Group8: PREDICTIVE MODELING OF BITCOIN PRICES

Waqar Sarguroh(16BM6JP40), Shivam Arora(16BM6JP43), Prakhar Gupta(16BM6JP34), Yashas Nataraj B(16BM6JP53)

Abstract:

- Of late, Bitcoin has generated tremendous interest as an alternative medium of exchange, owing to its characteristics of being Decentralized, having no single authority who controls the Network and ease of storage and use - a form of digital currency, created and held electronically. Bitcoin and other versions of Digital cryptocurrencies are slowly being adopted worldwide commercially. This has spurred research into the Behaviour of Bitcoin prices and its attendant causes and effects. In this project, we explore a couple of Bitcoin Pricing predictive models which have been expounded in the Literature namely ARIMA and Bayesian Regression

INTRODUCTION:

A wide range of virtual currencies have emerged during the last decade, such as BitCoin, LiteCoin, PeerCoin, AuroraCoin, DogeCoin and Ripple. The most successful among them is BitCoin, both in terms of its impressive growth in the number of currency users and popularity by retailers. Since its introduction in 2009, BitCoin has been characterized also by a phenomenal increase in the number of transactions and market capitalization, which surpassed 5 billion US dollar in 2015 and since then has recorded further growth.

In the public media and in scientific community there is ongoing a lively debate on wheather BitCoin can actually function as a substitute for standard currencies such as US dollar, Euro or Yen. There is no global agreement on the status of BitCoin, as there are no international laws regulating Bitcoin. Each country regards Bitcoin differently and regulations are constantly evolving

BitCoin is managed by an open source *software algorithm* that uses the global internet network both to create BitCoins as well as to record and verify its transactions. Being a decentralized cryptocurrency, BitCoin uses the principles of cryptography to control the creation and exchange of BitCoins. BitCoins can be stored in local wallets (e.g. personal computer, smartphone) using an open-source software or in an online wallet. Please refer to Figure 1 for a schematic depiction of Bitcoin's working.

Among the BitCoin features, which may facilitate its use as a currency are low transaction costs, high anonymity and privacy, learning spillover effects, infinite divisibility and no inflationary pressures. Among the BitCoin features, which may impede its use as a currency include the absence of a legal tender attribute, difficulty to procure BitCoins, relatively high fixed

costs of adoption, dependence on network externalities, absence of an institution enforcing dispute resolution, absence of BitCoin denominated credits, deflationary pressure, extremely high price volatility, and issues with cyber security.(Please refer to Figure 2)

Some of the abiding questions which linger are: 1. Is Bitcoin a currency or a speculative investment? 2. Is the Quantity Theory of Money applicable to Bitcoin? 3. What are the determinants of Bitcoin prices? 4. Can derivatives be built on Bitcoin prices? 5. Can Bitcoin prices be predicted at all? In this project we explore the last question and look at some predictive models expounded in the literature to study Bitcoin Price movements.

