CHAPTER 1
INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 What actually is our project?

Our project titled “Prediction And Analysis of Trends in Cryptocurrency Market” applies Exploratory Data Analysis on a set of data of historical prices of Cryptocurrencies to find answers to some fascinating questions.

This project aims at applying Data Science and Machine Learning in order to analyse and predict the nature of Cryptocurrency Market. This analysis and prediction should be such that it can be easily used by the users in order to exploit the Cryptocurrency Market and thus making rational decisions in choosing the right Cryptocurrency to invest in.

1.2 Objective of the Project

With Data Analysis, a large set of data can be analyzed and relations between different variables in the data set can be obtained. With Regression, a technique of Machine Learning, trends in data can be found. These trends can then be used to predict the nature of the variables based on the already present data.

This project aims at applying Data Science and Machine Learning to find answers to a number of questions related to the Cryptocurrency Market. This is done by applying Exploratory Data Analysis to draw graphs and look at the possible relations that can lead to these answers. This should be done in such a way that it enables other users to exploit the Cryptocurrency Market and thus making rational decisions in choosing the right Cryptocurrency to invest in.

Exploratory Data Analysis, commonly known as EDA, is used in this project which makes use of visualization techniques to clean and understand the data. EDA makes the data neat to be able to be used for Machine Learning.

By analyzing these graphs, the main aim is to find the trends which the changes in the prices of different Cryptocurrencies follow. Also, we aim to find relationships between the price variations of different Cryptocurrencies if they exist.

Lastly, this project aims at predicting the prices of different Cryptocurrencies based on the changes in their prices in the past. Once these features are implemented in entirety, the project aims to be good and simple enough to be used by a naïve user by reading just a simple set of instructions. This benefit should be global and thus any user anywhere in the world should be able to use this. In order to achieve this, all the functions used in the project will be shipped to a website where users can go and use them for according to their needs.

1.3 Scope of the Project

People usually have hard time deciding which Cryptocurrency to invest in. Since their prices vary a lot and most of the users find this variation unpredictable, they end up investing either in the wrong cryptocurrency, investing a wrong amount or investing at the wrong time leading to a loss to them. Analysis means detailed examination of the elements or structure of something. If anything is analysed properly, it can be understood clearly, and thus right decisions can be taken involving those things. This project is designed with the motive of helping users take rational decisions about investing their money in Cryptocurrency market. Since, the project provides an insight into how the prices of different Cryptocurrencies varied with time as well as providing predictions for fluctuations in their prices in the future, it will help users in understanding the Cryptocurrency market better. Moreover, in order to make everything user - friendly for the users, a prompt would be provided to them that would have different actions from which the user would be able to select the desired one. Using this, the user can see the analysis about any particular cryptocurrency that he/she is interested in, can compare two or more cryptocurrencies to decide which would be a better investment option and even predict their prices in future in order to avoid a loss.

The functions used in the project will be uploaded to a website in the form of Shiny App. This will help a user sitting in any part of the world to import these packages to perform various analysis and predictions on the Cryptocurrency Market in a seamless handy way.

CHAPTER 2
LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

2.1 What is Data?

In general, data is any set of characters that has been gathered and translated for some purpose, usually analysis. It can be any character, including text and numbers, pictures, sound, or video. If data is not put into context, it doesn't do anything to a human or computer.

Within a computer's storage, data is a collection of numbers represented as bytes that are in turn composed of bits (binary digits) that can have the value one or zero. Data is processed by the CPU, which uses logical operations to produce new data (output) from source data (input).

Data can also be seen as a set of values of *qualitative* or *quantitative* variables.

A *qualitative variable*, also called a *categorical variable*, are variables that are not numerical. It describes data that fits into categories. For example:

- Eye colors (variables include: blue, green, brown, and hazel).
- States (variables include: Florida, New Jersey, and Washington).
- Dog breeds (variables include: Alaskan Malamute, German Shepherd)

These are all qualitative variables as they have no natural order. On the other hand, *quantitative* variables have a value and they can be added, subtracted, divided or multiplied. As the name suggests, they represent some quantity. Examples of quantitative variables are *height, age, crop yield, GPA, salary, temperature, area, air pollution index (measured in parts per million), etc.*

2.2 Data Analysis

According to Wikipedia, Data Analysis is a process of inspecting, cleansing, transforming, and modelling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making. Analysis refers to breaking a whole into its separate components for individual examination. Data analysis is a process for obtaining raw data and converting it into information useful for decision-making by users. Data is collected and analysed to answer questions, test hypotheses or disprove theories. Data Analysis has become quite popular in the present days because of the large amount of data being generated every day. According to computer giant IBM, 2.5 Exabyte - that's 2.5 billion gigabytes

of data was generated every day in 2012. This much data provides a great scope for Data Analysis.

2.3 How is Data Analysis Performed?

Data Analysis is a part of a larger process of deriving business intelligence. The process includes one or more of the following steps:

- **Defining Objectives:** Any study must begin with a set of clearly defined business objectives. Much of the decisions made in the rest of the process depends on how clearly the objectives of the study have been stated.
- **Posing Questions:** An attempt is made to ask a question in the problem domain. For example, do red sports cars get into accidents more often than others?
- **Data Collection:** An attempt is made to ask a question in the problem domain. For example, do red sports cars get into accidents more often than others?
- **Data Wrangling**: An attempt is made to ask a question in the problem domain. For example, do red sports cars get into accidents more often than others?
- **Data Analysis:** This is the step where the cleaned and aggregated data is imported into analysis tools. These tools allow you to explore the data, find patterns in it, and ask and answer what-if questions. This is the process by which sense is made of data gathered in research by proper application of statistical methods.
- **Drawing Conclusions and Making Predictions:** This is the step where, after sufficient analysis, conclusions can drawn from the data and appropriate predictions can be made. These conclusions and predications may then be summarized in a report delivered to end-users.

