Analysis on YOCOIN Data

Abhisek Banerjee, Nirmohi Dave

December 01, 2018

Introduction

The YOCOIN analysis project aims to analyze data of the Etherium coin YOCOIN and find out if we can deduce anything significant from the available data.

What is Etherium?

The Etherium project provides a decentralized platform that uses smart contracts which allows applications to run exactly as programmed without any possibility of downtime., fraud or third-party interference. Applications run on a customized blockchain, which is a very powerful shared global infrastructure that can move value around and represent the ownership of property. [Source : https://www.ethereum.org/]

What is ERC20?

ERC20 is a significant standard for tokens on Etherium. This defines a common list of rules that enables developers to accurately predict how new tokens will function within the larger Ethereum system. [Source : https://www.investopedia.com/news/what-erc20-and-what-does-it-mean-ethereum/]

YOCOIN TOKEN

YOCOIN was founded & launched on December 7th, 2015. It is an open-ledger, publicly exchanged, peer-to-peer cryptocurrency that is designed for the general public worldwide and will be utilized to pay for goods and services by many different industries across the globe, including but not limited to the direct sales industry. It uses the Etherium network for transaction and storage. [Sources: https://yocoinweb.wordpress.com/, https://steemit.com/yocoin/@tonypeacock/things-you-should-know-about-yocoin]

We have analyzed transaction data of YOCOIN over a specific period of time. YOCOINs can be made of multiple tokens and 10^16 tokens make a single YOCOIN. Also, there were 310,000,000 YOCOINs available at the time of our analysis.

Our goal

We tried to achieve the following things during this project.

1> We have taken the sellers and buyers information out of the dataset and tried to plot their frequencies to find out what distribution they follow.

2> We have taken the number of transactions for different dates between 07-21-2016 and 02-05-2018, and the corresponding highest token prices for each day and then tried to find out any correlation between highest stock price and the number of transactions in a particular day. To do this we split our dataset into multiple layers(bins) and computed correlation value for each layer.

3> We have found a few more features such as unique buyers, unique sellers, average transaction amount, daily price change and using them tried to construct a linear regression model with the coin price.

4> We have tried to construct a random forest to create a prediction model for price prediction.

Preprocessing

Before starting the analysis, we have done some preprocessing on the data and removed few outliers.

1> We removed all the transactions which were dealing with coins more than the total available coins. These were spurious transactions and we do not need to consider these for our analysis.

2> Then we removed very big transactions and very small transactions. These were extreme outliers and removing these values yielded better results.

Packages Used

We used the following packages in our code.

1>‘fitdistrplus’: This has been used to plot sellers and buyers data in different distributions.

2>‘ggplot2’: This has been used to plot data along the different axes with different attributes.

3> ‘reshape’: This has been used to join tables.

4> ‘randomForest’: This has been used for price prediction.

Analysis

```
Number of rows in unprocessed data: 746397
```

Summary of the unprocessed data:

```
    Sellers            Buyers          TimeStamps
 Min.   : 309659   Min.   :  14514   Min.   :1.469e+09
 1st Qu.:9911594   1st Qu.:9913576   1st Qu.:1.478e+09
 Median :9911594   Median :9915742   Median :1.484e+09
 Mean   :9854046   Mean   :9816628   Mean   :1.484e+09
 3rd Qu.:9912282   3rd Qu.:9918141   3rd Qu.:1.489e+09
 Max.   :9927144   Max.   :9927159   Max.   :1.518e+09
  TokenAmounts
 Min.   :1.000e+00
 1st Qu.:5.250e+16
 Median :2.210e+17
 Mean   :1.396e+73
 3rd Qu.:8.730e+17
 Max.   :1.158e+77
```

```
Number of outlier rows: 90
```

Summary of the outliers:

```
    Sellers            Buyers          TimeStamps
 Min.   :9911653   Min.   :9911653   Min.   :1.472e+09
 1st Qu.:9917007   1st Qu.:9911653   1st Qu.:1.472e+09
 Median :9922951   Median :9911653   Median :1.472e+09
 Mean   :9921471   Mean   :9911697   Mean   :1.472e+09
 3rd Qu.:9922951   3rd Qu.:9911653   3rd Qu.:1.472e+09
 Max.   :9926770   Max.   :9915616   Max.   :1.472e+09
  TokenAmounts
 Min.   :1.158e+77
 1st Qu.:1.158e+77
 Median :1.158e+77
 Mean   :1.158e+77
 3rd Qu.:1.158e+77
 Max.   :1.158e+77
```

Below is the table of outlying buyers and the frequencies of transactions.

```
  outlying_buyers Freq
1         9911653   89
2         9915616    1
```

Below is the table of outlying sellers and the frequencies of transactions.

```
  outlying_sellers Freq
1          9911653    1
2          9917007   34
3          9922951   34
4          9926770   21
```

First we removed impossible transactions.

```
    Sellers            Buyers          TimeStamps
 Min.   : 309659   Min.   :  14514   Min.   :1.469e+09
 1st Qu.:9911594   1st Qu.:9913582   1st Qu.:1.478e+09
 Median :9911594   Median :9915743   Median :1.484e+09
 Mean   :9854038   Mean   :9816616   Mean   :1.484e+09
 3rd Qu.:9912282   3rd Qu.:9918141   3rd Qu.:1.489e+09
 Max.   :9927144   Max.   :9927159   Max.   :1.518e+09
  TokenAmounts
 Min.   :1.000e+00
 1st Qu.:5.250e+16
 Median :2.210e+17
 Mean   :7.134e+19
 3rd Qu.:8.720e+17
 Max.   :1.000e+24
```