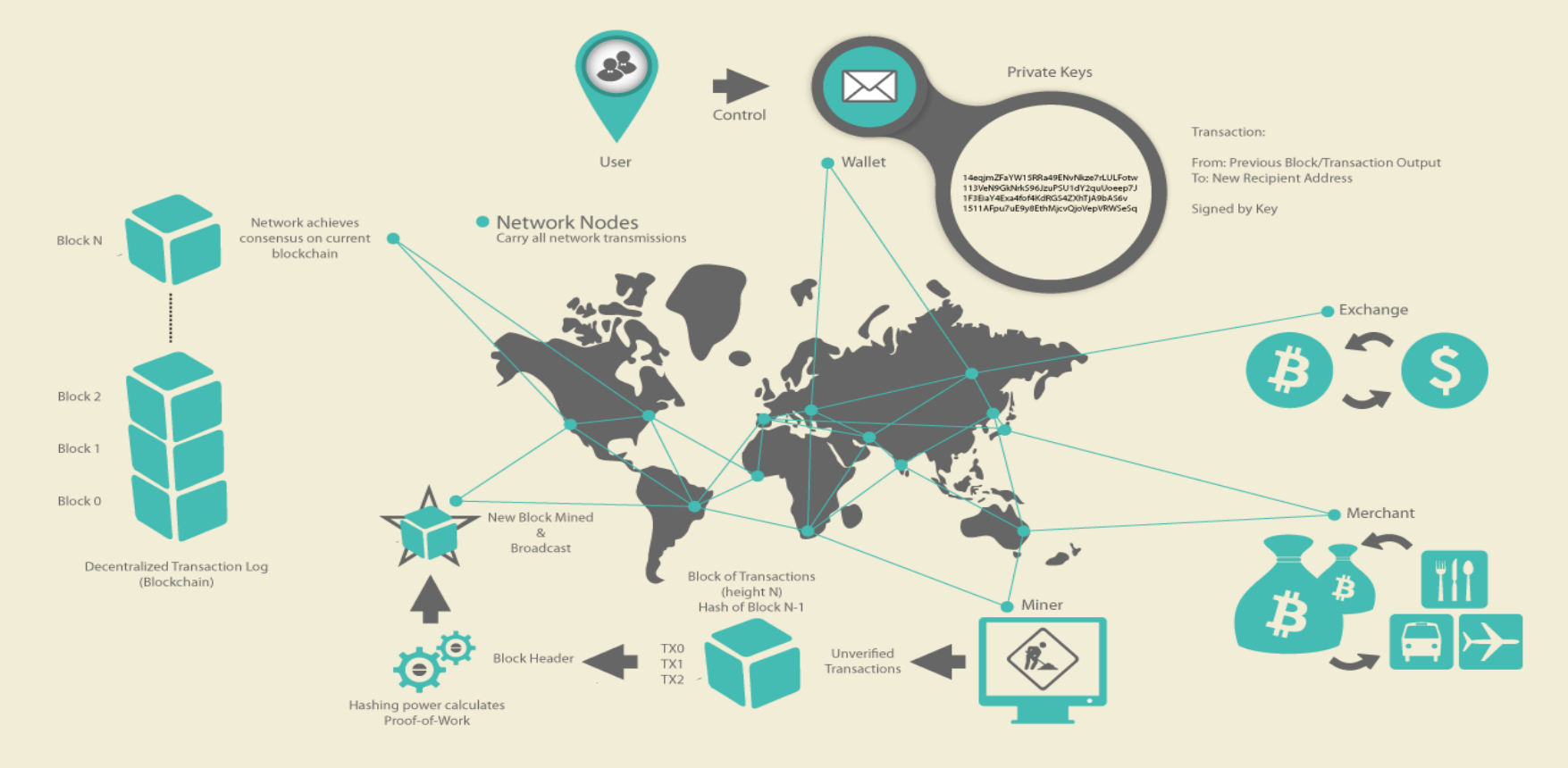


Figure 1

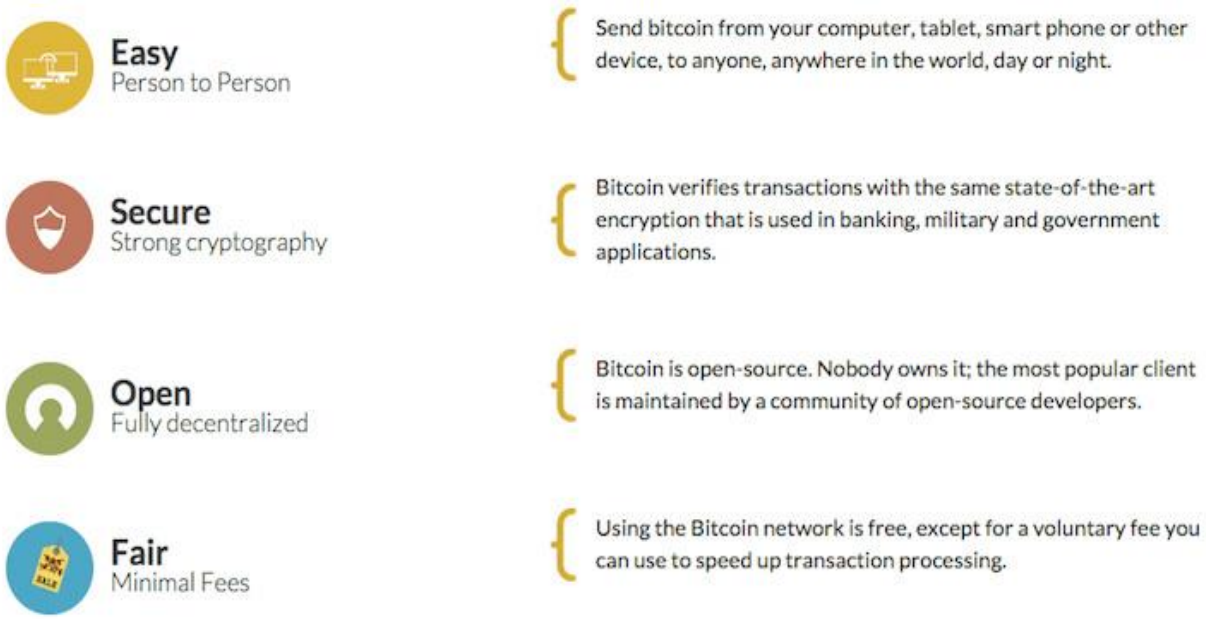
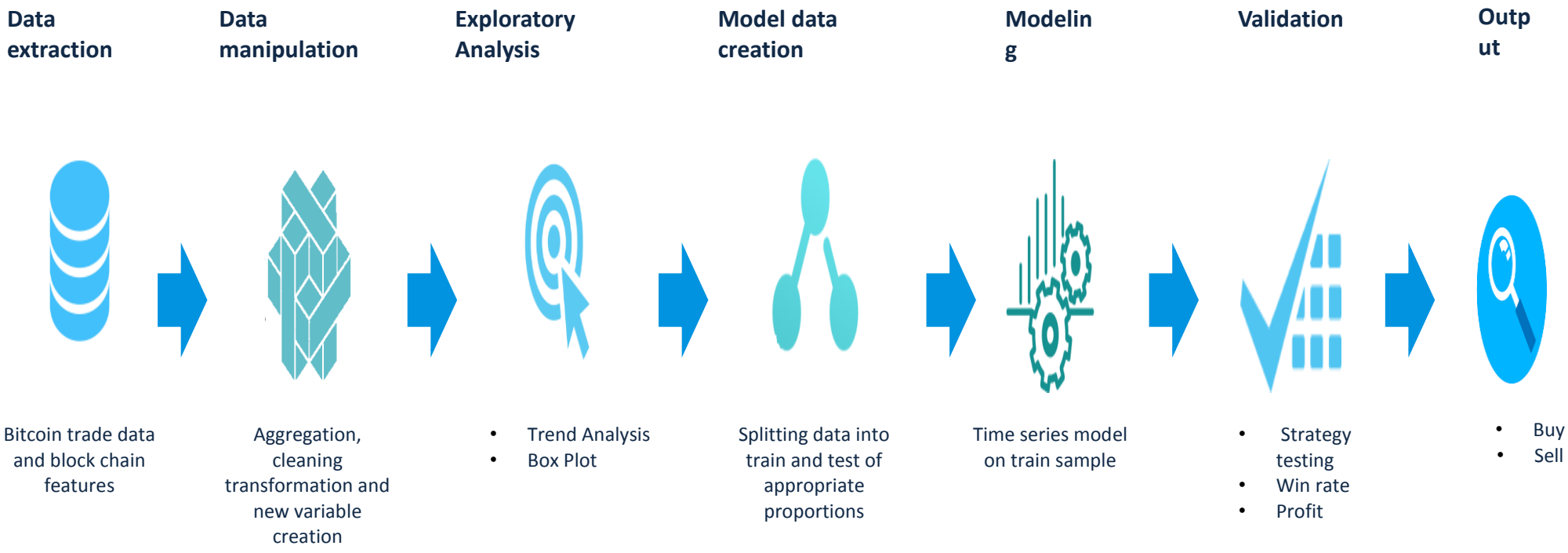


Figure 2

The subsequent sections are organized as follows: The Modeling Process section describes the overarching framework of our endeavor. The Data Extraction section elucidates on Web scraping and other Bitcoin data acquisition details. We then move into Exploratory Data Analysis , the nuances of Time series Modeling , Self-Similarity Tests and finally to Latent source models and the application of Bayesian Regression to such models.

MODELING PARADIGM

The following schematic describes the Modeling framework used:



MISSING DATA: There were missing price values for certain time instants. We used exponential smoothing techniques to smoothen the data and impute the values. We were not able to conclude whether they were Missing at Random or Missing Not at Random.

DATA SCRAPING and EXTRACTION: We wrote a script which extracted data from OKCoin and BTCcoin using their exposed REST APIs. The data were in JSON format. We later converted the data into CSV. **Trade data** was scraped through bitcoincharts.com for multiple exchanges. The **frequency** of data is **15min.** Block chain features were downloaded from blockchain.info

JSON Script


```
{"startUrl":"http://bitcoincharts.com/charts/okcoinCNY#rg30zig15-minzczsg2015-12-02zeg2016-1-01ztgMzm1g10zm2g25zv",
"selectors":[{"parentSelectors":["_root"],"type":"SelectorElementClick","multiple":false,"id":"shi","selector":"div.grid_16 div a",
"delay":"500","clickElementSelector":"div.grid_16 div a","clickElementUniquenessType":"uniqueText","clickType":"clickOnce",
"discardInitialElements":false},{"parentSelectors":["_root"],"type":"SelectorTable","multiple":true,"id":"data","selector":"table.data",
"tableHeaderRowSelector":"thead tr","tableDataRowSelector":"tbody tr","columns":[{"header":"Timestamp","name":"Timestamp","extract":true},
{"header":"Open","name":"Open","extract":true},{"header":"High","name":"High","extract":true},{"header":"Low","name":"Low","extract":true},
{"header":"Close","name":"Close","extract":true},{"header":"Volume (BTC)","name":"Volume (BTC)","extract":true},{"header":"Volume (Currency)",
"name":"Volume (Currency)","extract":true},{"header":"Weighted Price","name":"Weighted Price","extract":true}],"delay":"4000"}],"_id":"botcoins"}
```


Features Extracted

Preview of scrapped data

Timestamp	Open	High	Low	Close	Volume (BTC)	Volume (Currency)	Weighted Price
1/1/2015 0:00	1971.41	1973.5	1964.01	1967.66	1082.69	2130396.62	1967.69
1/1/2015 0:15	1967.69	1970.78	1965.38	1965.38	1421.63	2797379.07	1967.73
1/1/2015 0:30	1965.38	1966.67	1959.04	1959.94	1923.18	3775677.32	1963.25
1/1/2015 0:45	1959.94	1961.33	1957.08	1958	1887.64	3699312.03	1959.76

FEATURE EXTRACTION:

The following features were extracted from both the Trade data and the Blockchain:

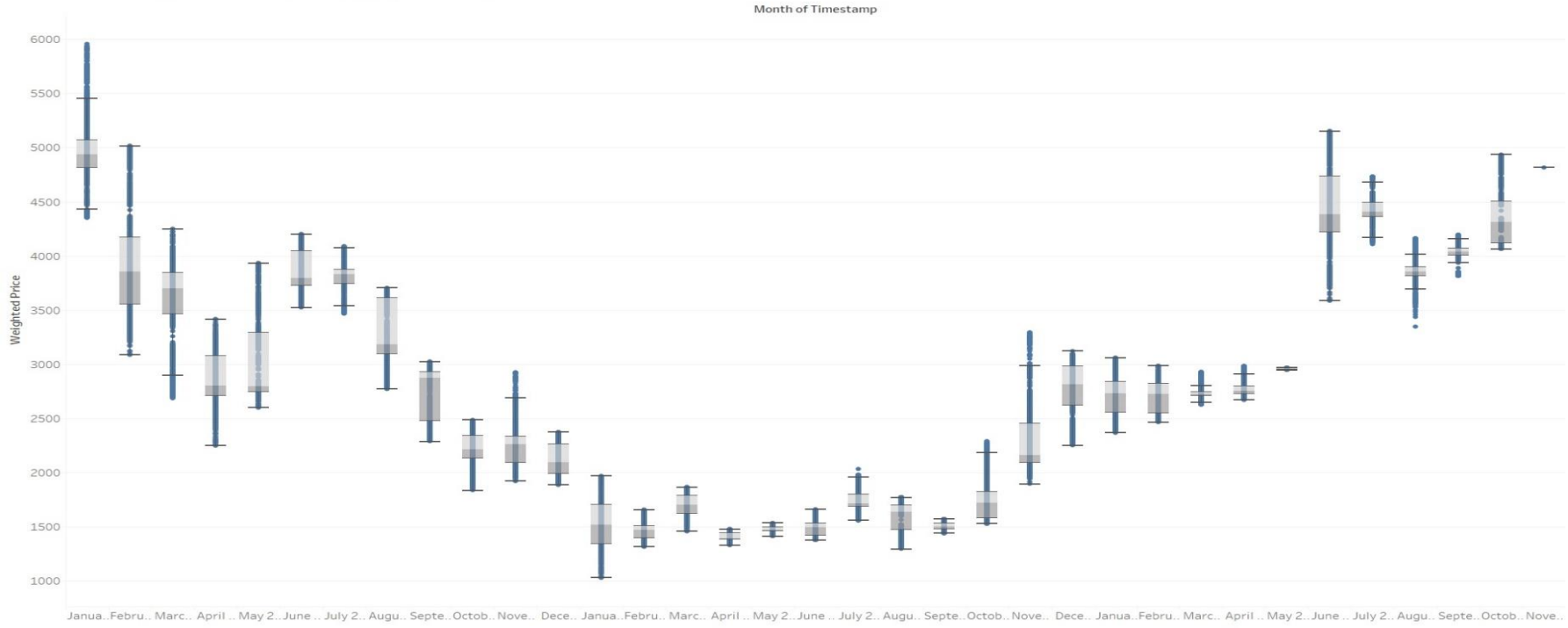
Bitcoin Trade data	
Tick data • Timestamp • **O**pen Price • **H**igh Price • **C**lose price • **L**ow Price • Volume (BTC) • Volume(Currency) • Weighted Price	• Timestamp • Avg. transaction time • Avg. block size • Cost per transaction • Difficulty • No of Transactions • Hash Rate • Market Capitalization • Miner's Revenue • No of transactions per block • Total bitcoins

EXPLORATORY DATA ANALYSIS:

Our first step was to explore the Data to understand the structure and possibly decipher the underlying processes which generated them. We undertook the following EDA methods:

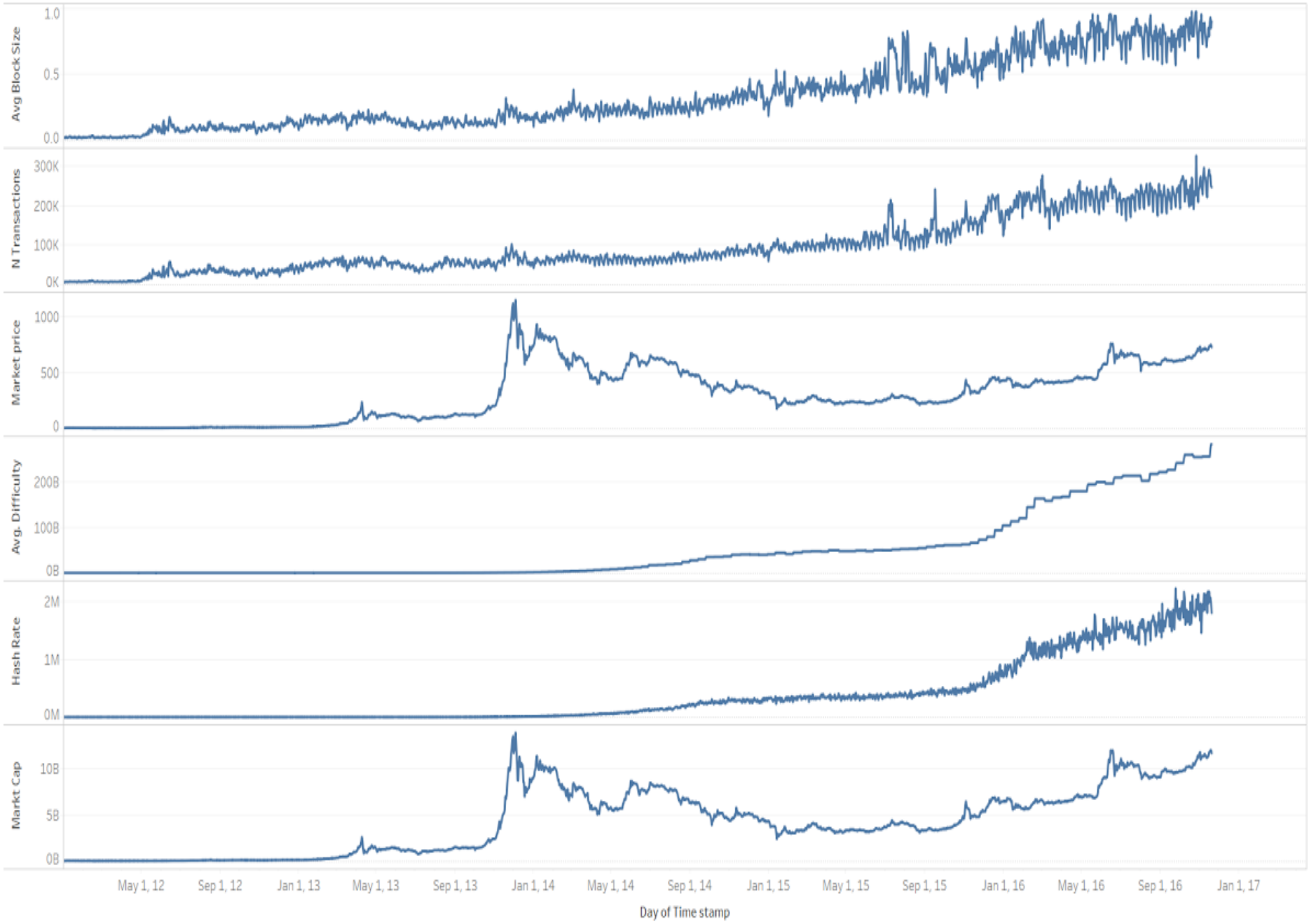
The trend of maximum of Weighted Price for Timestamp Minute.