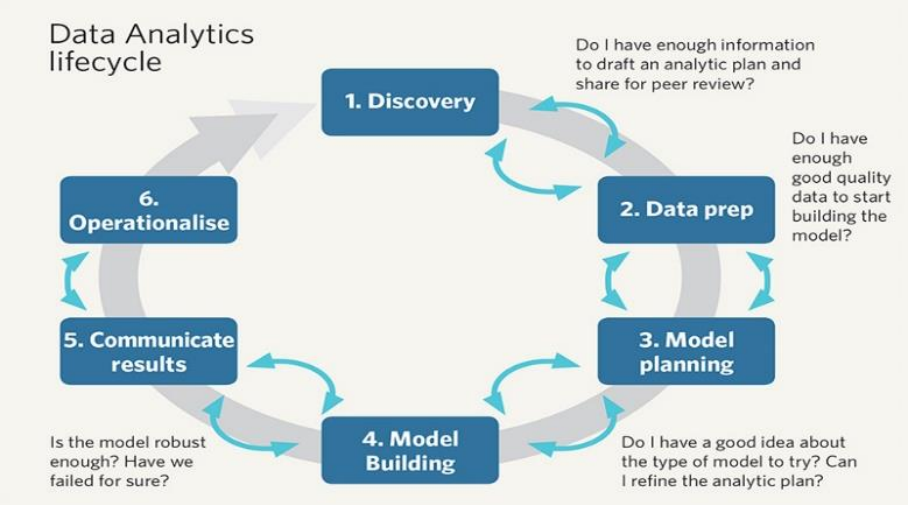

Figure 2.1 : Steps of a Data Analysis Process

2.4 Why does one need Data Analysis?

Data Analytics is needed in Business to Consumer applications (B2C). Organisations collect data that they have gathered from customers, businesses, economy and practical experience. Data is then processed after gathering and is categorised as per the requirement and analysis is done to study purchase patterns and etc.

The idea is to make sense of the data one has, to analyze it and share better business prospects in the near future and how one is going to do it, is with the concepts of analytics. It is the science of extracting trends, patterns and useful information from a set of existing data which will be of no use if not analyzed. It is a kind of business intelligence that is now used for gaining profits and making better use of resources. This can also help in improving managerial operations and leverage organizations to next level.

If not analyzed this data is going to get wasted whereas if analyzed properly this data can help us in finding information that is powerful to bring in a change in the patterns of how business is already working or going. Just imagine a source of unleashed information exists and you haven't dived in yet to get the grip of it. A business can take a competitive advantage of it and do wonders with the data. This is going to extract insights that will allow an advantage to a business or an organization in an economy.

Data and information are increasing rapidly, the growth rate of the information is so high that the information available to us in the near future is going to be unpredictable. Data is generated through hundreds of users, businesses and industries on a whole.

Modelling and visualizing is one of the major aspects of analytics and so to get an up gear from this, you really need to understand the intricacies of it as a whole. Earlier data needed a number of skilled analysts to process data whereas we now have tools that are used in running high-speed data analytics on massive amounts of data, and this gives an opportunity to the entrepreneurs to incorporate data analytics when making decisions.

Different decisions can be made as far as your target audience is concerned, your audience can change on the basis of the analysis you have done with the help of data analytics. Social media is another example that has increased the growth of the data and your organization can make changes based on that too. As the communication between you and consumer if analyzed can also help in making snap decisions.

2.5 Types of Data Analysis

Though based on the way of categorization, there can be many kinds of Data Analysis, they are mainly divided into six categories:

- **Descriptive Analysis:** Descriptive Analysis is the discipline of quantitatively describing the main features of a collection of data. In essence, it describes a set of data.
  - Typically the first kind of data analysis performed on a data set.
  - Commonly applied to large volumes of data, such as census data.
  - The description and interpretation processes are different steps.
  - Univariate and Bivariate are two types of statistical descriptive analyses.
  - *Type of data set applied to:* Census Data Set – a whole population.

- **Exploratory Analysis:** An approach to analyzing data sets to find previously unknown relationships.
  - Exploratory models are good for discovering new connections.
  - They are also useful for defining future studies/questions.
  - Exploratory analyses are usually not the definitive answer to the question at hand, but only the start.
  - Exploratory analyses alone should not be used for generalizing and/or predicting
  - Remember: correlation does not imply causation
  - *Type of data set applied to:* Census and Convenience Sample Data Set (typically non-uniform) – a random sample with many variables measured.

- **Inferential Analysis:** Aims to test theories about the nature of the world in general (or some part of it) based on samples of "subjects" taken from the world (or some part of it). That is, use a relatively small sample of data to say something about a bigger population.
  - Inference is commonly the goal of statistical models.

- Inference involves estimating both the quantity you care about and your uncertainty about your estimate.

- Inference depends heavily on both the population and the sampling scheme.

- Type of data set applied to: Observational, Cross Sectional Time Study, and Retrospective Data set – the right, randomly sampled population.

- **Predictive Analysis:** The various types of methods that analyze current and historical facts to make predictions about future events. In essence, to use the data on some objects to predict values for another object.
  - The models predicts, but it does not mean that the independent variables cause.
  - Accurate prediction depends heavily on measuring the right variables.
  - Although there are better and worse prediction models, more data and a simple model works really well.
  - Prediction is very hard, especially about the future references.
  - *Type of data set applied to:* Prediction Study Data Set – a training and test data set from the same population.
- **Casual Analysis:** To find out what happens to one variable when you change another.
  - Implementation usually requires randomized studies.
  - There are approaches to inferring causation in non-randomized studies.
  - Causal models are said to be the "gold standard" for data analysis.
  - Type of data set applied to: Randomized Trial Data Set – data from a randomized study.

2.6 Exploratory Data Analysis

We are going to use Exploratory Data Analysis in this project on our Cryptocurrency Data. Exploratory Data Analysis (EDA) is an approach/philosophy for data analysis that employs a variety of techniques (mostly graphical) to maximize insight into a data set, uncover underlying structure, extract important variables, detect outliers and anomalies, test underlying assumptions, develop parsimonious models; and determine optimal factor settings.

The primary goal of EDA is to maximize the analyst's insight into a data set and into the underlying structure of a data set, while providing all of the specific items that an analyst would want to extract from a data set, such a good - fitting, parsimonious model, a list of outliers, a sense of robustness of conclusions, estimates for parameters, uncertainties for those estimates, a ranked list of important factors, conclusions as to whether individual factors are statistically significant and a lot more.

Developed by John Tukey in the 1970s, exploratory data analysis is often described as a philosophy, and there are no hard-and-fast rules for how you approach it. That said, it also gave rise to a whole family of statistical-computing environments both used to help define what EDA is and to tackle specific tasks such as:

- Spotting mistakes and missing data.
- Mapping out the underlying structure of the data.
- Identifying the most important variables.
- Listing anomalies and outliers.
- Testing a hypotheses / checking assumptions related to a specific model.
- Establishing a parsimonious model (one that can be used to explain the data with minimal predictor variables).
- Estimating parameters and figuring out the associated confidence intervals or margins of error.

2.7 Tools and Techniques Used in EDA

Among the most important statistical programming packages used to conduct exploratory data analysis are S-Plus and R. The latter is a powerful, versatile, open-source programming language that can be integrated with many BI platforms.

Specific statistical functions and techniques you can perform with these tools include:

- Clustering and dimension reduction techniques, which help you to create graphical displays of high-dimensional data containing many variables.

- Univariate visualization of each field in the raw dataset, with summary statistics.
- Bivariate visualizations and summary statistics that allow you to assess the relationship between each variable in the dataset and the target variable you're looking at.
- Multivariate visualizations for mapping and understanding interactions between different fields in the data.
- K-means clustering, creating "centers" for each cluster based on the nearest mean.
- Predictive models, for example, linear regression.

