Cryptocurrency Market Data Analysis

Using Data Mining techniques and Time Series Analysis to find the Market Analysis of Cryptocurrency Data.

Submitted by -

Karishma Padale

Introduction

A cryptocurrency is an advanced or virtual currency that uses cryptography for security. A cryptocurrency is hard to fake on account of this security include. Numerous digital currencies are decentralized frameworks dependent on blockchain innovation, a disseminated record implemented by a dissimilar system of PCs. It is an alternative form of payment to cash, credit cards, and checks. The technology behind it allows you to send it directly to others without going through a 3rd party like a bank. The number of cryptocurrencies available over the internet like Bitcoin, Ripple, Ethereum few to be named. A new cryptocurrency can be created at any time. Cryptocurrency is completely anonymous, which is great for those that value their online privacy and are wary of handing over too much of their digital data. Its major advantage is its transparency and 24-hour accessibility. Using Cryptocurrency makes transactions easy and free with no additional charges which we pay during our normal money transfer with banks. Cryptocurrencies are safe to use but they are highly volatile. The most popular of all the Cryptocurrencies is Bitcoin.

Analysis

Bitcoin do not have centralize controlling authority and it was introduced by Satoshi Nakamoto in years 2009. Bitcoin trade happens between peers based on mutual trust which at first looks very suspicious and dangerous but due to use of Blockchain technology it is almost impossible to get hacked or cheated by malicious users. Bitcoin holders are using this virtual currency for trading things freely without any meddling of authority and without any fear of getting your account freeze due to some reason.

Most importantly Bitcoin claims to have solved below issues which are regularly observed in conventional financial system

1) Fake Currency

2) Long Banking Time

3) Bank Collapse

4) Financial Crisis (2008 money related emergency)

We do have PayPal, Paytm, Citrus however again they are reliant on banks. Some more advantages

1) No exchange expenses are required - All exchanges are finished inside moment

2)It is decentralized in that nobody individual claim blockchain organize

3) It can never be controlled or hacked as a result of the essential structure of Blockchain

About the Dataset:

Dataset was downloaded from https://www.kaggle.com/jessevent/all-crypto-currencies/home , which included historical price information of some of the top crypto currencies by market capitalization , the following fields are as follows:-

Date: date of observation

Open : Opening price on the given day

High : Highest price on the given day

Low : Lowest price on the given day

Close : Closing price on the given day

Volume : Volume of transactions on the given day

Market Cap : Market capitalization in USD

Close ratio: It is the daily close rate, min-maxed with the high and low values for the day.

Close Ratio = (Close-Low)/(High-Low)

Spread: It is the $USD difference between the high and low values for the day.

Slug : Info about type of currencies, which are as follows:-

1. Bitcoin
2. Ethereum
3. Ripple
4. Bitcoin cash
5. Bit connect
6. Dash
7. Ethereum Classic
8. Iota
9. Litecoin
10. Monero , etc.

R-code:

Read the data from csv file and load it

>> data = read.csv("crypto-markets.csv")

#Structure of data

>> str(data)

Output:

```
> str(data)
'data.frame':   942297 obs. of  13 variables:
 $ slug       : Factor w/ 2071 levels "0chain","0x",..: 209 209 209 209 209 209 209 209 209 209 ...
 $ symbol     : Factor w/ 2005 levels "$$$","$PAC","0XBTC",..: 275 275 275 275 275 275 275 275 275 275 ...
 $ name       : Factor w/ 2071 levels "0chain","0x",..: 205 205 205 205 205 205 205 205 205 205 ...
 $ date       : Factor w/ 2043 levels "01/01/14","01/01/15",..: 587 593 599 605 611 617 623 629 635 641 ...
 $ ranknow    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ open       : num  135 134 144 139 116 ...
 $ high       : num  136 147 147 140 126 ...
 $ low        : num  132.1 134 134.1 107.7 92.3 ...
 $ close      : num  134 145 139 117 105 ...
 $ volume     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ market     : num  1.49e+09 1.60e+09 1.54e+09 1.30e+09 1.17e+09 ...
 $ close_ratio: num  0.544 0.781 0.384 0.288 0.388 ...
 $ spread     : num  3.88 13.49 12.88 32.17 33.32 ...
```

head(data)

returns top 6 records

Output:

```
> head(data)
     slug symbol    name     date ranknow   open   high    low  close volume     market close_ratio
1 bitcoin    BTC Bitcoin 04/28/13       1 135.30 135.98 132.10 134.21      0 1488566728      0.5438
2 bitcoin    BTC Bitcoin 04/29/13       1 134.44 147.49 134.00 144.54      0 1603768865      0.7813
3 bitcoin    BTC Bitcoin 04/30/13       1 144.00 146.93 134.05 139.00      0 1542813125      0.3843
4 bitcoin    BTC Bitcoin 05/01/13       1 139.00 139.89 107.72 116.99      0 1298954594      0.2882
5 bitcoin    BTC Bitcoin 05/02/13       1 116.38 125.60  92.28 105.21      0 1168517495      0.3881
6 bitcoin    BTC Bitcoin 05/03/13       1 106.25 108.13  79.10  97.75      0 1085995169      0.6424
  spread
1   3.88
2  13.49
3  12.88
4  32.17
5  33.32
6  29.03
```

Context of Study:

Things like Block chain, Bitcoin, Bitcoin money, Ethereum, Ripple and so forth are always coming in the news stories that we read. Along these lines, we needed to see progressively about it which made us to begin with this analysis. Once the basics are done, the questions we felt which needs to be answerable were:

1. How numerous Cryptocurrencies are there and what are their costs and valuations?
2. Why is there a sudden surge in the interest in recent days?