Box Plot for Weighted price vs Month(2014-2016)



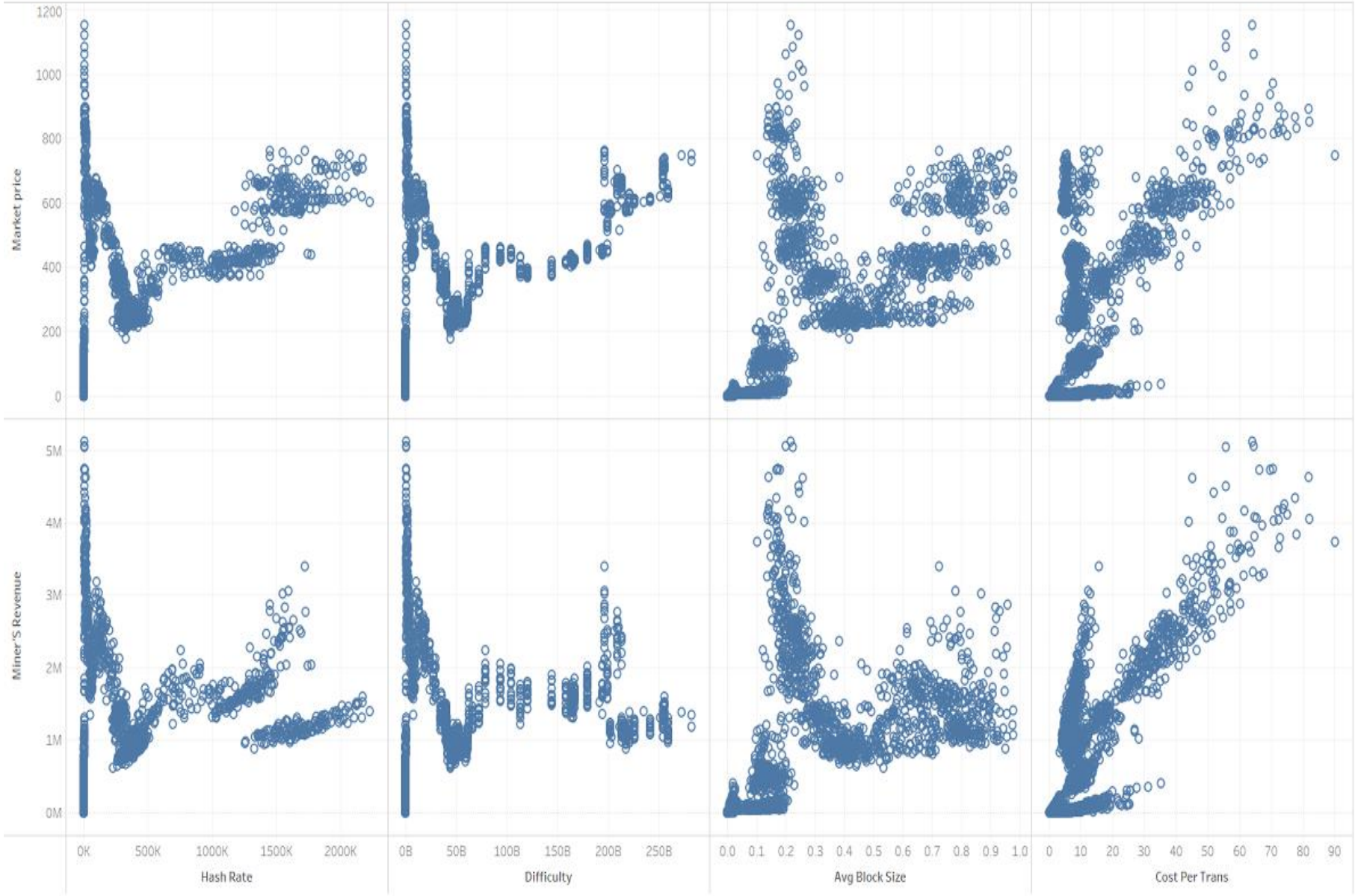
Weighted Price for each Timestamp Month. Details are shown for Timestamp Month.

Trend of Avg Block Size, Transaction, Market price, Avg. Difficulty, Hash Rate, Market Cap



The trends of Avg Block Size, N Transactions, sum of Market price, average of Difficulty, sum of Hash Rate and sum of Markt Cap for Time stamp Day.

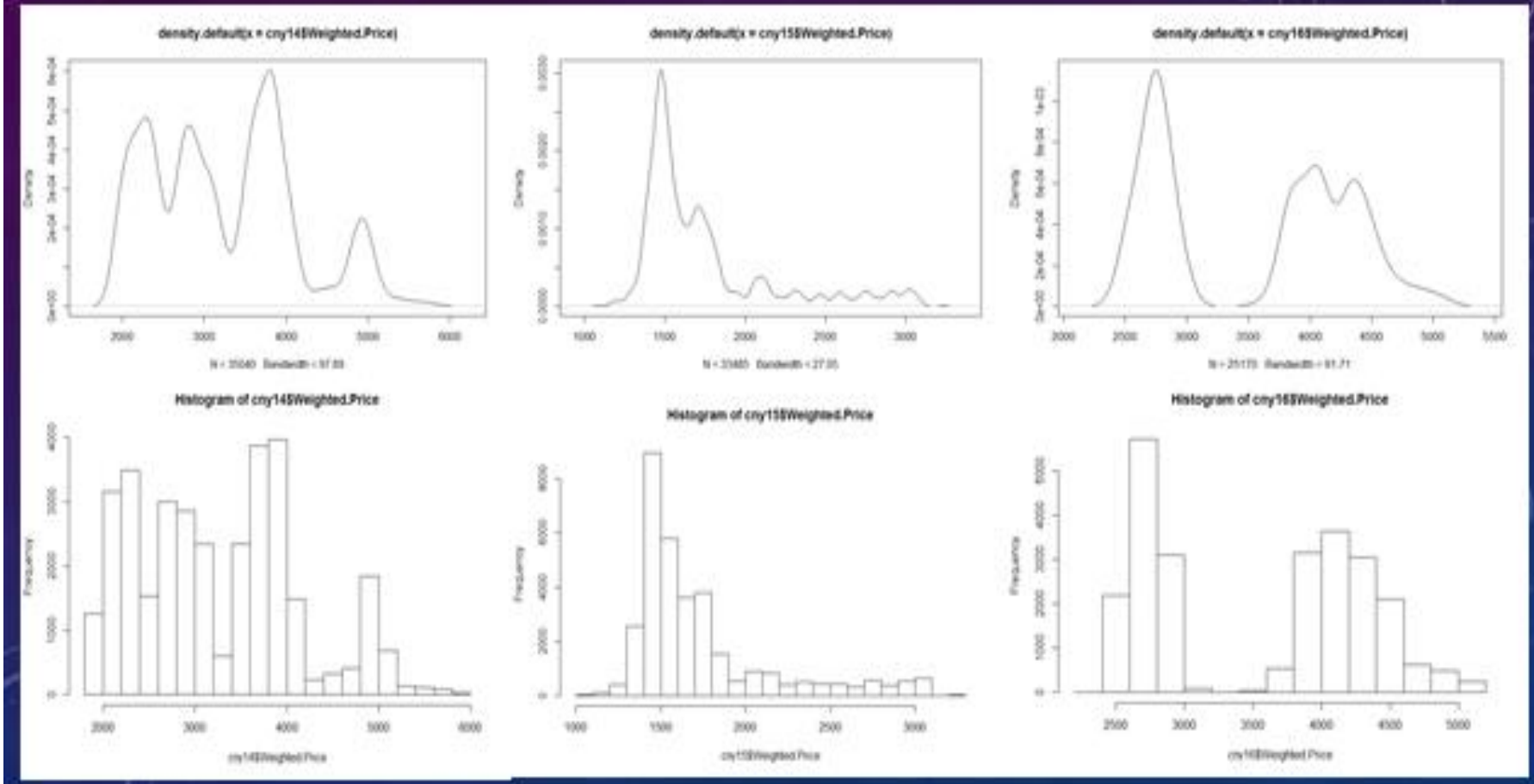
Comparative Trend



Hash Rate, Difficulty, Avg Block Size and Cost Per Trans vs. Market price and Miner'S Revenue.