2.8 The Role of Graphics

Statistics and data analysis procedures can broadly be split into two parts:

- quantitative
- graphical

Quantitative techniques are the set of statistical procedures that yield numeric or tabular output. Examples of quantitative techniques include:

- hypothesis testing
- analysis of variance
- point estimates and confidence intervals
- least squares regression

On the other hand, there is a large collection of statistical tools that are generally referred to as graphical techniques. These include:

- scatter plots
- histograms
- probability plots
- residual plots
- box plots
- block plot

The EDA approach relies heavily on these and similar graphical techniques. Graphical procedures are not just tools that we could use in an EDA context, they are tools that we must

use. Such graphical tools are the shortest path to gaining insight into a data set in terms of testing assumptions, model selection, model validation, estimator selection, relationship identification, factor effect determination and outlier detection.

2.9 R Programming Language

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering,…) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

2.10 ggplot2

gplot2 is a plotting system for R, based on the grammar of graphics, which tries to take the good parts of base and lattice graphics and none of the bad parts. It takes care of many of the fiddly details that make plotting a hassle (like drawing legends) as well as providing a powerful model of graphics that makes it easy to produce complex multi-layered graphics.

ggplot2 is an implementation of Leland Wilkinson's Grammar of Graphics—a general scheme for data visualization which breaks up graphs into semantic components such as scales and layers. ggplot2 can serve as a replacement for the base graphics in R and contains a number of defaults for web and print display of common scales. Since 2005, ggplot2 has grown in use to become one of the most popular R packages.

2.11 dplyr

dplyr is an R package that is considered to be a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges:

- mutate() adds new variables that are functions of existing variables
- select() picks variables based on their names.
- filter() picks cases based on their values.
- summarise() reduces multiple values down to a single summary.
- arrange() changes the ordering of the rows.

These all combine naturally with group_by() which allows you to perform any operation “by group”. You can learn more about them in vignette("dplyr"). As well as these single-table verbs, dplyr also provides a variety of two-table verbs, which you can learn about in vignette("two-table").

dplyr is designed to abstract over how the data is stored. That means as well as working with local data frames, you can also work with remote database tables, using exactly the same R code. Install the dbplyr package then read vignette("databases", package = "dbplyr").
If you are new to dplyr, the best place to start is the data import chapter in R for data science.

2.12 plotly

Plotly for R is an interactive, browser-based charting library built on the open source JavaScript graphing library, plotly.js. It works entirely locally, through the HTML widgets framework.

By default, ggplotly() tries to replicate the static ggplot2 version exactly (before any interaction occurs), but sometimes you need greater control over the interactive behaviour. The ggplotly() function itself has some convenient "high-level" arguments, such as dynamicTicks, which tells plotly.js to dynamically recompute axes, when appropriate.

Moreover, since ggplotly() returns a plotly object, you can apply essentially any function from the R package on that object. Some useful ones include layout() (for customizing the layout), add_traces() (and its higher-level add_*() siblings, for example add_polygons(), for adding new traces/data), subplot() (for combining multiple plotly objects).

2.13 Regression Analysis

Regression is a statistical measure used in finance, investing and other disciplines that attempts to determine the strength of the relationship between one dependent variable (usually denoted by Y) and a series of other changing variables (known as independent variables). Regression helps investment and financial managers to value assets and understand the

relationships between variables, such as commodity prices and the stocks of businesses dealing in those commodities.

Most commonly, regression analysis estimates the conditional expectation of the dependent variable given the independent variables – that is, the average value of the dependent variable when the independent variables are fixed. Less commonly, the focus is on a quantile, or other location parameter of the conditional distribution of the dependent variable given the independent variables.

In all cases, a function of the independent variables called the regression function is to be estimated. In regression analysis, it is also of interest to characterize the variation of the dependent variable around the prediction of the regression function using a probability distribution.

For our project, regression can be used to predict the prices of different cryptocurrencies in future. By seeing the patterns in the fluctuations in prices of a cryptocurrency in the past, regression can be used to predict the future price of that cryptocurrency.

CHAPTER 3

SYSTEM REQUIREMENTS

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 Hardware Requirements

- An Intel-compatible platform running Windows 2000, XP/2003/Vista/7/8/2012 Server/8.1/10.
- At least 32 MB of RAM, a mouse, and enough disk space for recovered files, image files, etc.
- The administrative privileges are required to install and run R-Studio utilities under Windows 2000/XP/2003/Vista/7/8/2012 Server/8.1/10.
- A network connection for data recovering over network.

3.1 Software Requirements

- MS Windows/ MAC OS or any Linux based OS.
- R Version 3.4.2 or higher.
- R Studio Version 1.1.383 or higher.
- Sublime Text Editor.

CHAPTER 4
IMPLEMENTATION

CHAPTER 4

IMPLEMENTATION

Any Data Science project must start with some objectives, some goals that are to be achieved by analyzing the data. Cryptocurrency have a major role today for the investors. Though unknown to most people in their earlier days, cryptocurrencies now have become an attractive investment option for many people.

Still there are many questions about cryptocurrencies that remain unanswered. Which Cryptocurrencies are the best to invest in? What affects the prices of these Cryptocurrencies? Are there any small number of cryptocurrencies that dominate the cryptocurrency market or are all the Cryptocurrencies equally important? Does the change in the price of one Cryptocurrency affects the price of the other? Are they better options to invest in than other investment options?

Our project tries to find answers to all these questions by applying Exploratory Data Analysis on a set of Cryptocurrency Data.

We have a data containing data of 1320 types of Cryptocurrencies. This data contains about 65000 observations of 10 variables. These 10 variables are Date, Open, High, Low, Close, Volume, Market.Cap, coin and Delta.

The meaning of all of these variables is explained below:

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
coin	:	Name of the cryptocurrency
Volume	:	Volume of transactions on the given day
Market Cap	:	Market capitalization in USD

Added to this, we have added one extra variable, delta, to this data. Delta, for a particular day, is the difference of Open and Close divided by Open. This parameter Delta gives

us an idea of direction in which the prices of different cryptocurrencies change over a period of time. By analyzing this direction, the safeness of a Cryptocurrency can be found.

The first step was to find out the most dominant or important Cryptocurrencies in the market. So a parameter had to be chosen which could be used as a measure of this dominance. We decided to use the parameter Market.Cap.

According to Investopedia, *Market capitalization refers to the total dollar market value of a company's outstanding shares. Commonly referred to as "market cap," it is calculated by multiplying a company's shares outstanding by the current market price of one share. The investment community uses this figure to determine a company's size, as opposed to using sales or total asset.*

But this really doesn't stand true for the *Cryptocurrencies* as this definition of Market Capitalization stands true for companies.