We did one more round of pre-processing and removed 1 percentile of data from both the sides.

```
Number of rows in processed data: 716323
```

Summary of the processed data:

```
    Sellers            Buyers          TimeStamps
 Min.   : 309659   Min.   :  14514   Min.   :1.469e+09
 1st Qu.:9911594   1st Qu.:9913670   1st Qu.:1.478e+09
 Median :9911594   Median :9915788   Median :1.483e+09
 Mean   :9852930   Mean   :9815750   Mean   :1.483e+09
 3rd Qu.:9912282   3rd Qu.:9918216   3rd Qu.:1.489e+09
 Max.   :9927144   Max.   :9927159   Max.   :1.518e+09
  TokenAmounts
 Min.   :1.100e+15
 1st Qu.:6.000e+16
 Median :2.295e+17
 Mean   :9.189e+18
 3rd Qu.:8.630e+17
 Max.   :2.000e+21
```

Summary of seller frequencies:

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
     1.0      1.0      2.0     89.2      4.0 497900.0
```

Summary of buyer frequencies:

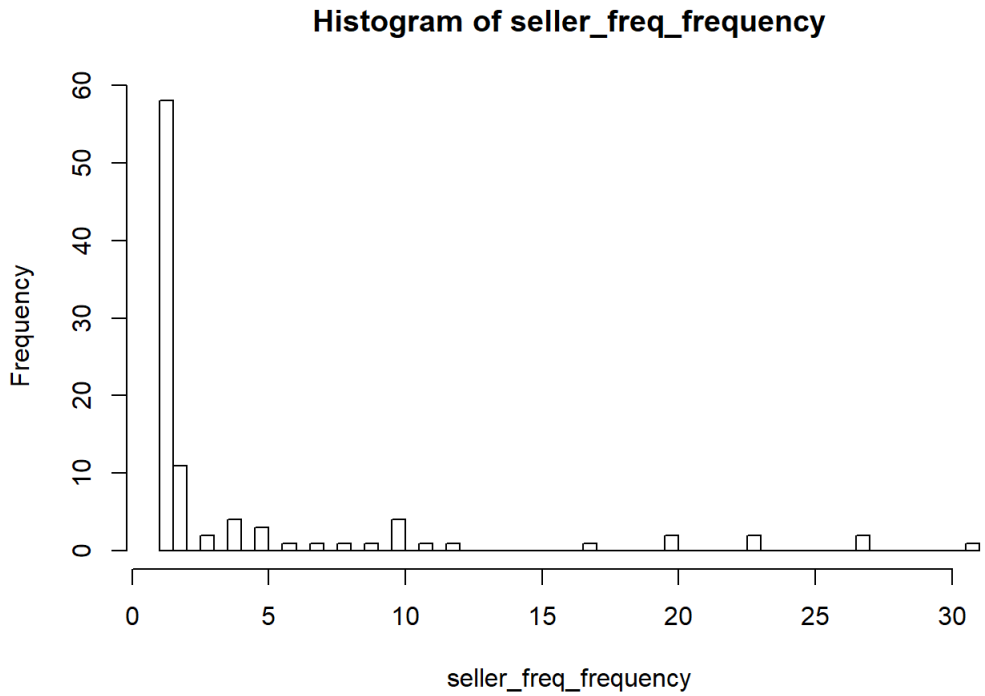
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    2.00    8.00   46.18   25.00 7224.00
```

Summary of frequencies of sellers frequencies:

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    1.00    1.00   73.02    9.75 3586.00
```

We removed some outliers by only keeping values < (.01*max value) as the median and max value varied greatly.

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.000   1.000   4.146   4.000  31.000
```

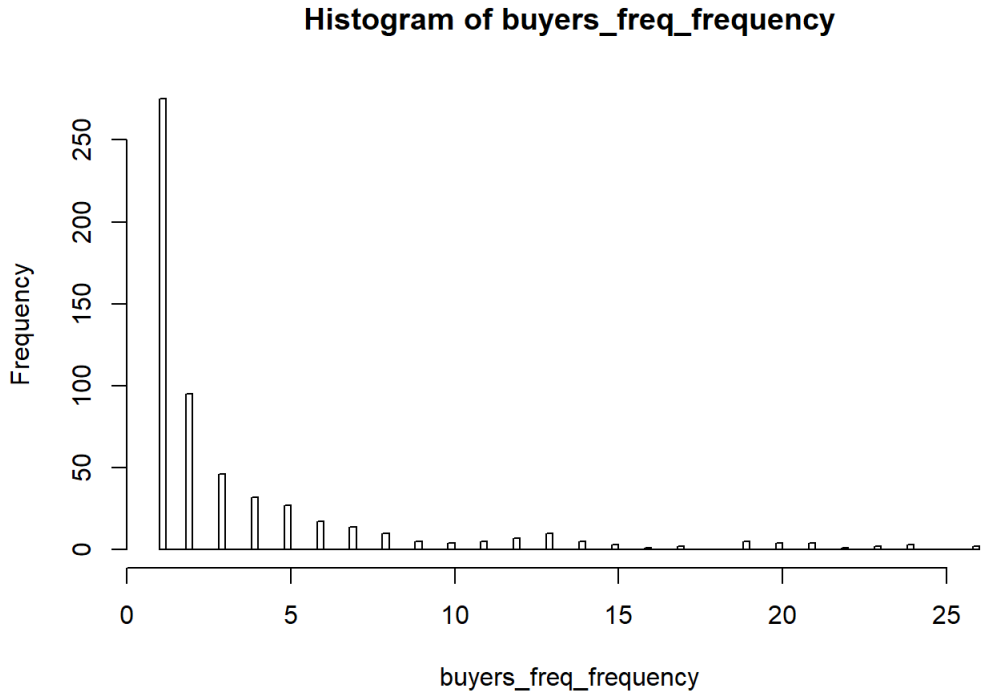


Summary of frequencies of buyers frequencies:

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   1.00    1.00    2.00   24.24    6.00 2621.00 
```

We removed some outliers by only keeping values < (.01*max value) as the median and max value varied greatly.

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  1.000   1.000   2.000   3.655   4.000  26.000 
```

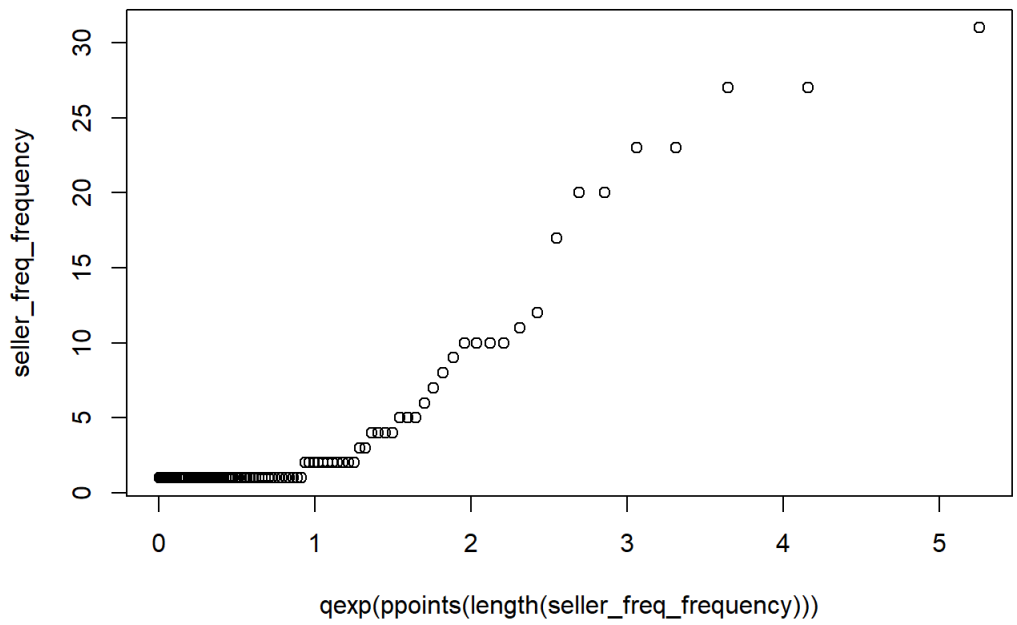


After analyzing the data and plotting the histogram for the same, we made an assumption that frequencies of sellers frequencies and frequencies of buyers frequencies follow exponential distributions.

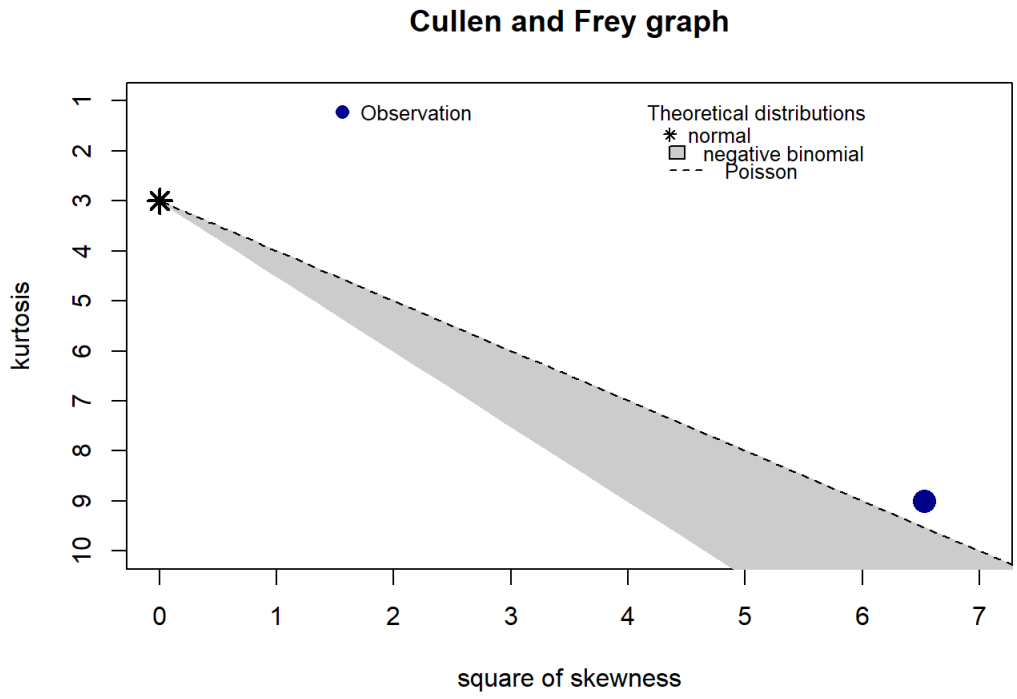
Analysis of the sellers and buyers data

We analyzed the frequencies of frequencies of sellers.