For finding solutions to every one of these questions (and if conceivable to foresee the future costs), we began gathering information from https://coinmarketcap.com/about the cryptocurrencies.

Since we have the value information, it was expected to burrow somewhat more about the elements influencing the cost of coins. At that point we began off with Bitcoin and there are many parameters which influence the cost of Bitcoin. Because of https://www.blockchain.com/traveler , we had the option to get many parameters on once in two-day premise. This will help comprehend different components identified with Bitcoin cost and help one make future forecasts in a superior manner than simply utilizing the verifiable cost.
Inferences needed to be drawn from the data set:
Some of the questions which could be inferred from this dataset are:

1. How did the historical prices / market capitalizations of currencies change over time?
2. Predicting the future price of the currencies.

Data Preprocessing:

Steps involved in data preprocessing included:

Installing Packages and Loading Libraries

R-code:

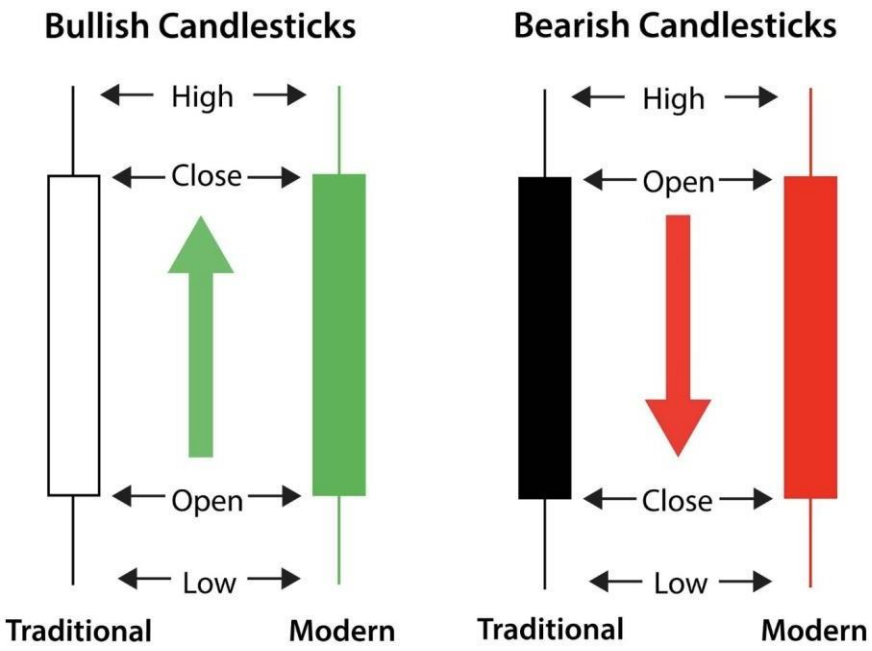
```
# Install packages and load libraries
>> install.packages("colorspace")
# Is used in Manipulating and Assessing colors and Palettes
>> install.packages("lubridate") # Is used to format dates
>> install.packages("dplyr")
>> install.packages("stringi") # String Preprocessing
>> install.packages("prophet”) # Is used to implement prophet model
>> library(dplyr)
>> library(prophet)
>> library(lubridate)
>> library(colorspace)
```

The next step is to subset the Bitcoin data from the rest of the dataset as we are going to work on only Bitcoin. But first why and how we chose Bitcoin only we will explain that through Candlestick chart.

What is Candlestick chart?

Given below image shows what is Candlestick chart and what is represents. Just like a bar chart, a daily candlestick shows the market's open, high, low, and close price for the day. The candlestick has a wide part, which is called the "real body."

This real body represents the price range between the open and close of that day's trading. When the real body is filled in or black, it means the close was lower than the open. If the real body is empty, it means the close was higher than the open.



Using our data, we have the Candlestick chart for top 3 currencies that are Bitcoin, Ethereum and Ripple.

Code:

```
View(df)
#candlechart for BITCOIN

BTC_df<-BTC
colnames(BTC_df)<-tolower(colnames(BTC_df))
str(BTC_df)
BTC_df$date<-as.Date(BTC_df$date)
BTC_df<-tail(BTC_df,200)
v1<-c(1:nrow(BTC_df))
v2<-seq(1,nrow(BTC_df),5)
color<-ifelse(BTC_df$close>=BTC_df$open,"green","red")
png("Bitcoin",res = 500,height = 5000,width = 9000)
```