For *Cryptocurrencies*, according to coinmarketcap.com, *Market Capitalization* is one way to rank the relative size of a cryptocurrency. It's calculated by multiplying the *Price* by the *Circulating Supply*.

Market Cap = Price X Circulating Supply

Circulating Supply is the best approximation of the number of coins that are circulating in the market and in the general public's hands. Total Supply is the total amount of coins in existence right now (minus any coins that have been verifiably burned). Max Supply is the best approximation of the maximum amount of coins that will ever exist in the lifetime of the cryptocurrency.

So to find out the most dominant Cryptocurrencies in the market, we plotted a bar graph of the Top 5 Cryptocurrencies in the market.

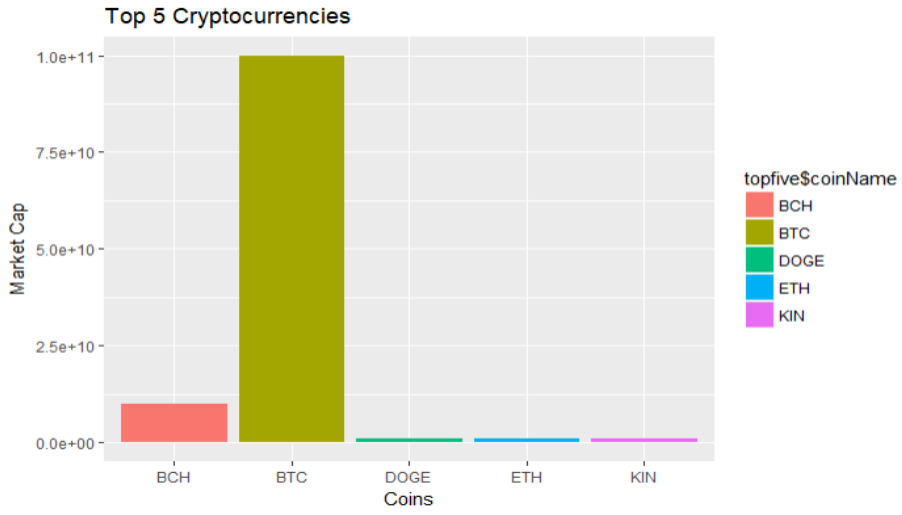

Figure 4.1: Top 5 Cryptocurrencies by Market Cap

This bar graph gives us the top 5 Cryptocurrencies by Market Cap. They are:

Coin Name	**Market Cap**
BTC(Bitcoin)	99941600000
BCH(Bitcoin Cash)	9866110000
ETH (Ethereum)	999757000
KIN	999563000
DOGE	998995000

As it is evident from the above Bar Graph, Bitcoin (BTC) alone dominates the Cryptocurrency market by a very large margin. It has a Market Capitalization of 99941600000 USD according to our data. Bitcoin Cash follows Bitcoin (BTC) in the Market Capitalization. It has a Market Capitalization of 9866110000 USD. Ethereum (Eth), KIN and DOGE follows BCH. All of these have almost same Market Capitalization. The market Capitalization of Ethereum (ETH), KIN and DOGE are 999757000 USD, 999563000 USD and 998995000 USD respectively.

So what we did after this was plotting the Time Series of all these Cryptocurrencies. A Time Series of a Cryptocurrency shows the variation in the price, in USD, of that Cryptocurrency with time. We obtained the following graph from it.

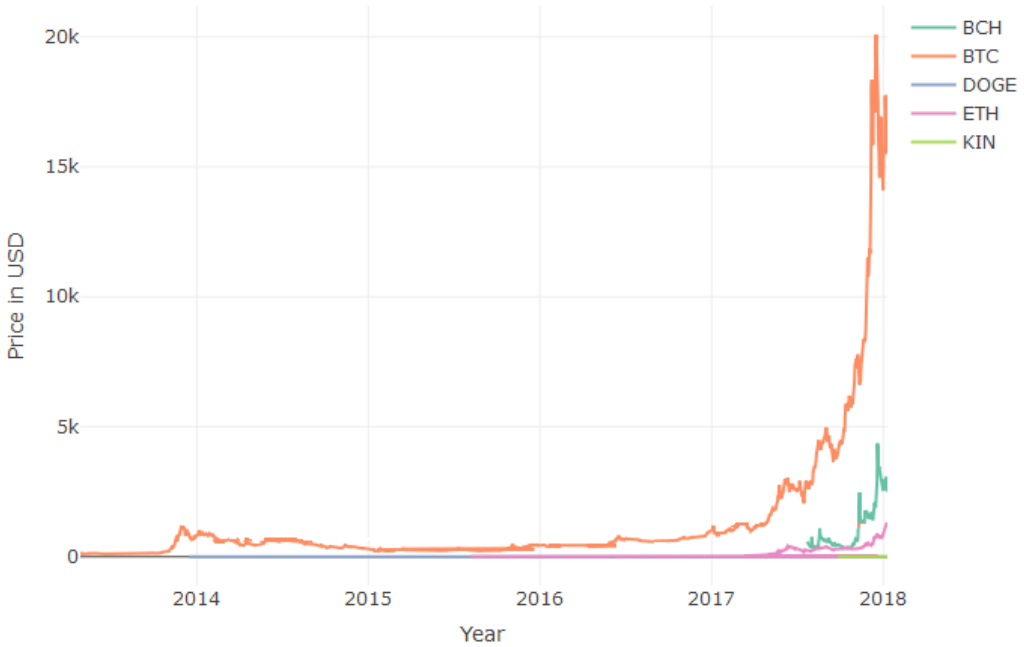

Figure 4.2: Time Series of top five Cryptocurrencies

From the above Time Series, it is quite evident that the Price of Bitcoin(BTC), is much more than that of other coins. Bitcoin House (BCH) and Ethereum (ETH) come second and third respectively. DOGE and KIN show up a little bit in this graph.

The price of DOGE is much lower than 1000 USD. Still, it's market capitalization is very high which means it must have a very high Circulating Supply to support the formula.

Market Cap = Price X Circulating Supply

CHAPTER 5
SYSTEM DESIGN AND ANALYSIS

CHAPTER 5

SYSTEM DESIGN AND ANALYSIS

5.1 Methodology

Step 1: First step is to analyze the input format of the data.

Step 2: If the format correspond to the correct type then move further otherwise change to the correct format.

Step 3: Have a look at the deviation of the values in the data or missing value

Step 4: Based on the percentage of deviation. Decide what method to use for replacing it with the correct value.

Step 5:

a) If the total percent is vary less than the whole population then replace it with 0.
b) If the total percent is less but the value is uniform than mean value can be placed in place of missing or corrupt data.
c) If the total percent is very less but outliers is high the median can be used.
d) If it does not match with the above type then different approach can be used based on the amount of invariability shown by the data.