Plots for frequencies of frequencies of sellers with exponential distribution:

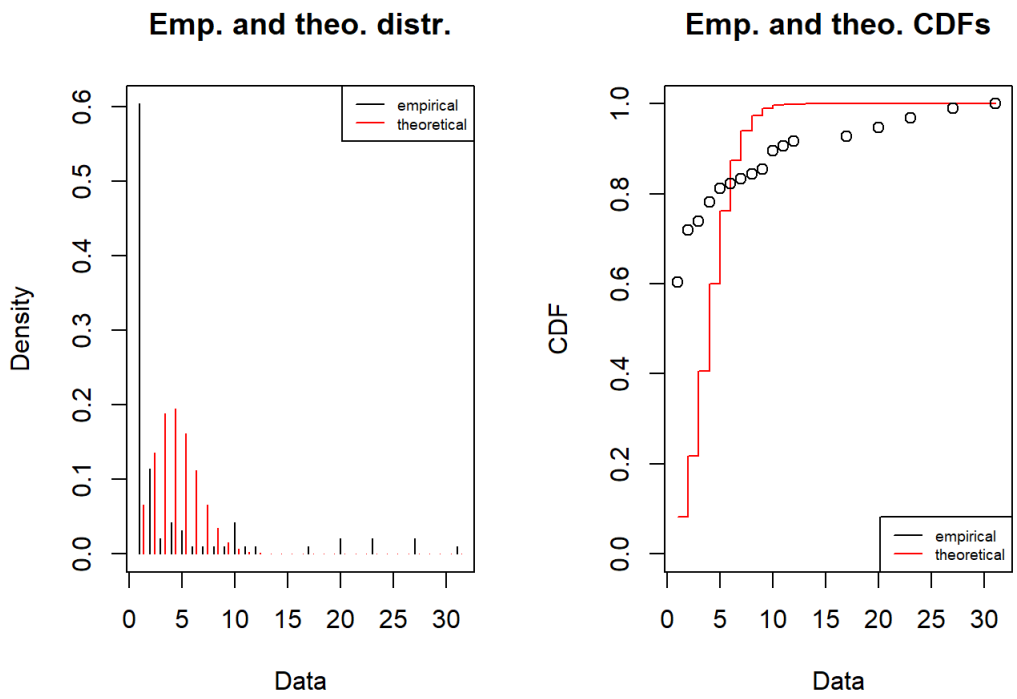


Now, we would draw the Cullen and Frey graph for this data set.



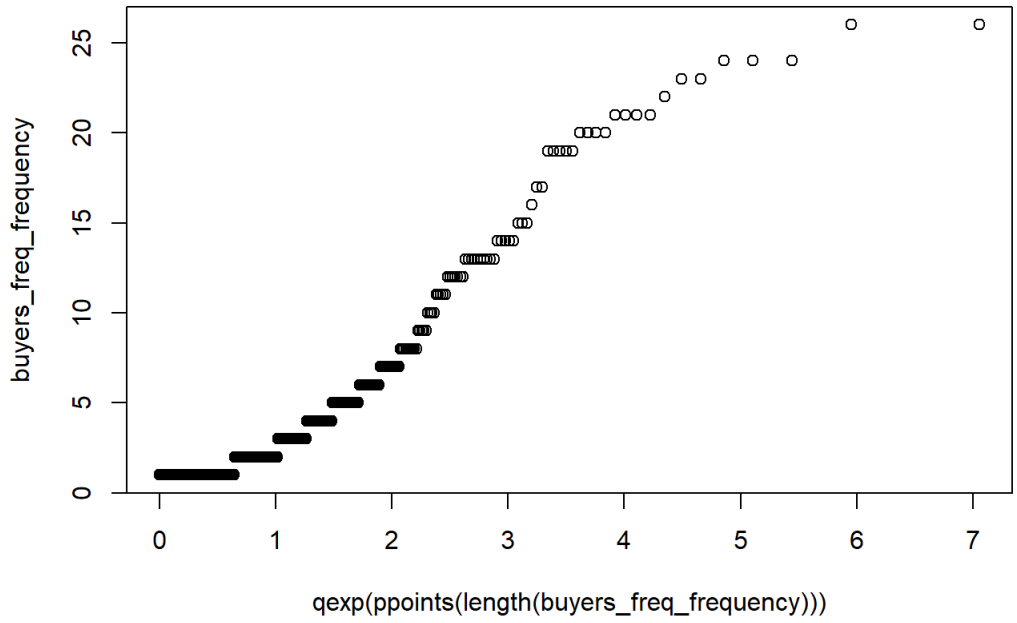

```
summary statistics
------
min:  1   max:  31
median:  1
mean:  4.145833
estimated sd:  6.545355
estimated skewness:  2.554525
estimated kurtosis:  9.002647
```

After drawing the Cullen and Frey graph, this looked somewhat close to Poisson distribution. As our assumption was not in line with our findings with Cullen and Frey graph, we would consider our prior assumption as false and plot according to our findings.