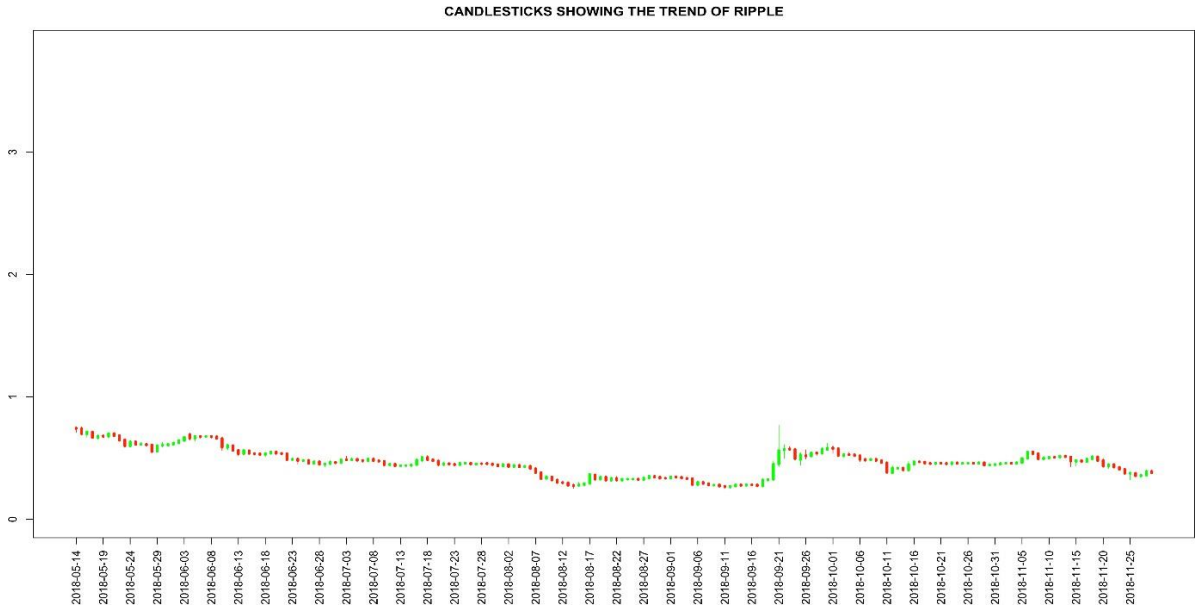
```
plot(BTC_df$high,main = "candlechart", xaxt="n",xlab = " ",ylab = "Price",ylim =
c(BTC_lowest,BTC_highest),type = "n")
par(new=T)
plot(BTC_df$low,main = "candlechart", xaxt="n",xlab = " ",ylab = "",ylim =
c(BTC_lowest,BTC_highest),type = "n")
segments(x0=v1,y0=BTC_df$open,x1=v1,y1=BTC_df$close,col = color,lwd = 4)
segments(x0=v1,y0=BTC_df$low,x1=v1,y1=BTC_df$high,col = color,lwd = 1)
axis(1, at=v2,labels = BTC_df$date[v2],las = 2)
```

Output:

Candlestick chart showing the trend of Bitcoin.

Candlestick chart showing the trend of Ethereum.

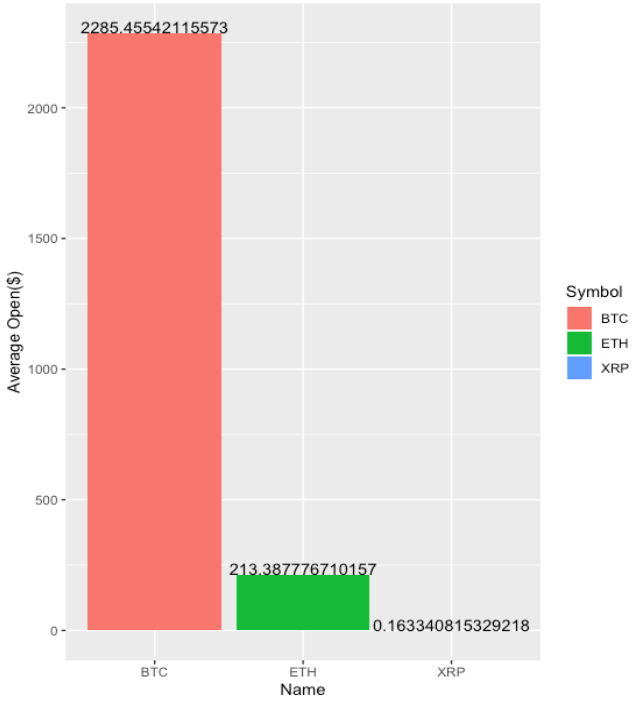
Candlestick chart showing the trend of Ripple.