Step 6: Next step is to find relationship between different attribute.

Step 7: Generate hypothesis which data should follow.

Step 8: Based on the hypothesis Explore the data.

a) Exploration can be graph based. (ggplot2 will be used for graph
b) Exploration can be Probabilistic.

Step 9: After the relationship is found, device a model which the dataset will follow.

Step 10: Divide the data into test and train set. The model will be formed with the train data and tested on train data. The outcome can be of three types:

a) Model is correct because hypothesis is good.

b) Hypothesis is wrong leading to wrong model, change the hypothesis

c) Data does not contain any relationship to model.

Step 11: If the model is correct then we can predict the outcome of the next behaviour of data, using which a user can invest properly in the Cryptocurrencies.

5.2 Technology

We have used R Studio as the IDE for our Project. **RStudio** is a free and open-source integrated development environment (IDE) for R, a programming language for statistical and graphics. RStudio was founded by JJ Allaire, creator of the programming language ColdFusion. Hadley Wickham is the Chief Scientist at RStudio.

RStudio is available in two editions: RStudio Desktop, where the program is run locally as a regular desktop application; and RStudio Server, which allows accessing RStudio using a web browser while it is running on a remote Linux server. Prepackaged distributions of RStudio Desktop are available for Windows, macOS, and Linux.

RStudio is available in open source and commercial editions and runs on the desktop (Windows, macOS, and Linux) or in a browser connected to RStudio Server or RStudio Server Pro (Debian, Ubuntu, Red Hat Linux, CentOS, openSUSE and SLES).

RStudio is written in the C++ programming language and uses the Qt framework for its graphical user interface.

CHAPTER 6
SOURCE CODE

CHAPTER 6

SOURCE CODE

Data Analysis

6.1 Code to set up the working directory and load the libraries

```
setwd("your_path_for_data")
library(dplyr)
library(ggplot2)
library(plotly)
library(extRemes)
library(moments)
```

6.2 Code to Find Out the Most Dominant Cryptocurrencies in the Market

```
file.list <- list.files(path="your_path_for_splitted_data ")
CoinName = c()
highestMarketCap = c()
for (i in 1:length(file.list)){
    file.df <- read.csv(file.list[i],header = TRUE)
    maxVal <- gsub(",","",max(as.character(file.df$Market.Cap)))
    maxVal <- as.numeric(maxVal)
    highestMarketCap <- c(highestMarketCap,maxVal)
    coinName <- c(coinName, as.character(file.df$coin[[1]]))
}
newdf <- data.frame(coinName,highestMarketCap)
newdf <- arrange(newdf,coinName,highestMarketCap)
newdf <- arrange(newdf,desc(highestMarketCap))
topfive <- newdf[seq(1:5),]
ggplot(topfive,aes(x=topfive$coinName,y=topfive$highestMarketCap,fill=t
opfive$coinName))+geom_bar(stat =
"identity")+xlab("Coins")+ylab("Market Cap")+ggtitle("Top 5
Cryptocurrencies")
```

6.3 Code to plot a graph showing the highest price attained by the Top Cryptocurrencies

```
btcData <- read.csv('btc.csv')
btcData$Date<-as.character(btcData$Date)
btcData$Date <- as.Date(btcData$Date,format='%Y-%m-%d')
ethData <- read.csv('ETH.csv')
ethData$Date<-as.character(ethData$Date)
ethData$Date <- as.Date(ethData$Date,format='%Y-%m-%d')
bchData <- read.csv('BCH.csv')
bchData$Date<-as.character(bchData$Date)
```

```
bchData$Date <- as.Date(bchData$Date,format='%Y-%m-%d')
DOGEData <- read.csv('DOGE.csv')
DOGEData$Date<-as.character(DOGEData$Date)
DOGEData$Date <- as.Date(DOGEData$Date,format='%Y-%m-%d')
KinData <- read.csv('KIN.csv')
KinData$Date<-as.character(KinData$Date)
KinData$Date <- as.Date(KinData$Date,format='%Y-%m-%d')
df1<- btcData %>%bind_rows(ethData,bchData,DOGEData,KinData)
plot_ly(x=df1$Date,y=df1$High,type='scatter',mode='lines',color=df1$coi
n)%>%
layout(xaxis =list(title='Year'), yaxis = list(title='Price in USD'))
```

6.4 Code to plot the Open, Close, Low and High Prices of the Top 5 Cryptocurrencies

```
ethData <- read.csv('ETH.csv')
DogeData <- read.csv('Doge.csv')
ethData$Date<-as.character(ethData$Date)
ethData$Date <- as.Date(ethData$Date,format='%Y-%m-%d')
DogeData$Date<-as.character(DogeData$Date)
DogeData$Date <- as.Date(DogeData$Date,format='%Y-%m-%d')
openData <-  function(coinData){

print(ggplot(data=coinData,aes(x=coinData$Date,y=coinData$Close))+
          geom_line(color="blue"))
}
closeData <-  function(coinData){
print(ggplot(data=coinData,aes(x=coinData$Date,y=coinData$Close))+
          geom_line(color="yellow"))
}
highData <-  function(coinData){
  print(ggplot(data=coinData,aes(x=coinData$Date,y=coinData$High))+
          geom_line(color="red"))
}
lowData <-  function(coinData){
  print(ggplot(data=coinData,aes(x=coinData$Date,y=coinData$Low))+
          geom_line(color="green"))
}
openData(ethData)
closeData(ethData)
lowData(ethData)
highData(ethData)
openData(DogeData)
closeData(DogeData)
lowData(DogeData)
highData(DogeData)
```

6.5 Code to iterate through files to obtain coins whose prices have gone greater than 1000$

```
highCoins = c()
for (i in 1:length(file.list)){
  file.df <- read.csv(file.list[i],header = TRUE)
  max.value <-max(file.df$High,na.rm=TRUE)

  if(max.value >= 1000)
  {
    highCoins <- c(highCoins,as.character(file.df$coin[1]))
    next
  }
}
length(highCoins)
highCoins
```

6.6 Plot for analysing the distribution of the delta of all coins in a yearly format

```
total.file<- read.csv("your_path_for_data")
length(total.file$High)
head(total.file)
total.file$Date<-as.character(total.file$Date)
total.file$Date <- as.Date(total.file$Date,format='%Y-%m-%d')
total.file$Date
new.total.file<- total.file %>% mutate(Year=
format(as.Date(total.file$Date, format="%Y-%m-%d"),"%Y"))%>%
mutate(Month=format(as.Date(total.file$Date, format="%Y-%m-%d"),"%m"))
head(new.total.file)
summary(new.total.file)
year<-c("2014","2015","2016","2017","2018")
typeof(year)
typeof(new.total.file$Year)
for(y in c("2014","2015","2016","2017","2018")){
  delta.yearly<-new.total.file%>%group_by(Year)%>%filter(Year==y)
 delta.yearly

print(ggplot(delta.yearly[delta.yearly$Delta>1&delta.yearly$Delta<10,],
aes(x=Date,y=Delta,    color=coin)) +geom_point()+
scale_x_date(date_breaks = "1 month", date_labels = "%B")+
theme(legend.position="none"))
}
```