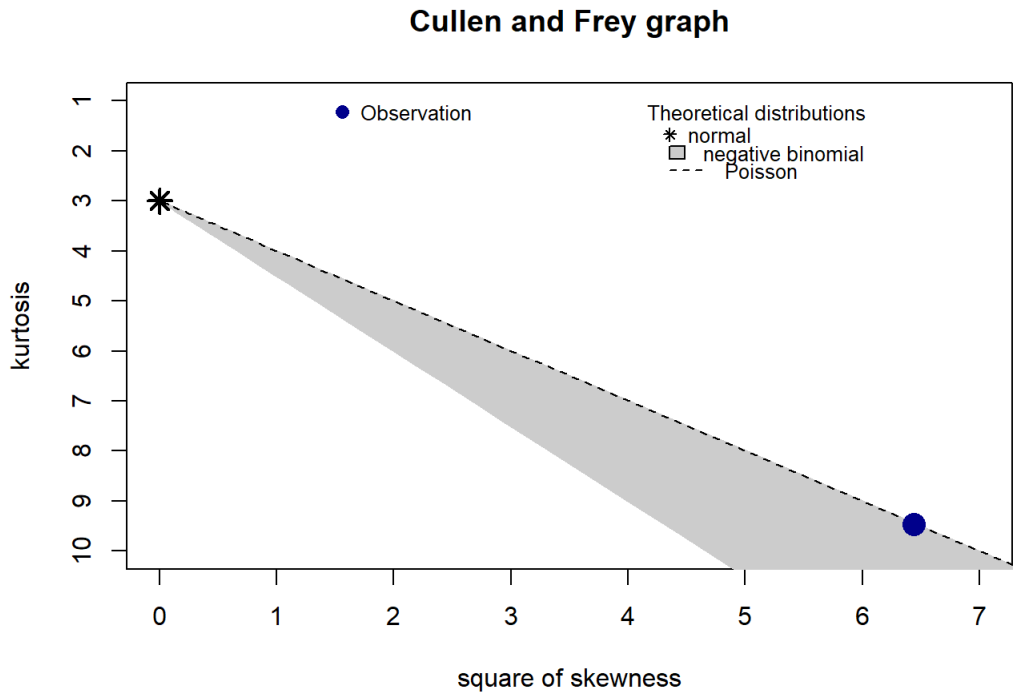


We analysed the frequencies of frequencies of buyers.

Plots for frequencies of frequencies of buyers with exponential distribution:

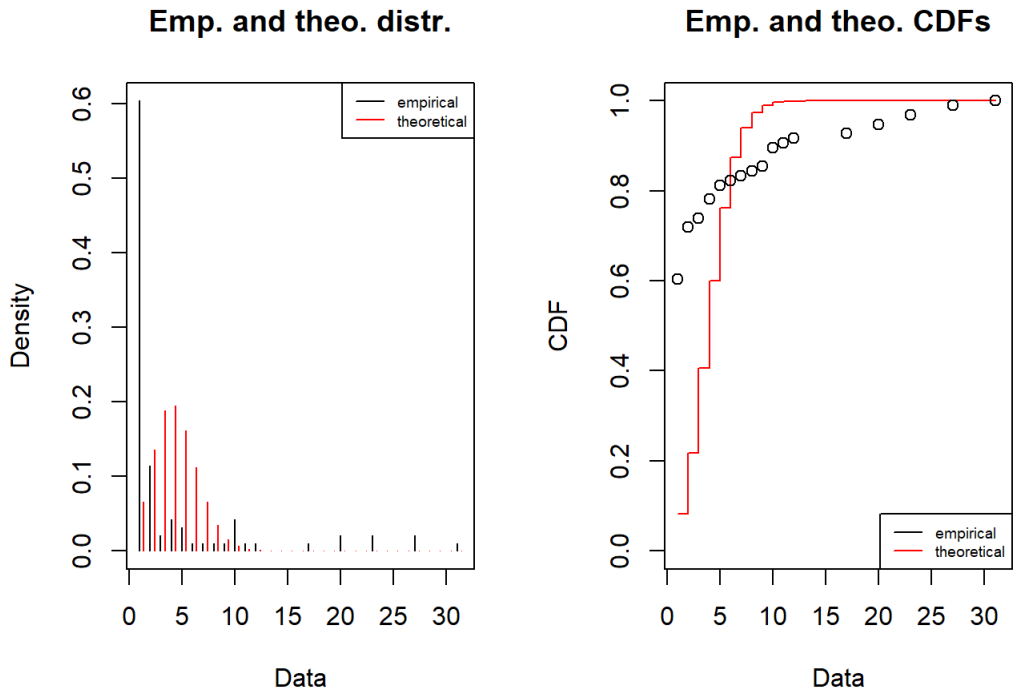


Now, we would draw the Cullen and Frey graph for this data set.




```
summary statistics
------
min:  1   max:  26
median:  2
mean:  3.654577
estimated sd:  4.714866
estimated skewness:  2.538258
estimated kurtosis:  9.465777
```

After drawing the Cullen and Frey graph, this looked somewhat close to Poisson distribution. As our assumption was not in line with our findings with Cullen and Frey graph, we would consider our prior assumption as false and plot according to our findings.



We would create multiple layers from the data and merge them with the highest price for each day and check the correlation between price and number of transactions using the Pearsons correlation formula.