DENSITY AND HISTOGRAM FOR WEIGHTED PRICE

2014 2015 2016

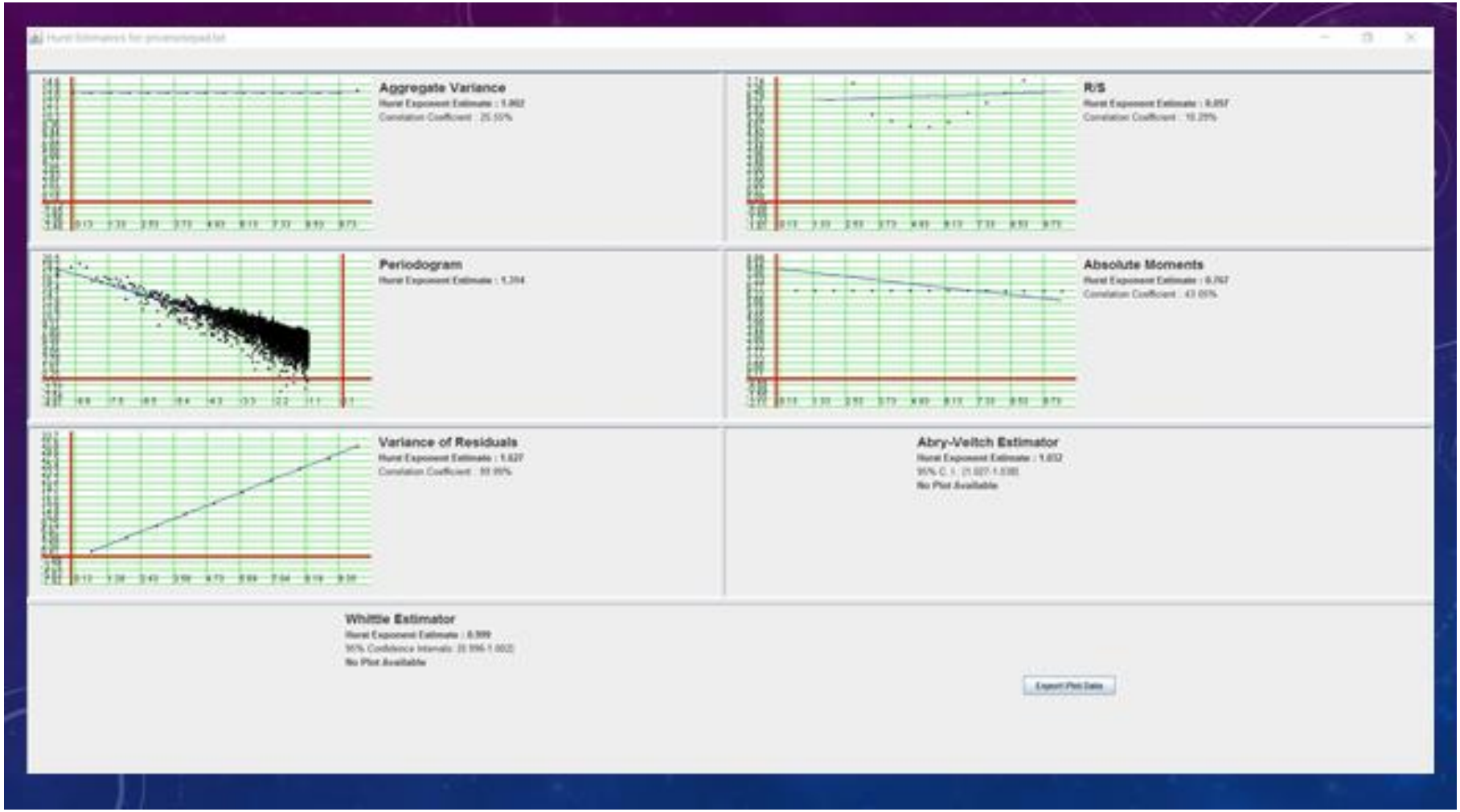


REMARKS about EDA:

a. The graphs show that there is less clarity about the Stationarity of the time series data. The propounded models regarding Classical Decomposition of time series like Seasonality, Cycles, Residuals are perhaps not applicable.
b. The volume Boxplot shows the presence of several OUTLIERS which needs to be investigated further
c. The trend plot shows that there has been an upward trend for most measures post 2011.
d. The Comparative trend plots show that the Miner's revenue and market price have a positive correlation with the Cost per transaction.Miners' revenue seemingly has a +ve correlation with the Hash rate as expected. The Market price also has a +ve correlation with the Hash rate which is also the same with the Difficulty levels.
e. More CORRELATION Plots are required.
f. The weighted price indicates the presence of Multi-modality.

SELF-SIMILARITY TESTS: We also conducted Tests for Self-Similarity. We used the SELFIS tool from University of California, Irvine for this purpose. The HURST parameter as recorded by the R/S statistic turns out to be LESS than 0.5 (0.057) which rules out Self-Similarity of Bitcoin prices. Bitcoin Prices are not Heavy- tailed.

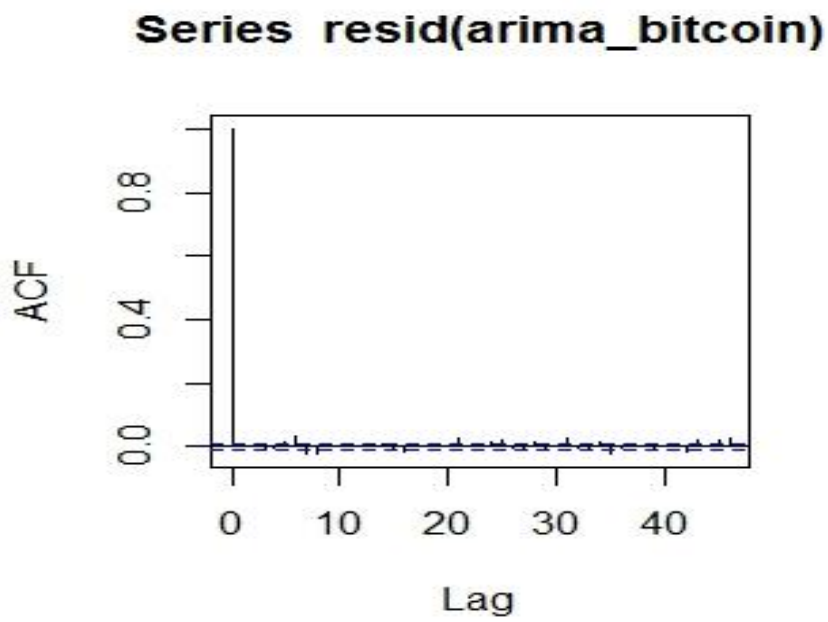
z



BITCOIN TIME-SERIES MODELING:

We started off by applying the ARIMA model in an attempt to model Bitcoin Price movements**. We ended up with a contradictory inference with the LJung-Box test hinting at dependence and the ARIMA giving us 0 values for both the AR and MA processes as illustrated below.**

