Box-Cox transformation. The Yeo-Johnson transformation is defined as follows:

$$Y(\lambda) = \begin{cases} \frac{(y+1)^{\lambda}-1}{\lambda} & \text{if } y \geq 0, \lambda \neq 0 \\ \log(y+1) & \text{if } y \geq 0, \lambda = 0 \\ \frac{-(|y|+1)^{2-\lambda}-1}{2-\lambda} & \text{if } y < 0, \lambda \neq 2 \\ -\log(|y|+1) & \text{if } y < 0, \lambda = 2 \end{cases}$$

To choose a lambda for an optimal transformation of each variable, the log likelihood under a normal distribution is maximized with respect to lambda. To improve the normality of the distributions, we will apply the Yeo-Johnson transformation to each variable and then reassess the normality using histograms and QQ plots with confidence intervals.

```
## Loading required package: carData
```



The figure above shows a histogram and qq plot of X1 before and after the Yeo-Johnson transformation. Initially, X1 exhibited right skewness and very heavy tails. After applying the Yeo-Johnson transformation, the distribution of X1 appears more symmetric and less heavy-tailed. The qq plot indicates that while the transformation has improved the fit to a normal distribution, there are still deviations from normality, particularly in the tails. While we would consider this transformation a step toward normality, it is still not perfectly normal.

Each of the other variables exhibited similar improvements in normality after the Yeo-Johnson transformation, but none achieved perfect normality. The heavy-tailed nature of the distributions persists to some extent across all variables. Their lambda values are recorded in the table below:

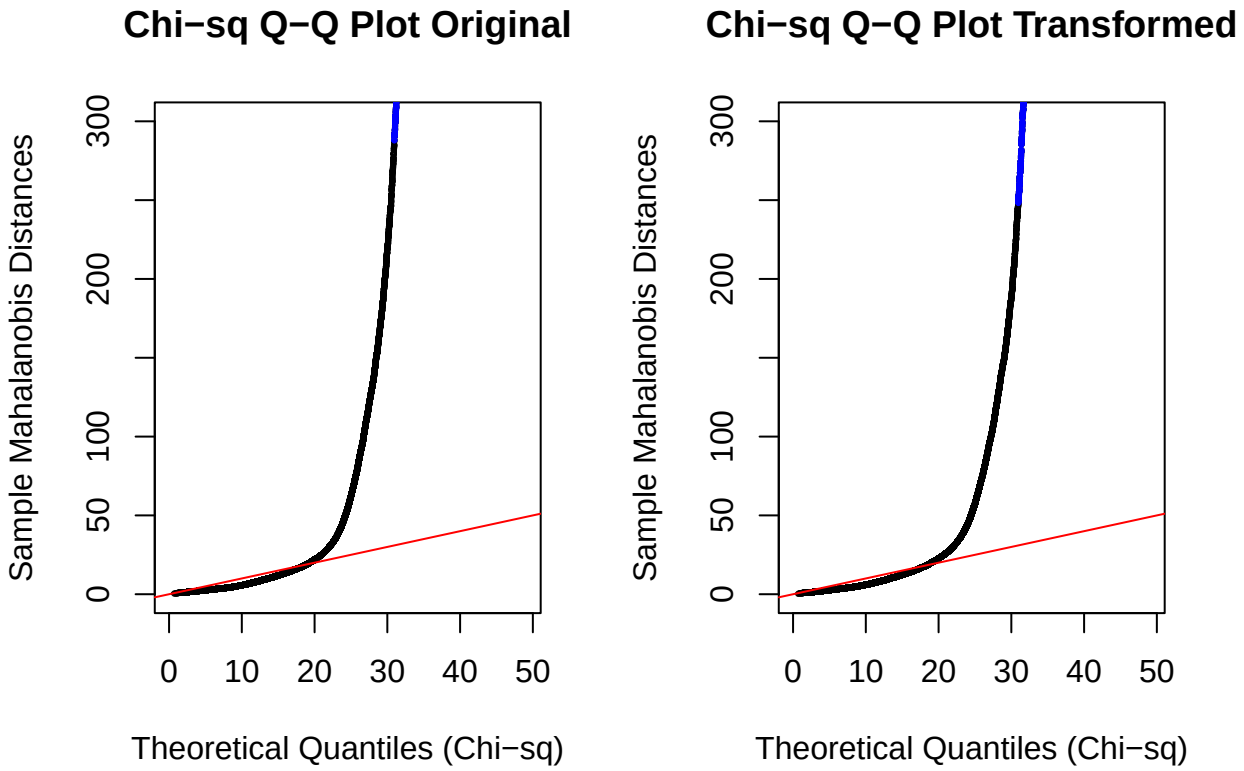
```
lambda_df <- data.frame(Variable = paste0("X", 1:ncol(df)), Lambda = lambda_vect)
knitr::kable(lambda_df, caption = "Yeo-Johnson Lambda Values for Each Variable")
```

Table 1: Yeo-Johnson Lambda Values for Each Variable

Variable	Lambda
X1	0.6527787
X2	0.8719703
X3	1.0548654
X4	1.0675768
X5	1.0589825
X6	1.2059133
X7	0.8577286
X8	1.0319135
X9	0.9048296
X10	1.0760808
X11	1.1444107
X12	0.7493411

To further assess the multivariate normality of the dataset, we will use Mahalanobis distance-based Q-Q plots. The Mahalanobis distance measures how far each observation is from the mean of the distribution, taking into account the correlations between variables. If the data were multivariate normal, the squared Mahalanobis distances would follow a Chi-square distribution with degrees of freedom equal to the number of variables. Mahalanobis distance is defined as follows:

$$D^2 = (x - \mu)^T S^{-1} (x - \mu)$$ where μ is the mean vector and S is the covariance matrix.



We will proceed with our analysis under the assumption of multivariate normality, acknowledging that the heavy-tailed nature of the distributions may lead to a higher rate of false positives in our outlier detection.

Outlier Detection (Potential Fraud Identification)

We identify potential fraudulent transactions in the data set by identifying outliers, assuming multivariate normality in the data. As was previously described, after transforming the data, there is still reason to believe that this data is not multivariate normal, but we will move forward with this assumption to be able to employ Mahalanobis distance as a method to detect the outliers.

In our case, assuming normality, we know that Mahalanobis Distance follows a χ^2_{12} distribution. We then say that any record with a Mahalanobis Distance greater than the 99% quantile of the χ^2_{12} is an outlier. This method gave us 11739 outliers, much more than we were expecting. Again, after the transformation, there is still skewness and heavy tails, so this result is not unexpected. Using the information we have, though, we believe that the records with the highest Mahalanobis Distance have the greatest chance of being fraudulent. The table below shows the top 10 records with the highest Mahalanobis Distance.

```
knitr::kable(top_10_table, caption = "Records with the highest Mahalanobis Distance")
```

Table 2: Records with the highest Mahalanobis Distance

Row	Mahalanobis.Distance
130115	16233.809
239483	4204.829
178487	3807.244
151505	3436.276
10957	2877.559
220686	2729.211
85388	2592.572
78754	2285.229
137598	2269.037
16197	2184.955

Discussion

Conclusion