6.7 Code to plot the distribution of the count of all coins delta whose value exceeds one

```
ggplot(total.file[total.file$Delta > 1  & total.file$Delta < 10, ],
aes(x=Delta, color=coin)) +geom_histogram() +
theme(legend.position="none")
```

6.8 Code to plot the distribution of the count of all coins delta whose value is less than one

```
ggplot(total.file[total.file$Delta <1, ], aes(x=Delta, color=coin)) +
geom_histogram() +
theme(legend.position="none")
```

6.9 Code to plot the distribution of the count of all coins delta whose value is less than 0

```
ggplot(total.file[total.file$Delta <0, ], aes(x=Delta, color=coin))
+geom_histogram() + theme(legend.position="none")
```

6.10 Code for calculating all the monthly delta for the whole year of top 5 coin by market share

```
coin.name<-c("BTC","BCH","DOGE","ETH","KIN")
total.delta.yearly.2017$Month<-
as.numeric(total.delta.yearly.2017$Month)
total.delta.yearly.2017
typeof(coin.name)
f<-list()
for(c in coin.name){
  monthly.delta=c()
  for(i in 1:12){
    total.delta.yearly.2017.months<-
total.delta.yearly.2017%>%filter(coin==c)%>%filter(Month==i)%>%
    summarise(TotalSum=sum(Delta))monthly.delta[i]<-
total.delta.yearly.2017.months
    }
    f[[c]]<-monthly.delta
}
df1 <- data.frame(matrix(unlist(f), nrow=5,
byrow=T),stringsAsFactors=FALSE)
df1 <- as.data.frame(t(df1))
colnames(df1)<-c("BTC","BCH","DOGE","ETH","KIN")
df1$month <-
c("jan","feb","mar","apr","may","jun","jul","aug","sep","oct","nov","de
c")
```

6.11 Code to plot the bar graph of monthly delta value

```
ggplot(data=df1,aes(y=df1$BTC,x=df1$month))+geom_bar(stat='identity')+l
abs(x="Month",y="Delta")+ggtitle("Monthly delta of BTC for the year
2017")

ggplot(data=df1,aes(y=df1$BCH,x=df1$month))+geom_bar(stat='identity')+l
abs(x="Month",y="Delta")+ggtitle("Monthly delta of BCH for the year
2017")

ggplot(data=df1,aes(y=df1$DOGE,x=df1$month))+geom_bar(stat='identity')+
labs(x="Month",y="Delta")+ggtitle("Monthly delta of DOGE for the year
2017")

ggplot(data=df1,aes(y=df1$ETH,x=df1$month))+geom_bar(stat='identity')+l
abs(x="Month",y="Delta")+ggtitle("Monthly delta of ETH for the year
2017")

ggplot(data=df1,aes(y=df1$KIN,x=df1$month))+geom_bar(stat='identity')+l
abs(x="Month",y="Delta")+ggtitle("Monthly delta of KIN for the year
2017")
```

6.12 Function used for analyzing the distribution of data having outliers

```
analysis<-function(openDataCoin){
 print(mean(openDataCoin))
 print(median(openDataCoin))
 print(sqrt(var(openDataCoin)))
 print(max(openDataCoin)-min(openDataCoin))
 print(hist(openDataCoin))
 print(skewness(openDataCoin))
 print(kurtosis(openDataCoin))
 print(hist(openDataCoin,prob=TURE))
 print(curve(dnorm(x,mean(openDataCoin),sd(openDataCoin)),col="red"
 ,add=TRUE))
 print(qqnorm(openDataCoin))
 print(qqline(openDataCoin,col="red"))
 print(qqPlot(openDataCoin,distribution = "norm"))
}
```

6.13 Checking outliers of open column for btc

```
BTC.2014 <- new.total.file%>%filter(coin=="BTC")%>%filter(Year==2014)
BTC.Open.2014<-BTC.2014$Open
boxplot(BTC.Open.2014,
        horizontal = TRUE,
        las=1,
```

```
        notch = TRUE,
        col="slategray3",
        ylim=c(100,1000),
        boxwex=0.5,
        whisklty=1,
        main="Opening of btc for the year 2014"
        ,xlab="btc Open ")
BTC.2015 <- new.total.file%>%filter(coin=="BTC")%>%filter(Year==2015)
BTC.Open.2015<-BTC.2015$Open
boxplot(BTC.Open.2015,
        horizontal = TRUE,
        las=1,
        notch = TRUE,
        col="slategray3",
        ylim=c(100,500),
        boxwex=0.5,
        whisklty=1,
        main="Opening of btc for the year 2015"
        ,xlab="btc Open ")
analysis(BTC.Open.2015)
BTC.2016 <- new.total.file%>%filter(coin=="BTC")%>%filter(Year==2016)
BTC.Open.2016<-BTC.2016$Open
boxplot(BTC.Open.2016,
        horizontal = TRUE,
        las=1,
        notch = TRUE,
        col="slategray3",
        ylim=c(100,1000),
        boxwex=0.5,
        whisklty=1,
        main="Opening of btc for the year 2016"
        ,xlab="btc Open ")
BTC.2017 <- new.total.file%>%filter(coin=="BTC")%>%filter(Year==2017)
BTC.Open.2017<-BTC.2017$Open

boxplot(BTC.Open.2017,
        horizontal = TRUE,
        las=1,
        notch = TRUE,
        col="slategray3",
        ylim=c(800,12000),
        boxwex=0.5,
        whisklty=1,
        main="Opening of btc for the year 2017"
        ,xlab="btc Open ")
analysis(BTC.Open.2017)
```

6.14 Checking outliers of open column for bch

```
BCH.2017 <- new.total.file%>%filter(coin=="BCH")%>%filter(Year==2017)
BCH.Open.2017<-BCH.2017$Open
boxplot(BCH.Open.2017,
        horizontal = TRUE,
        las=1,
        notch = TRUE,
        col="slategray3",
        ylim=c(100,5000),
        boxwex=0.5,
        whisklty=1,
        main="Opening of bch for the year 2017"
        ,xlab="bch Open ")
analysis(BCH.Open.2017)
BCH.2018 <- new.total.file%>%filter(coin=="BCH")%>%filter(Year==2018)
BCH.Open.2018<-BCH.2018$Open
boxplot(BCH.Open.2018,
        horizontal = TRUE,
        las=1,
        notch = TRUE,
        col="slategray3",
        ylim=c(100,5000),
        boxwex=0.5,
        whisklty=1,
        main="Opening of bch for the year 2018"
        ,xlab="bch Open ")
```