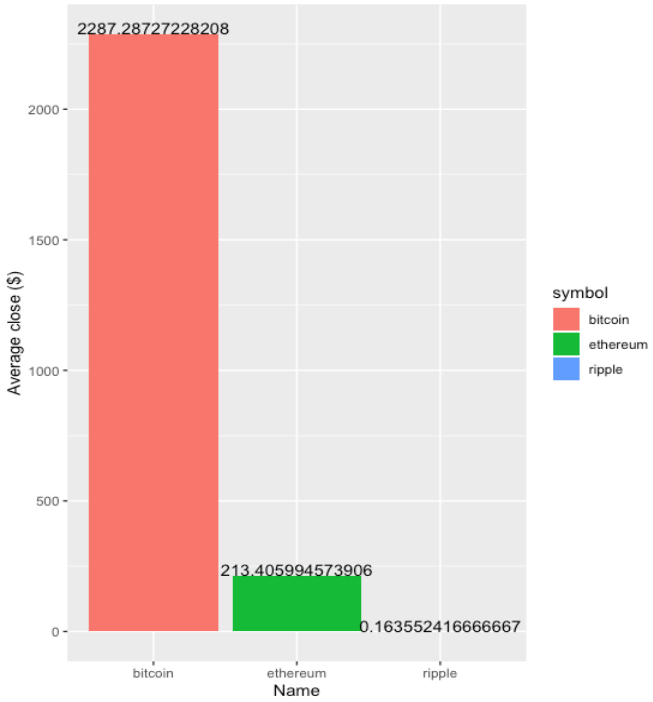
This representation of Candlestick chart shows the trends of these currencies. As we can see that the trend of bitcoin shows that people invest more in bitcoins. It is always trending. The opening and closing market of bitcoin is always high than the other two(Ethereum and Ripple).

We have also shown this with further analysis shown below:

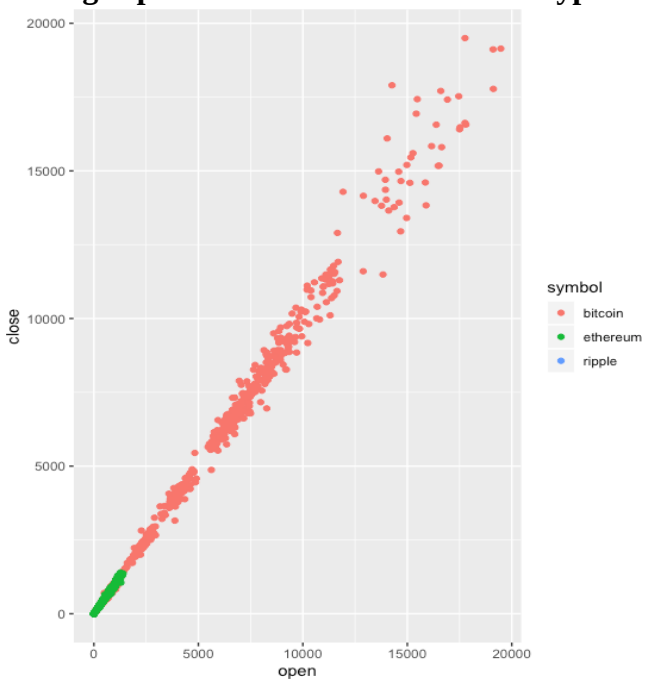
Average open market of Cryptocurrencies



Average close market of Cryptocurrencies



Scatter plot showing both average open and close of market for Cryptocurrencies.



The subsequent stage is to subset the Bitcoin information from the remainder of the dataset as we are getting down to business on just Bitcoin.

R-code:

#Separating the bitcoin data from rest

>> data_bitcoin = subset(data,(data$name == 'Bitcoin'))

>> data_bitcoin # display the data

Output:

```
> data_bitcoin
      slug symbol    name     date ranknow   open   high    low  close volume     market close_ratio
1  bitcoin    BTC Bitcoin 04/28/13       1 135.30 135.98 132.10 134.21      0 1488566728      0.5438
2  bitcoin    BTC Bitcoin 04/29/13       1 134.44 147.49 134.00 144.54      0 1603768865      0.7813
3  bitcoin    BTC Bitcoin 04/30/13       1 144.00 146.93 134.05 139.00      0 1542813125      0.3843
4  bitcoin    BTC Bitcoin 05/01/13       1 139.00 139.89 107.72 116.99      0 1298954594      0.2882
5  bitcoin    BTC Bitcoin 05/02/13       1 116.38 125.60  92.28 105.21      0 1168517495      0.3881
6  bitcoin    BTC Bitcoin 05/03/13       1 106.25 108.13  79.10  97.75      0 1085995169      0.6424
7  bitcoin    BTC Bitcoin 05/04/13       1  98.10 115.00  92.50 112.50      0 1250316563      0.8889
8  bitcoin    BTC Bitcoin 05/05/13       1 112.90 118.80 107.14 115.91      0 1288693176      0.7521
9  bitcoin    BTC Bitcoin 05/06/13       1 115.98 124.66 106.64 112.30      0 1249023060      0.3141
10 bitcoin    BTC Bitcoin 05/07/13       1 112.25 113.44  97.70 111.50      0 1240593600      0.8767
11 bitcoin    BTC Bitcoin 05/08/13       1 109.60 115.78 109.60 113.57      0 1264049202      0.6424
12 bitcoin    BTC Bitcoin 05/09/13       1 113.20 113.46 109.26 112.67      0 1254535382      0.8119
13 bitcoin    BTC Bitcoin 05/10/13       1 112.80 122.00 111.55 117.20      0 1305479080      0.5407
14 bitcoin    BTC Bitcoin 05/11/13       1 117.70 118.68 113.01 115.24      0 1284207489      0.3933
15 bitcoin    BTC Bitcoin 05/12/13       1 115.64 117.45 113.43 115.00      0 1281982625      0.3905
16 bitcoin    BTC Bitcoin 05/13/13       1 114.82 118.70 114.50 117.98      0 1315710011      0.8286
17 bitcoin    BTC Bitcoin 05/14/13       1 117.98 119.80 110.25 111.50      0 1243874488      0.1309
18 bitcoin    BTC Bitcoin 05/15/13       1 111.40 115.81 103.50 114.22      0 1274623813      0.8708
19 bitcoin    BTC Bitcoin 05/16/13       1 114.22 118.76 112.20 118.76      0 1325726787      1.0000
20 bitcoin    BTC Bitcoin 05/17/13       1 118.21 125.30 116.57 123.01      0 1373723882      0.7377
21 bitcoin    BTC Bitcoin 05/18/13       1 123.50 125.25 122.30 123.50      0 1379574546      0.4068
22 bitcoin    BTC Bitcoin 05/19/13       1 123.21 124.50 119.57 121.99      0 1363204703      0.4909
23 bitcoin    BTC Bitcoin 05/20/13       1 122.50 123.62 120.12 122.00      0 1363709900      0.5371
24 bitcoin    BTC Bitcoin 05/21/13       1 122.02 123.00 121.21 122.88      0 1374013440      0.9330
25 bitcoin    BTC Bitcoin 05/22/13       1 122.89 124.00 122.00 123.89      0 1385778993      0.9450
```