Layer 1

```
Minimum trx val= 5.9991e+20 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :2936
```

```
Number of rows once this layer is removed :713387
```

```

    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 2.0143, df = 313, p-value = 0.04483
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.002652062 0.220873564
sample estimates:
      cor
0.1131269
```

Layer 2

```
Minimum trx val= 5.9991e+19 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :10748
```

```
Number of rows once this layer is removed :702639
```

```

    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = -4.2584, df = 425, p-value = 2.537e-05
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.2915907 -0.1094946
sample estimates:
       cor
-0.2022905
```

Layer 3

```
Minimum trx val= 5.9991e+18 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :41295
```

```
Number of rows once this layer is removed :661344
```

```

    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 1.3265, df = 431, p-value = 0.1854
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.03065788  0.15705739
sample estimates:
       cor
0.06376374
```

Layer 4

```
Minimum trx val= 5.9991e+17 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :156497
```

```
Number of rows once this layer is removed :504847
```

```

    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 3.4464, df = 431, p-value = 0.0006239
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.07061805 0.25408180
sample estimates:
      cor
0.1637656
```

Layer 5

```
Minimum trx val= 3.299505e+17 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :85221
```

```
Number of rows once this layer is removed :419626
```

```

    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 2.311, df = 365, p-value = 0.02139
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.01793621 0.21975455
sample estimates:
      cor
0.1200859
```

Layer 6

```
Minimum trx val= 2.9695545e+17 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :17349
```

```
Number of rows once this layer is removed :402277
```

```

    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 5.4215, df = 284, p-value = 1.264e-07
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.1972714 0.4077482
sample estimates:
      cor
0.3062478
```

Layer 7

```
Minimum trx val= 2.9695545e+16 Maximum trx val = 1.9997e+21
```

```
Number of rows in this layer :277734
```

```
Number of rows once this layer is removed :124543
```

```
    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 3.7158, df = 405, p-value = 0.0002311
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.08588027 0.27393769
sample estimates:
      cor
0.1815685
```

Correlation for the entire data set

```
    Pearson's product-moment correlation

data:  merged_table$High and merged_table$Freq
t = 2.5619, df = 485, p-value = 0.01071
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.02697443 0.20232875
sample estimates:
      cor
0.1155518
```

Findings from different layers of data

The correlation values for different layers are very small with Layer 6 having the highest correlation value of 0.3062478 between the number of transactions and the price of the coin.

Selecting other features

We tried with the following 4 features and checked if they had any significant correlation with stock opening price.

a> Average transaction per day.

b> Unique buyers.

c> Unique sellers.

d> (Closing price - opening price)/opening price.

Also, we did not take opening price each day directly. We computed the simple price return with the opening price of the next day. The goal of this exercise would be to select some good features and create a regression model with them.

Regression model

Correlation between unique buyers and opening price.

```
    Pearson's product-moment correlation

data:  feature_table$Open and feature_table$Buyers
t = -0.55231, df = 484, p-value = 0.581
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.11378887  0.06399151
sample estimates:
        cor
-0.02509711
```

Correlation between unique sellers and opening price.

```
    Pearson's product-moment correlation

data:  feature_table$Open and feature_table$Sellers
t = -0.73495, df = 484, p-value = 0.4627
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.12197178  0.05572302
sample estimates:
        cor
-0.03338824
```

Correlation between average token amount per transaction and opening price.

```
    Pearson's product-moment correlation

data:  feature_table$Open and feature_table$TokenAmounts
t = -0.71285, df = 484, p-value = 0.4763
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.12098261  0.05672383
sample estimates:
        cor
-0.03238533
```

Correlation between percentage change in price in a day and opening price.

```
    Pearson's product-moment correlation

data:  feature_table$Open and feature_table$Close
t = 258.15, df = 484, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.9956844 0.9969773
sample estimates:
      cor
0.9963882
```

So, only the daily price change among all the features had a very high correlation with the next day's opening price. We will now fit a linear model with all the features mentioned above.

We checked the median of the opening prices.

```
Median of the opening prices : -0.0115665855159506
```

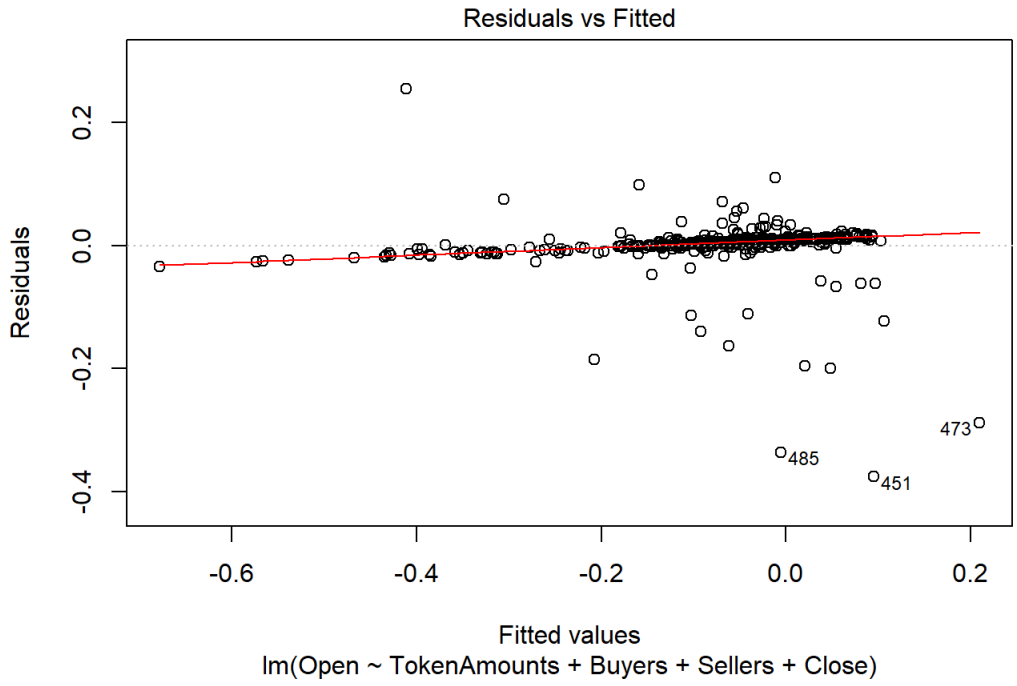
We checked the max value of the opening prices.