Regression Analysis

6.15 Code for data cleaning and modification

```
import numpy as np
import pandas as pd
import os
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
os.chdir("C:/Users/sudarshan/FYP/Data")
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
crypto = pd.read_csv('CryptocoinsHistoricalPrices.csv')
crypto.head()
crypto.info()
```

```
crypto.describe()
crypto.columns
crypto.drop(['Unnamed: 0'], axis=1, inplace=True)
crypto.head()
crypto.dropna(axis=0, inplace=True)
crypto['Year']=crypto['Date'].apply(lambda x: x.split('-')[0])
df= crypto[crypto['Market.Cap']!='-']
df['Market.Cap'] = df['Market.Cap'].str.replace(',', '')
df['Market.Cap']=df['Market.Cap'].astype(float)
df= df[df['Year']=='2017']
coins_type=temp_df.groupby('coin').count().sort_values(by='Volume',asce
nding=False).head(10)
top_coins_name=list(coins_type.index)
df_top = df[df['coin'].isin(top_coins_name)]
df_top['Month']=df_top['Date'].apply(lambda x: x.split('-')[1])
df_top['Volume'] = df_top['Volume'].str.replace(',', '')
df_top['Volume']=df_top['Volume'].astype(float)
```

6.16 Code for generating bar graph showing the volume of top ten cryptocurrencies

```
plt.figure(figsize=(12,6))
sns.barplot(x='coin', y='Volume', data=df_top, estimator=np.mean)
```

6.17 Code for prediction of REP coin

```
rip_table = df_btc = df[df['coin']=='REP']
sns.heatmap(rip_table.corr(),annot=True)
rip_table['Volume']=rip_table['Volume'].str.replace(',','')
rip_table['Volume']=rip_table['Volume']
X=rip_table[['Open', 'High', 'Low', 'Volume', 'Market.Cap']]
y=rip_table['Close']
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=101)
lm=LinearRegression()
lm.fit(X_train, y_train)
print(lm.intercept_)
cdf=pd.DataFrame(lm.coef_,X.columns, columns=['Coeff'])
predictions=lm.predict(X_test)
plt.scatter(y_test, predictions)
sns.distplot((y_test- predictions))
```

6.18 Code for prediction of BTC coin

```
btc_table = df[df['coin']=='BTC']
btc_table['Volume']=btc_table['Volume'].str.replace(',','')
X=btc_table[['Open','High', 'Low', 'Volume', 'Market.Cap']]
```

```
y=btc_table['Close']
lmlm==LinearRegressionLinearReg ()
lm.fit(X_train, y_train)
print(lm.intercept_)
cdf=pd.DataFrame(lm.coef_,X.columns, columns=['Coeff'])
predictions=lm.predict(X_test)
plt.scatter(y_test, predictions)
```

CHAPTER 7

Snapshots

CHAPTER 7

SNAPSHOTS

TOP 5 Cryptocurrencies by Market Cap

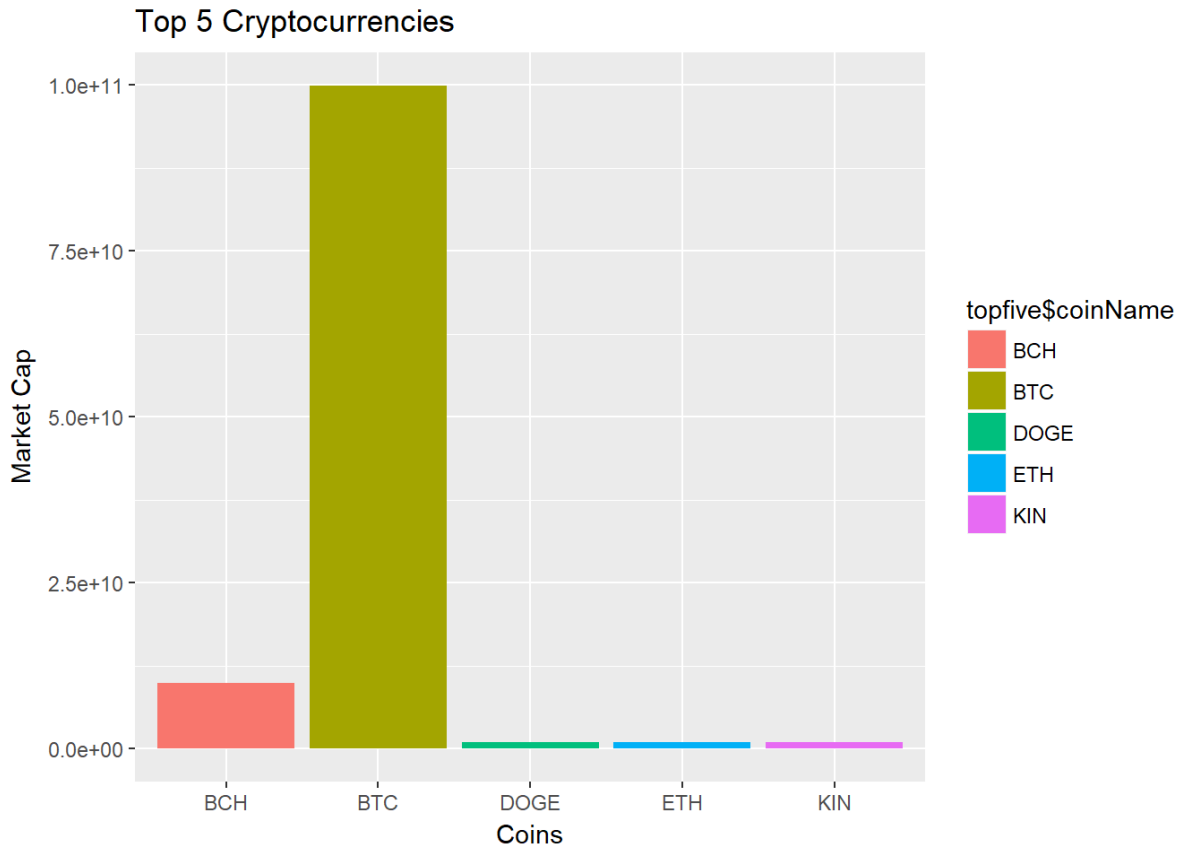

Figure 7.1 Top 5 Cryptocurrencies by Market Cap

This bar graph shows the Top 5 Cryptocurrencies by Market Cap according to our data. Since, or data is static and the cryptocurrency market fluctuates at a very high rate, the top 5 Cryptocurrencies at the current moment may differ from those found by the graph.