Analyzing Data:

 # Creating a time series

R-code:

>> data_bitcoin_series <- ts(data_bitcoin$close, frequency=12, start=2013, end = 2019)

>> data_bitcoin_series# display the time series data

Output:

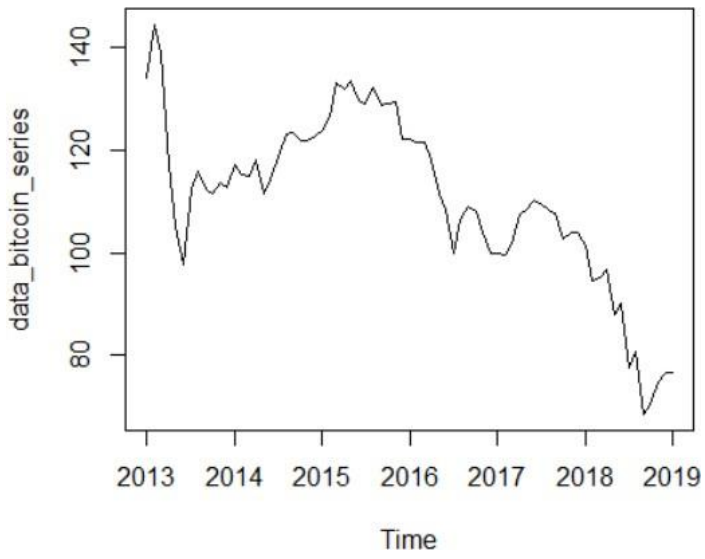
```
> data_bitcoin_series
        Jan    Feb    Mar    Apr    May    Jun    Jul    Aug    Sep    Oct    Nov    Dec
2013 134.21 144.54 139.00 116.99 105.21  97.75 112.50 115.91 112.30 111.50 113.57 112.67
2014 117.20 115.24 115.00 117.98 111.50 114.22 118.76 123.01 123.50 121.99 122.00 122.88
2015 123.89 126.70 133.20 131.98 133.48 129.74 129.00 132.30 128.80 129.00 129.30 122.29
2016 122.22 121.42 121.65 118.00 111.50 108.30 100.00 106.35 108.90 108.15 104.00  99.98
2017  99.99  99.51 101.70 107.40 108.25 110.15 109.50 108.30 107.60 102.74 103.95 104.00
2018 101.44  94.65  94.99  96.61  88.05  90.13  77.53  80.53  68.43  70.28  74.56  76.52
2019  76.69
```

The ts function converts the numeric data to time series, start parameter is used to specify the starting point of the series.

Plotting Time series

>> class(data_bitcoin_series)

>> plot(data_bitcoin_series)



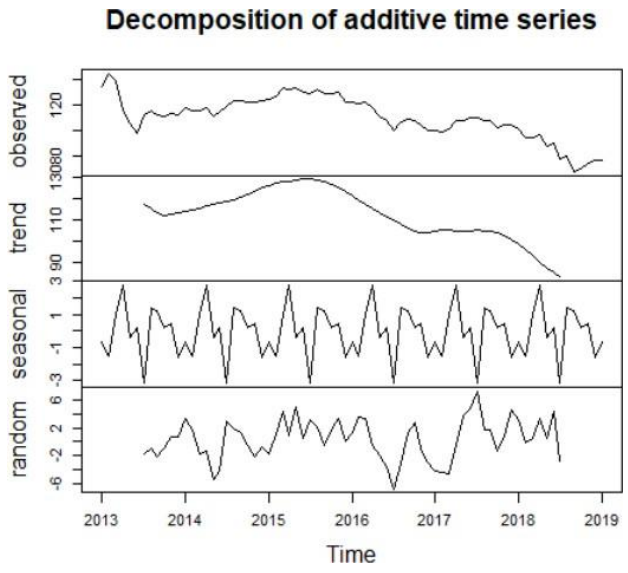
We can see from the above plot of time series and the variance of the bitcoin is by all accounts not symmetric.