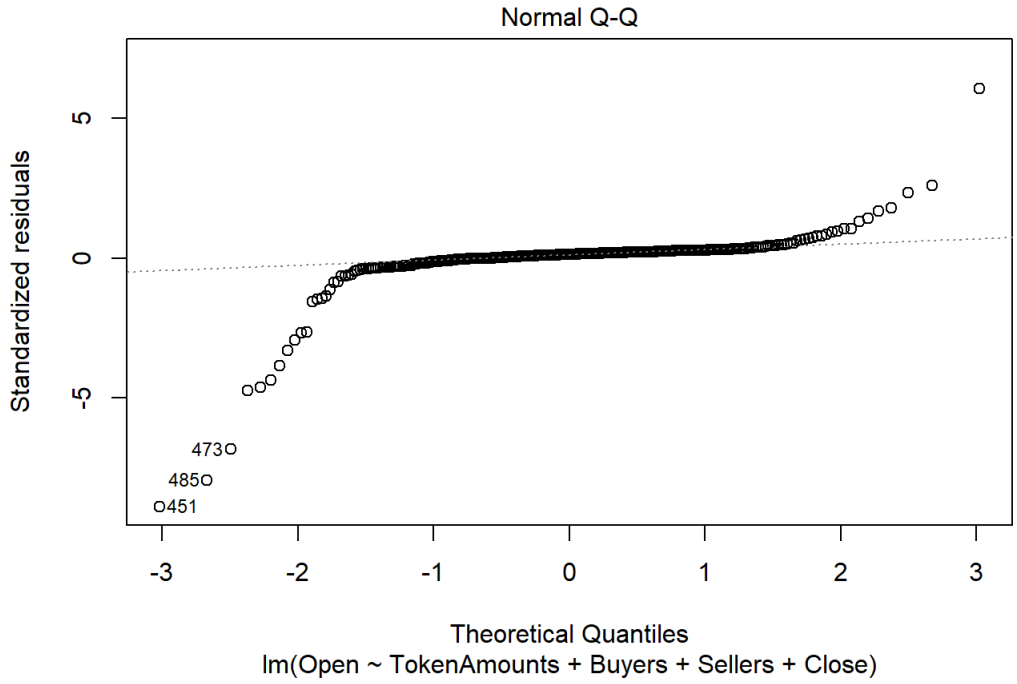
```
Maximum value of the opening prices :22.837852494577
```

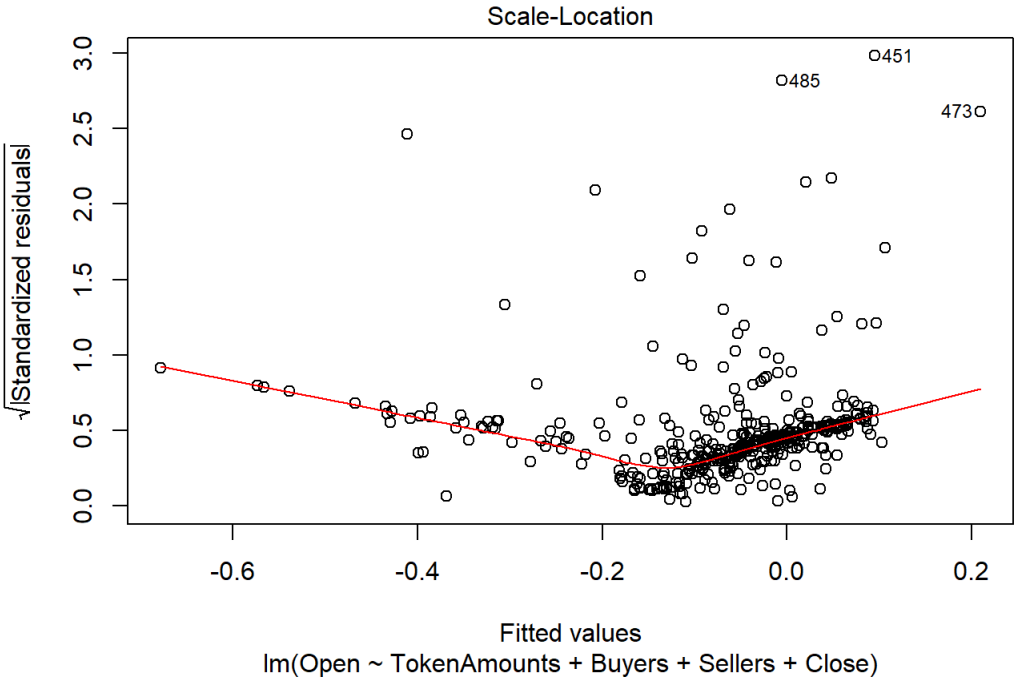
THe maxium value and median values differed by a lot. So we would do some preprocessing and removing all the rows that have opening value < 0.005*max(opening value).