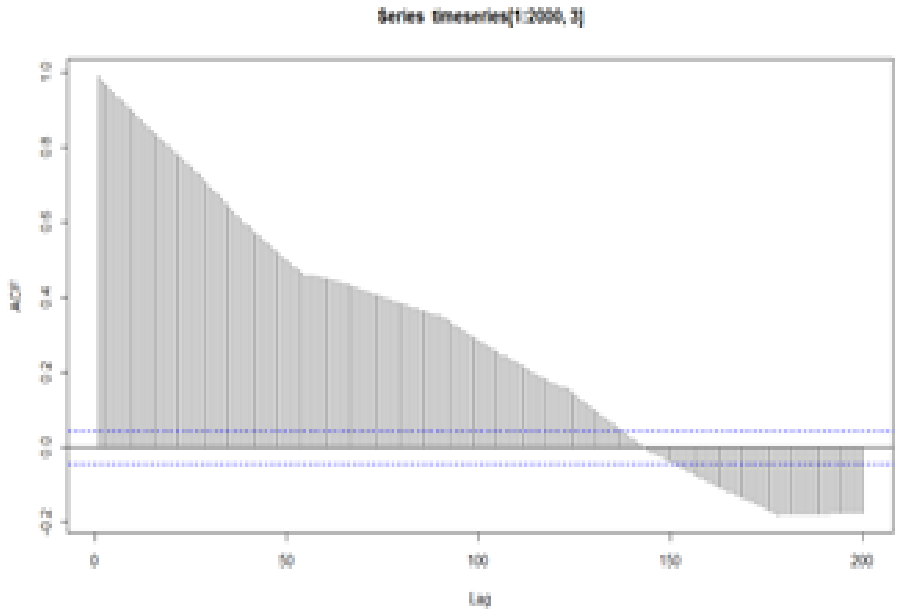
```
arima_bitcoin <- auto.arima(timeseries[1:50000, 3])

data <- timeseries[, 9]

forecastprices <- forecast.Arima(arima_bitcoin, h = 1000)

plot.forecast(forecastprices)

forecastprices$model

acf(resid(arima_bitcoin))
```


Series resid(arima_bitcoin)
ACF
0.0
0.4
0.8
0
10
20
30
40
Lag

Time Series Model : ARIMA

```
Series: timeseries[1:30000, 3]
ARIMA(0,1,0) with drift

Coefficients:
       drift
      0.0226
s.e.  0.0306

sigma^2 estimated as 46.84:  log likelihood=-167110.7
AIC=334225.5   AICc=334225.5   BIC=334243.1
```

```
Series: timeseries[1:2000, 3]
ARIMA(0,1,0) with drift

Coefficients:
       drift
     -0.0828
s.e.  0.1343

sigma^2 estimated as 36.05:  log likelihood=-6415.7
AIC=12835.41   AICc=12835.41   BIC=12846.61
```

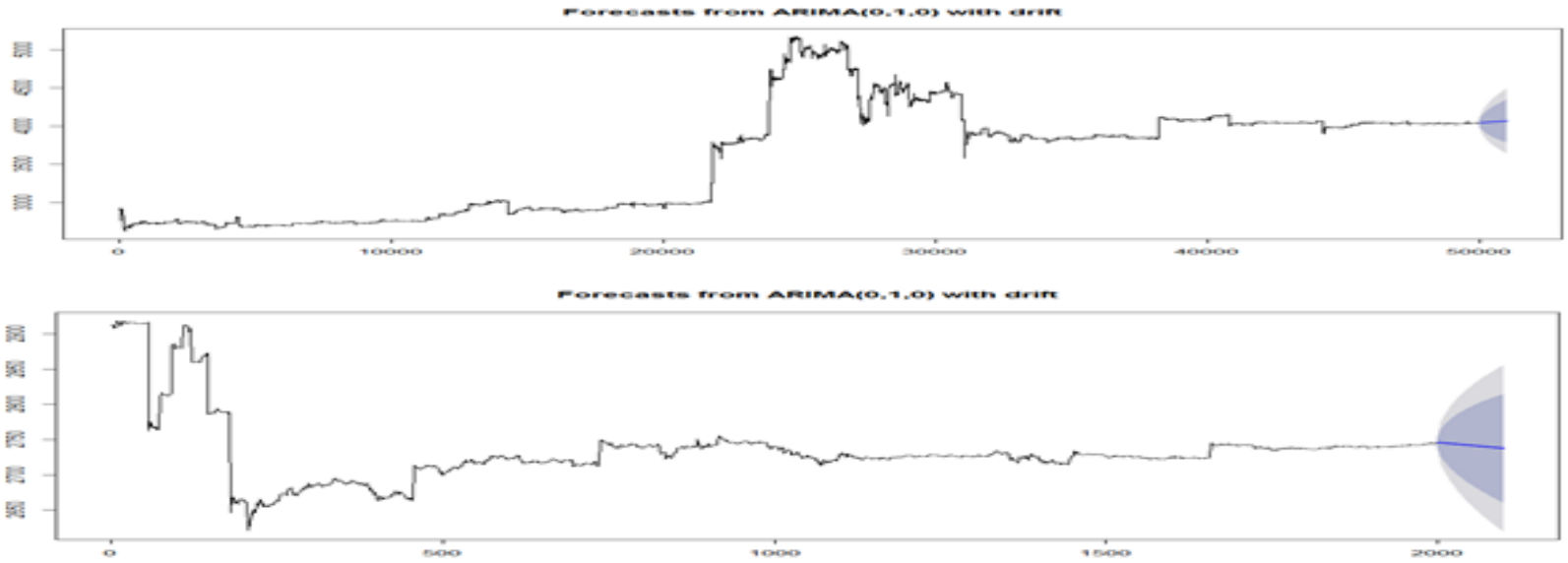


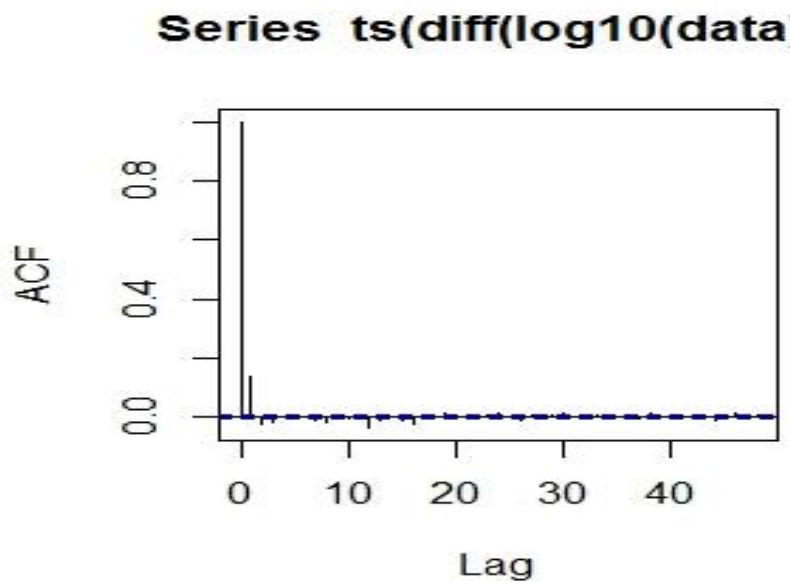
REMARKS:

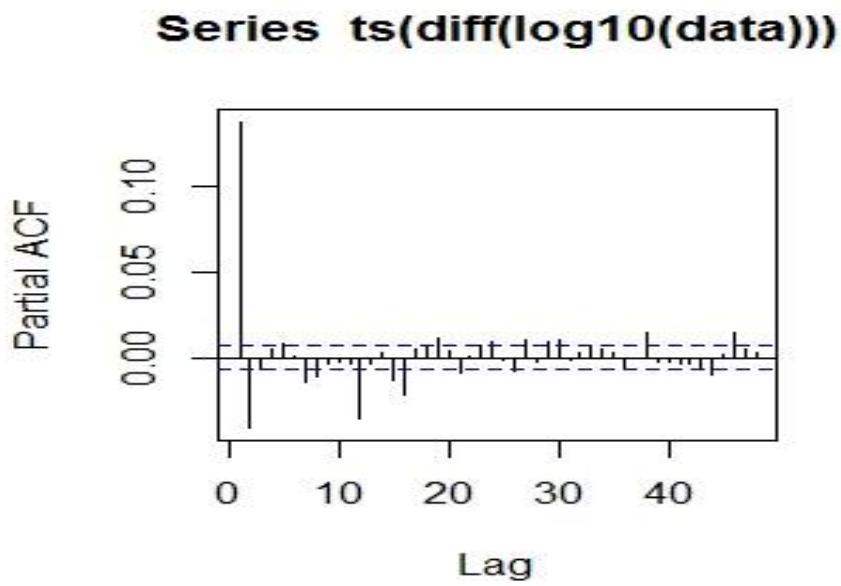
We found that the p and q parameters turn out to be 0 from which we concluded that ARIMA would not be the right model for Bitcoin. The ACF plot also showed that the Bitcoin price movements are not stationary and there is not much information in the Residuals as there are hardly any spikes going out of the safety region.

The Arima Output below which depistc the PREDICTED values for the next 1000 datapoints shows a very wide confidence interval which again undermined our confidence about ARIMA being the Best fit for Bitcoin.

ARIMA output







LJUNG-BOX Test/Box-Pierce Test: To test the hypothesis that the elements of the sequence of data in a sample are random, we aalso ran the Box-Pierce test. The Ljung-Box test is based on the autocorrelation plot. If the autocorrelations are very small, we conclude that series is random. The statistic is an average of the lags up to the stated lag.

OUTPUT:

```
Box.test(resid(arima_bitcoin),type="Ljung",lag=50,fitdf=1)
```

```
        Box-Ljung test

data:  resid(arima_bitcoin)
X-squared = 372.64, df = 49, p-value < 2.2e-16
```

Since the p-value is less than 0.05 (5% significance level), we can reject the null hypothesis of randomness.

LATENT SOURCE MODELS:

Nearest-neighbor-like methods have been widely used in practice, there is little theoretical understanding of when, why, and how well these methods work in terms of the amount of training data available and relevant structural properties in the data. Latent source models are a new addition for certain timeseries classification problems using the Nearest neighbor approach. The approach begins with a generic model that assumes very little structure and later an oracle algorithm is described by weighted plurality voting. The oracle algorithm is approximated with a nearest-neighbor-like method akin to what's used in practice.
For each case study, the training data is treated as random i.i.d. samples from an underlying probabilistic model that is intentionally chosen to be simple with few assumptions.

The approach is as follows:
The hypothesis is that in various time series classification problems, there are not many prototypical time series relative to the number of time series we have access to. For example, we suspect that news topics only go viral on Twitter in a relatively small number k of distinct ways whereas we can collect a massive number $n >> k$ of Twitter time series corresponding to different news topics. To operationalize this hypothesis, a latent source model for time series is proposed, where there are k unknown prototypical time series referred to as latent sources, each of which has label "viral" or "not viral". A new time series is generated by randomly choosing one of these latent sources, adding noise, and then introducing a random time shift. The true unobserved label for the time series is the same as that of whichever latent source the time series is generated from. The goal is to infer what this label is, given the time series observed at time steps $1; 2; : : : ; T$.
The Latent Source Model is attempts to to model existence of underlying patterns leading to price variation. Trying to develop patterns with the help of a human expert or trying to identify patterns explicitly in the data, can be challenging and to some extent subjective. Instead, using Bayesian regression approach as outlined above allows us to utilize the existence of patterns for the purpose of better prediction without explicitly finding them.

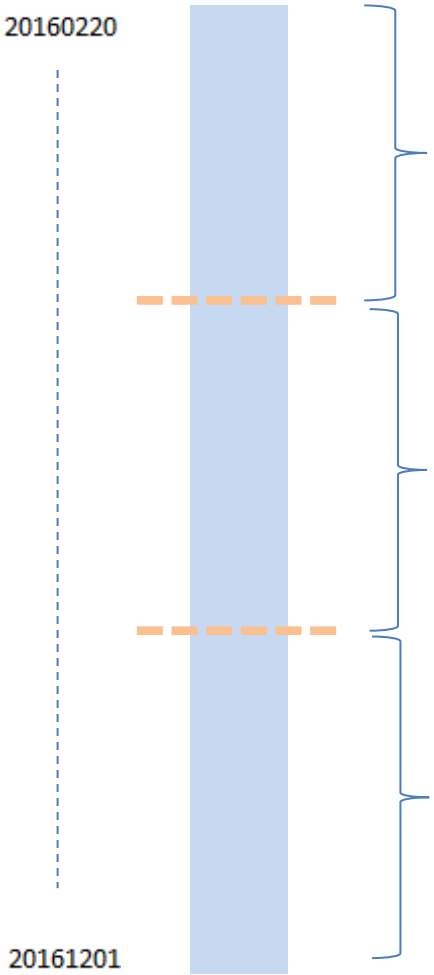
BAYESIAN REGRESSIONFOR LATENT SOURCE MODELS FOR PRICE PREDICTION

APPROACH: The basic idea here is to predict the price changes(the delta) for some t in the future using trends of past time windows of t1,t2 and t3 durations. From the training data, the time series is clustered and trends are obtained. To predict the future price changes, say at time t, 3 windows of length t-30,t-60 and t-120 are taken and compared to the training data trend clusters using a suitable similarity measure which is used to bucket the price change at t to one of the clusters and its sign deduced. The Regression also involves the Average Bid/Ask volume for the last 60 time series slots – an idea which is well taken as the Bid/Ask volume is reflective of the net Demand-Supply for Bitcoin which should ideally be one of the primary

determinants of Bitcoin Prices. This approach basically combines a Nearest Neighbour Approach with Bayesian Regression on a Latent Source Model to predict Prices

The following steps are followed chronologically in the Bayesian regression approach which elucidate the procedure in greater detail:

a. MODEL DATA CREATION:
   Data is partitioned into three parts of 20000 price points each.



- First part is used for clustering. Clustering is done thrice on windows of three sizes, 360 min (t1), 720 minutes(t2), 1440 minutes(t3)

Second part is used to train a linear model (Lasso) for

predicting price change

Third part is used to test the model.

b. Clustering is done by moving window of size τ_1, τ_2, τ_3 minutes through first 20000 data points to form three data matrices – S_1, S_2, S_3.

$$S_1 = \begin{pmatrix} p_{t1} & p_{t2} & p_{t3} & p_{t4} & \cdots & p_{t360} & \Delta p_1 \\ p_{t2} & p_{t3} & p_{t4} & \cdots & p_{t360} & p_{t361} & \Delta p_2 \\ . & & & & & & \\ . & & & & & & \\ . & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & \cdots & p_{t20000} & \Delta p_{20000} \end{pmatrix}$$

Three different sets of 100 clusters are formed and 20 'best' clusters are taken from each of three sets using maximum entropy criterion

Second set of data is used to calculate expected Δp at each of 20000 points and probabilities of the expected change in prices is calculated using:

$$P(\Delta p|Trend) = \sum_i P(\Delta p|Trend, Trend \in Cluster_i) P(Trend \in Cluster_i|Trend)$$

For each τ there are corresponding clusters.