Decompose Time Series

#decompose time series

>> decomp_data_bitcoin_series <- decompose(data_bitcoin_series)

>> plot (decomp_data_bitcoin_series)



>> Above can be seen the decomposition components of trends and seasonality. Pattern shows to go down with time. We check the stationarity of the information.

Here we get 4 components:

- Observed – the actual data plot
- Trend – the overall upward or downward movement of the data points
- Seasonal – any monthly/yearly pattern of the data points
- Random – unexplainable part of the data

The next step is to select the date and the closing columns.

R-code:

#Selecting the date and closing value of the day in the dataset

>> newdata <- data_bitcoin[c(4,9)]

>> tail(newdata)

displays the last 6 records

Output:

```
> tail(newdata)
         date   close
2037 11/24/18 3880.76
2038 11/25/18 4009.97
2039 11/26/18 3779.13
2040 11/27/18 3820.72
2041 11/28/18 4257.42
2042 11/29/18 4278.85
>
```

Here we can see that the date format is m/d/Y format we need to change the date format for the further process.

R-code:

Changing the date format

>> newdata$date = as.Date(newdata$date, format = "%m/%d/%y")

>> tail(newdata)#display last 6 records

as.Date() is used to set the format of the date column. Here we are changing the date column format to "%m/%d/%y".

Output:

```
> tail(newdata)
           date   close
2037 2018-11-24 3880.76
2038 2018-11-25 4009.97
2039 2018-11-26 3779.13
2040 2018-11-27 3820.72
2041 2018-11-28 4257.42
2042 2018-11-29 4278.85
```

Plotting date and closing value for analysis.

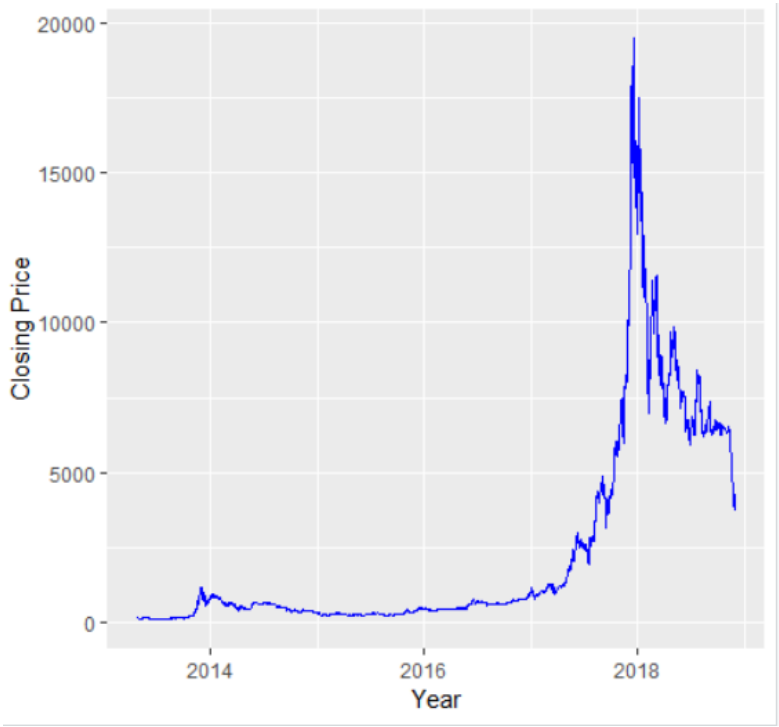
R-code:

#Plotting the data

>> colnames(newdata) <- c("ds", "y") # rename the columns

>> plot(y ~ ds, newdata)

Output:



The above ggplot() demonstrates the end rate of each date of digital currency = Bitcoin from 2013 to finish of 2018. We can see that in the year 2017 the bitcoin began its development in market and before the finish of 2017, bitcoin achieved the most astounding pinnacle. Before long the bitcoin worth began to fall and till now its dropping.

Data Modelling:

We are going to utilize one of the best estimating calculations known as prophet. Prophet is a method for estimating time arrangement information dependent on an added substance model where non-straight patterns are fit with yearly, week by week, and every day regularity, in addition to occasion impacts. It works best with time arrangement that have solid occasional impacts and a few periods of authentic information.

Prophet Model: When applied to data frame data_bitcoin_analysis, to forecast price of bitcoin 90 days in future using the fit and predict function as follows:

R-code:

#To forecast 90 days further

>> m <- prophet(newdata, daily.seasonality = TRUE)# using prophet model to forecast

>> future <- make_future_dataframe(m, periods = 180)

>> tail(future)

Output:

```
> tail(future)
           ds
2217 2019-05-23
2218 2019-05-24
2219 2019-05-25
2220 2019-05-26
2221 2019-05-27
2222 2019-05-28
```

Prophet has a built-in helper function make_future_dataframe to create a data frame of future dates. The make_future_dataframe function lets you specify the frequency and number of periods you would like to forecast into the future.