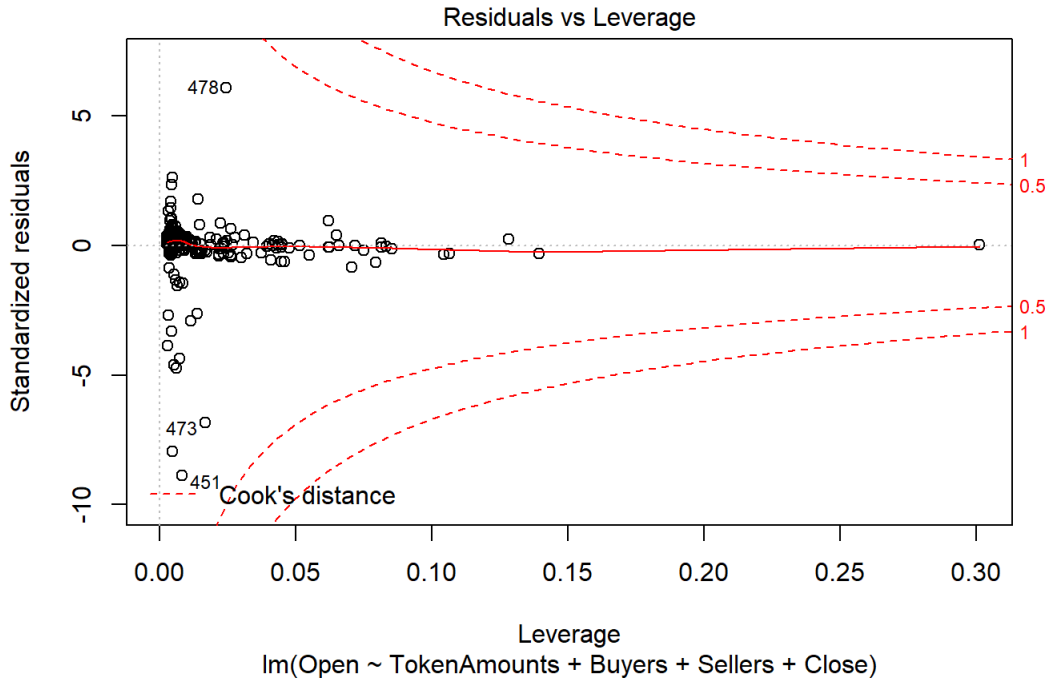
We would now fit the data in a regression model.

```
Call:
lm(formula = Open ~ TokenAmounts + Buyers + Sellers + Close,
    data = feature_table)

Residuals:
     Min       1Q   Median       3Q      Max
-0.37645 -0.00060  0.00574  0.01019  0.25480

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -9.141e-03  2.931e-03  -3.119  0.00195 **
TokenAmounts  1.216e-26  1.393e-25   0.087  0.93047
Buyers        3.243e-06  4.231e-06   0.766  0.44387
Sellers       7.284e-06  2.959e-05   0.246  0.80567
Close         9.426e-01  1.651e-02  57.102  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04253 on 392 degrees of freedom
Multiple R-squared:  0.8948,    Adjusted R-squared:  0.8937
F-statistic: 833.4 on 4 and 392 DF,  p-value: < 2.2e-16
```




Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Open ~ TokenAmounts + Buyers + Sellers + Close)
473
485
451


Scale-Location
√|Standardized residuals|
Fitted values
lm(Open ~ TokenAmounts + Buyers + Sellers + Close)
451
485
473



Observations from the linear model

We found that most of the points were along the fitted models and the residuals for those points are very less. There were a few points that remained outside, but we kept them as they were not varying with a great degree with other points and should not be considered as outliers.

Even though we forcefully fitted multiple transaction information as features, the most correlated feature was the daily price change and that yielded us result better than the result computed otherwise. Our linear model generated very less residual values for most of the points, so it was to some extent adequate to represent our dataset.

Randomforest model for price prediction

We split the available price table into training and validation sets and used the random forest to predict the house prices and got the following mean squared error value for the training and validation set.

30% of the available data were kept for validation.

```
Call:
 randomForest(formula = Open ~ ., data = train, ntree = 1000,      mtry = 4, importance = TRUE)
               Type of random forest: regression
                     Number of trees: 1000
No. of variables tried at each split: 4

          Mean of squared residuals: 0.002140342
                    % Var explained: 87.6
```

```
MSE for training data: 0.000453102657437817
```

```
MSE for validation data: 0.00285253290861994
```

Conclusion

After analyzing the YOCOIN data we found out that it's price did not depend on the transaction information greatly. We got a rather high correlation with the coin price of the day before. Also barring few spikes the there was not much jump in the coin prices as well. So we do not have enough data to hypothesize any conclusion regarding how the price should change for this token. We need more data to have a deeper understanding of the transactions. For example looking at the transaction data, we can not tell how this affects the net available coins in the market and so on. Gathering that information will help us greatly to find a strong model for YOCOIN.