$\Delta p_{\tau 1}$, $\Delta p_{\tau 2}$, $\Delta p_{\tau 3}$ is calculated at each of 20000 points using

$$E[\Delta p|x] = \frac{\sum_{i=1}^{n} \Delta p_i \, exp\left(-\frac{1}{4}||x - c_i||_2^2\right)}{\sum_{i=1}^{n} exp\left(-\frac{1}{4}||x - c_i||_2^2\right)}|$$

c. For $\Delta p_{\tau 1}$ at a certain time t, x in the formula is the vector of prices at 360 previous time intervals C_i is the vector of prices of the centroid in cluster i and Δp_i is the value of price change given C_i using the above formula

d. $\Delta p_{\tau 1}, \Delta p_{\tau 2}, \Delta p_{\tau 3}$ are used as explanatory variables to calculate Δp at each time step in the regression model

$$[\Delta p_{\tau 1}, \Delta p_{\tau 2}, \Delta p_{\tau 3}] = \begin{pmatrix} \Delta p_{\tau 11} & \Delta p_{\tau 12} & \Delta p_{\tau 13} \\ \Delta p_{\tau 21} & \Delta p_{\tau 22} & \Delta p_{\tau 23} \\ . & & \\ . & & \\ . & & \\ & & \\ & & \\ & \dots & \Delta p_{\tau n,3} \end{pmatrix}$$

e. **Linear Regression (via Lasso) for PREDICTION:**
The final price change predictions are then calculated using a Linear Regression Model as follows

- The final set of data points is used to test the model
- At each data point $\Delta p_{\tau 1}, \Delta p_{\tau 2}, \Delta p_{\tau 3}$ using the Bayesian formula

- This is now fed to the learned Lasso regression model to predict Δp
  $\Delta p = w_0 + w_1\Delta p_{\tau 1} + w_2\Delta p_{\tau 2} + w_3\Delta p_{\tau 3}$

  This model is trained on using actual Δp at each time point of 20000 using Lasso
  The POINT to NOTE is that we have used REGULARIZED MODELS for the Regression part – both Ridge and Lasso.
  The Paper only mentions Linear Regression.

f. **FINALLY, The following TRADING STRATEGY is used**
   Start with position=0
   Buy when predicted Δp > threshold and position = 0
   Sell when predicted Δp < -threshold and position = 1
   Hit rate = (#times a bitcoin is bought and actual price increases)/(#times a bitcoin is bought)
   Total earnings is earned money calculated at each sell step from actual price change

IMPLEMENTATION:

The Bayesian regression Model was implemented using MATLAB. The model was tested with different similarity measures like Euclidean and Cosine. We also experimented with Lasso and Ridge for Linear Regression for Prediction purposes. We used the Data from OKCoin for the year 2015 scraped at a granularity of 15 seconds amounting to 77055 data points. A snapshot of the data is given below:

Timestamp	Open	High	Low	Close	Volume..BTC.	Volume..Currency.	Weighted.Price

RESULTS:

a. Lasso regression:

Lasso Coefficients

-0.0079
-0.0052
0.0015
0

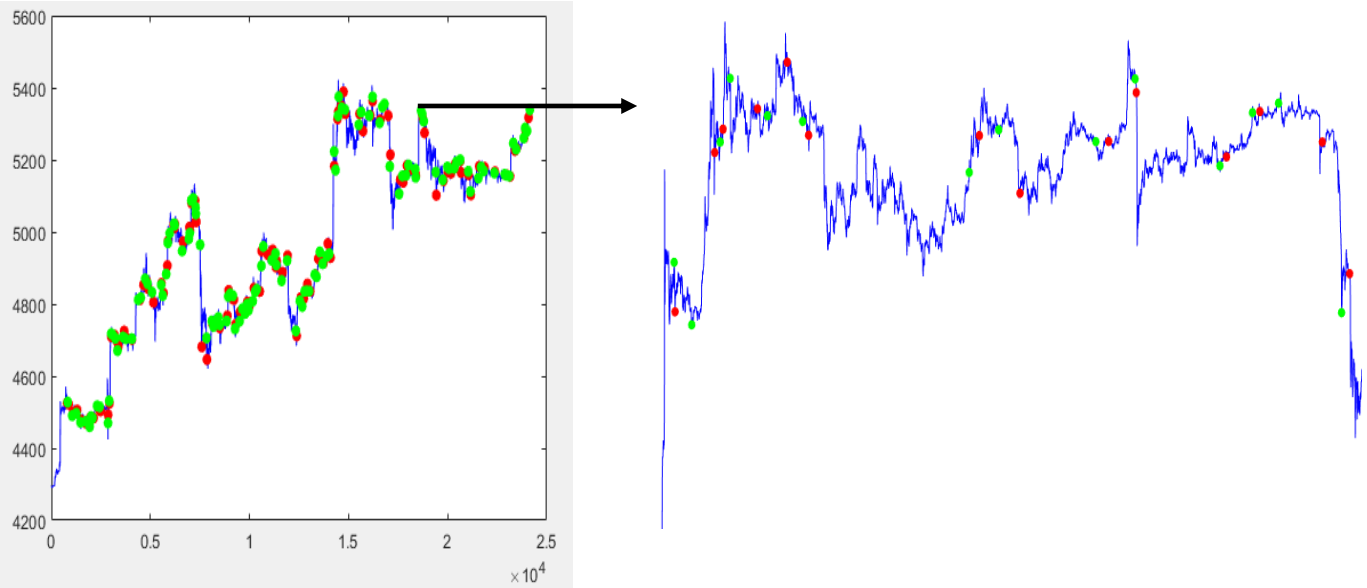
Fit Information

Intercept	0.0202
Lambda	1.0000e-03
Alpha	1
DF	3
MSE	2.6990

b. Trading strategy: The Win rate is a measure of Predictive accuracy which is 58%. As the number of windows are increased from 3, the win rate sometimes increases and decreases during other instances. The Win rate also depends on the granularity of prediction. More granular predictions lead to more trades. There seems to be a trade-off between the Win rate and Total Profit garnered which needs to be explored further.

```
Win rate: 5.819672e+01 percent
Total profit: 8.638600e+02
```

c. Predicted Vs. Actual prices (Red dots – Buy, Blue dots – Sell)



CONCLUSION and FUTURE WORK:

- This was a preliminary study undertaken and needless to say has to be expanded further.

  We faced several challenges like the following:

  The data for different bitcoin exchanges is not readily available.

  - Limited literature on application of machine learning for predicting bitcoin price.
  - Limited granularity for data sets

We intend to explore further on the following lines:

1. Need to look at post Demonetisation data to decipher patterns
2. Need to explore the Currency Vs. Stock debate
3. Apply Time series Factor Analysis Models
4. Regression using other Explanatory variables(features extracted from the Block Chain).
5. Using Bayesian regression on Latent Source Models for Stocks and other instruments.
6. Errors are assumed to be Normally distributed. What if they are not? Try extreme value distributions!
7. Most importantly, need to look at the contradictory conclusions of the Box-Pierce test and the ARIMA inferences.
8. Attempt to apply the Quantity Theory of Money, Brownian motion tests and other Macro-Economic factors.
9. Granular Bid/Ask Ratio, Graph modeling

REFERENCES:

- Bayesian regression and bitcoin by Devavrat Shah, Kang Zhang
- Economics of bitcoin price formation.