#Predict() is used to make predictions for each row in the future

>> forecast <- predict(m, future)

>> tail(forecast)

Output:

```
> tail(forecast)
           ds    trend additive_terms additive_terms_lower additive_terms_upper    daily daily_lower daily_upper
2217 2019-05-23 6232.083       98.62366             98.62366             98.62366 201.9134    201.9134    201.9134
2218 2019-05-24 6227.884       99.10669             99.10669             99.10669 201.9134    201.9134    201.9134
2219 2019-05-25 6223.684      101.48974            101.48974            101.48974 201.9134    201.9134    201.9134
2220 2019-05-26 6219.484       84.87240             84.87240             84.87240 201.9134    201.9134    201.9134
2221 2019-05-27 6215.285       84.41670             84.41670             84.41670 201.9134    201.9134    201.9134
2222 2019-05-28 6211.085       81.29007             81.29007             81.29007 201.9134    201.9134    201.9134
         weekly weekly_lower weekly_upper     yearly yearly_lower yearly_upper multiplicative_terms
2217 -7.2299746   -7.2299746   -7.2299746  -96.05979    -96.05979    -96.05979                   0
2218  3.2161861    3.2161861    3.2161861 -106.02292   -106.02292   -106.02292                   0
2219 12.9358982   12.9358982   12.9358982 -113.35958   -113.35958   -113.35958                   0
2220  1.0186306    1.0186306    1.0186306 -118.05966   -118.05966   -118.05966                   0
2221  2.6968913    2.6968913    2.6968913 -120.19362   -120.19362   -120.19362                   0
2222 -0.7122504   -0.7122504   -0.7122504 -119.91111   -119.91111   -119.91111                   0
     multiplicative_terms_lower multiplicative_terms_upper yhat_lower yhat_upper trend_lower trend_upper     yhat
2217                          0                          0   4336.851   8124.493    5213.253    7238.745 6330.707
2218                          0                          0   4466.759   7993.514    5195.510    7245.062 6326.990
2219                          0                          0   4580.450   8214.429    5170.764    7253.144 6325.174
2220                          0                          0   4264.028   8282.115    5146.383    7260.938 6304.357
2221                          0                          0   4439.301   8187.452    5131.192    7267.444 6299.701
2222                          0                          0   4295.783   8011.780    5117.183    7273.726 6292.375
```

R-code:

>> tail(forecast[c('ds', 'yhat', 'yhat_lower', 'yhat_upper')])

Output:

```
> tail(forecast[c('ds', 'yhat', 'yhat_lower', 'yhat_upper')])
            ds     yhat yhat_lower yhat_upper
2217 2019-05-23 6330.707   4336.851   8124.493
2218 2019-05-24 6326.990   4466.759   7993.514
2219 2019-05-25 6325.174   4580.450   8214.429
2220 2019-05-26 6304.357   4264.028   8282.115
2221 2019-05-27 6299.701   4439.301   8187.452
2222 2019-05-28 6292.375   4295.783   8011.780
```

The upper and lower limit of the prediction for next 6 months will be given by the above-mentioned values.

Terms used in forecast matrix:

ds -> denotes the future dates

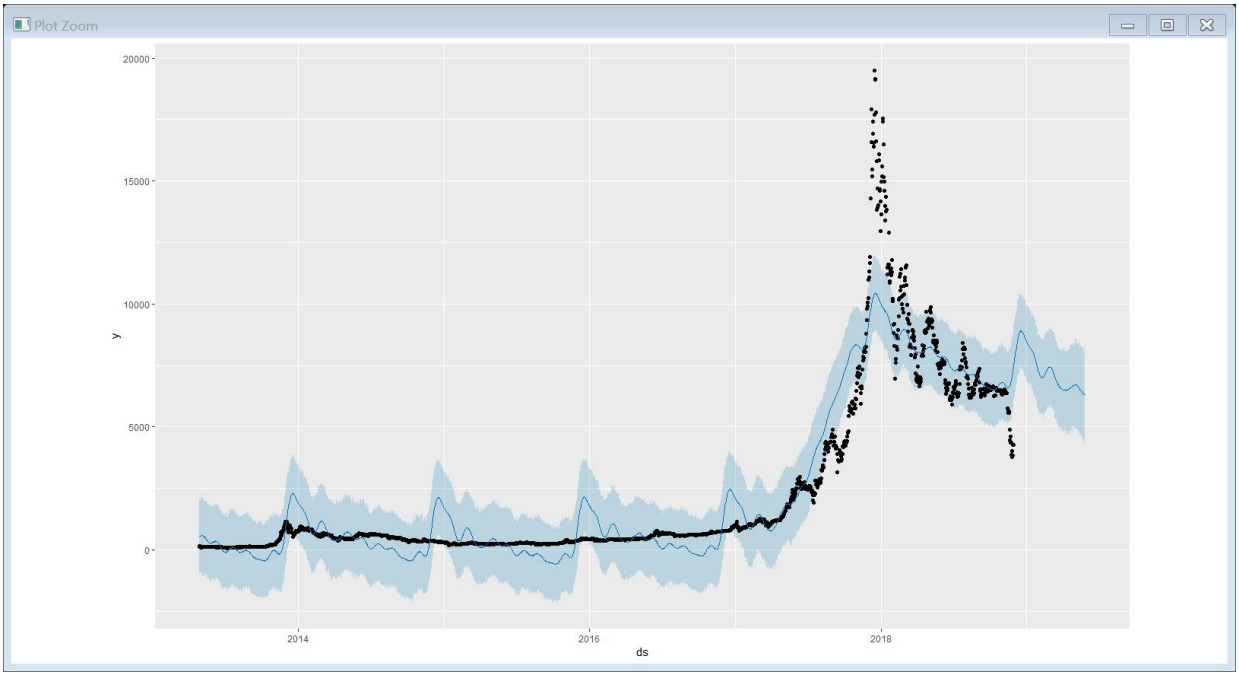
yhat -> predicted value of bitcoin price on that specific day

yhat_lower, yhat_upper -> confidence intervals

Plotting the Forecast:

>> plot (m, forecast)

Output



The above plot talks about the anticipated estimation of bitcoin until the period it was mentioned. We could see from the plot that there was an abrupt increment in its incentive since mid of 2017, for example (as far back as financial specialists began accepting this installment mode).

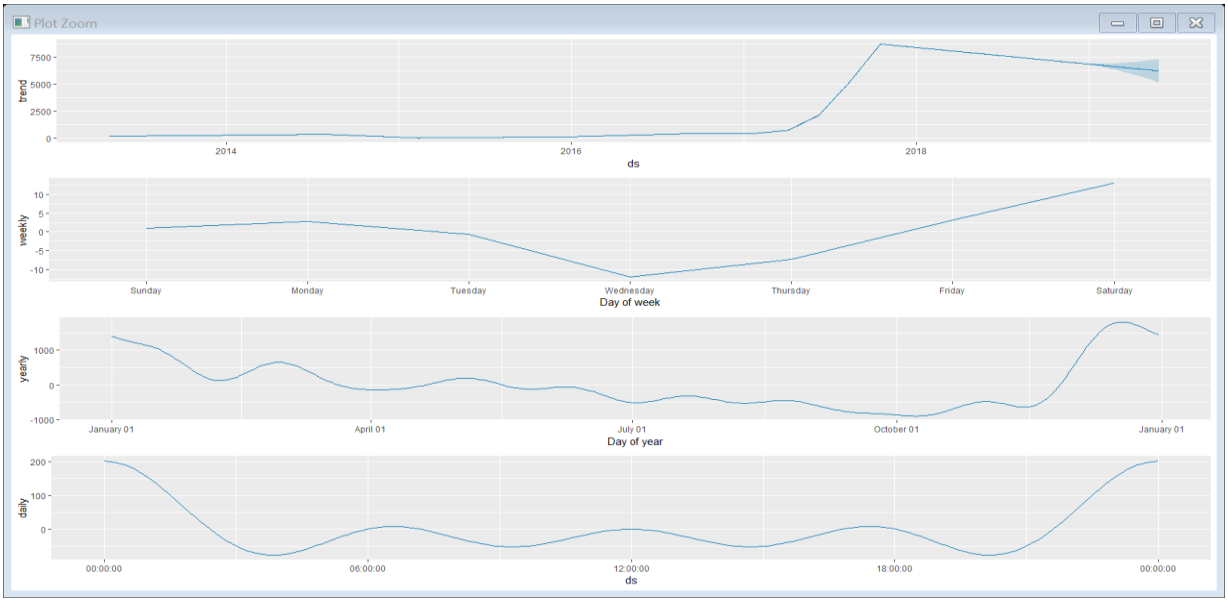
We can break down the analysis on raw data to its sub components of time series using the components () in prophet, this is as follows:

Plotting the prophet components

#components plot

>> prophet_plot_components(m, forecast)

Output



Taking a gander at the week by week pattern, there does not have all the earmarks of being any significant sign This is not out of the ordinary as the irregular walk hypothesis in financial aspects states there is no anticipated example in stock costs every day. As confirm by our analysis, over the long haul, bitcoin cost will in general increment, yet on an everyday scale, there is no example that we can exploit even with the best models.

Conclusion

After analyzing the data, we found that bitcoin covers the most extreme market of cryptocurrencies with different digital forms of money, for example, Ethereum, Litecoin and so forth. The time estimation of cash is significant so subsequent to anticipating the estimation of Bitcoin for next 180 days we found that there is no indicate change in the worth that we can say the worth is going to rise or it will fall. Thus, the financial specialists simply must be cautious that when the market drops, don't pull back in light of the fact that it will return up as indicated by history. On the general scale, the everyday variances are too little to even consider evening be seen and on the off chance that we are thinking like data scientists, we understand that playing day by day stocks is stupid contrasted with putting resources into the whole market and holding for significant lots of time.

References:

1. https://wall-street.com/the-pros-cons-of-cryptocurrency/
2. https://www.trustetc.com/blog/September-2018/types-of-cryptocurrency
3. https://en.wikipedia.org/wiki/List_of_cryptocurrencies
4. https://www.kaggle.com/jessevent/all-crypto-currencies/home
5. https://www.nuwavesolutions.com/fact-tables/
6. Pivoting The columns using unstack()
7. https://nikgrozev.com/2015/07/01/reshaping-in-pandas-pivot-pivot-table-stack-and-unstack-explained-with-pictures/