Time Series of the Top 5 Cryptocurrencies

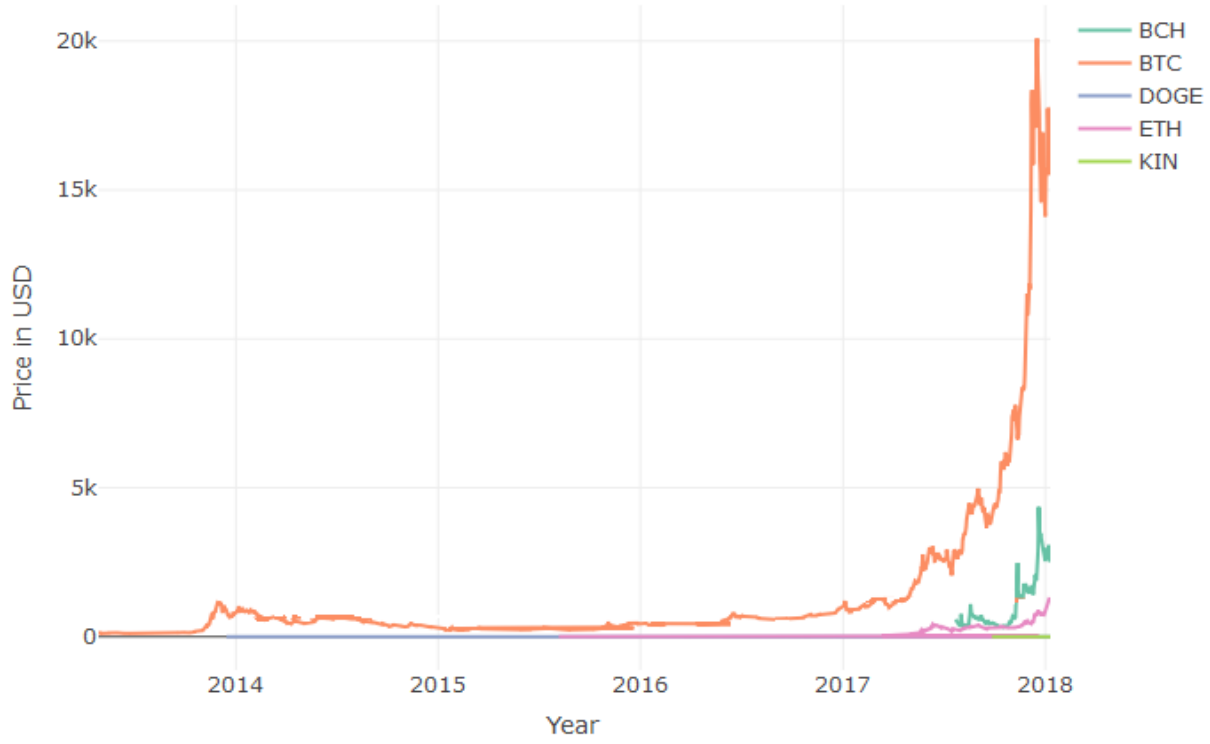

Figure 7.2 Highest Prices attained by Top 5 Coins

This time series shows the prices attained by each of the top 5 Cryptocurrencies by Market Cap found in the previous graph. As it is evident from the graph, the price of Bitcoin(BTC) is much more higher than other that of other Cryptocurrencies. Also, the fluctuation in the prices happen mainly after 2016.

Time Series of Bitcoin

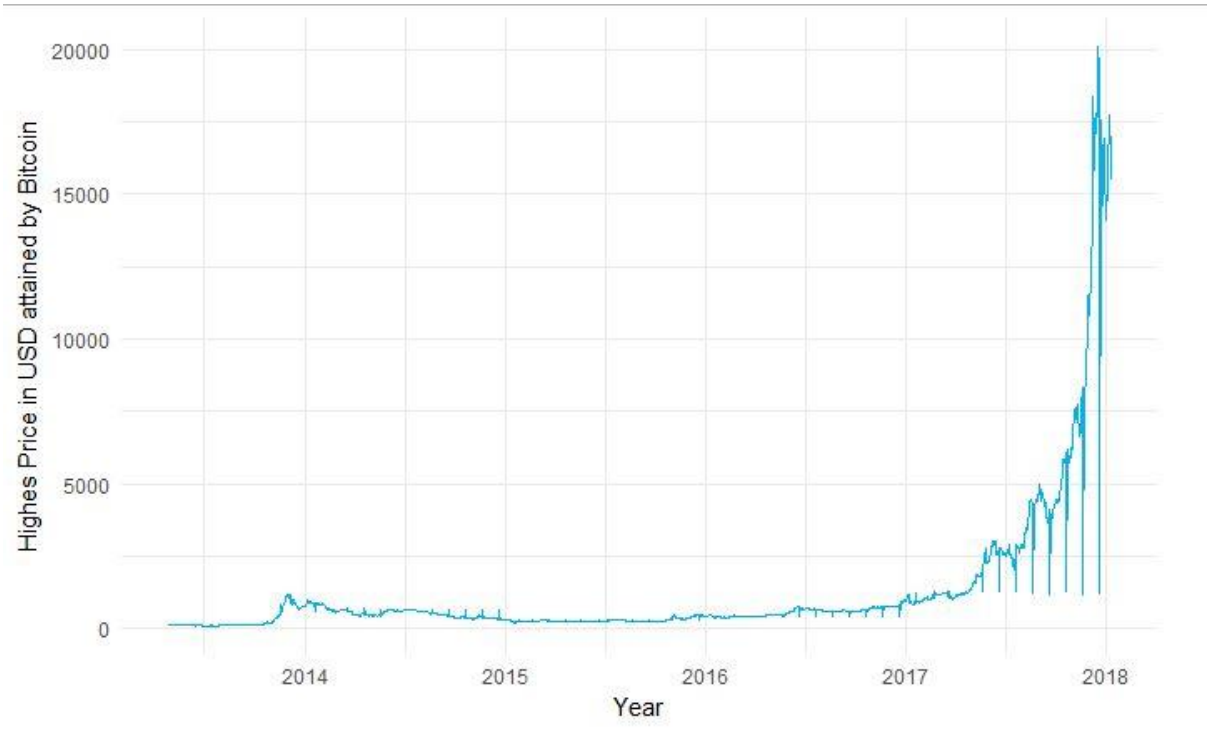

Figure 7.3 Highest Prices attained by Bitcoin over the years

This time series shows the prices attained by Bitcoin (BTC) over the years. Bitcoin was the first cryptocurrency that came in the market and is the most important cryptocurrency till date. The graph shows us that the prices increased significantly after 2016. Before 2016, the little fluctuations in the prices we see was in 2014.

Scatter Plot that shows Delta for 2014

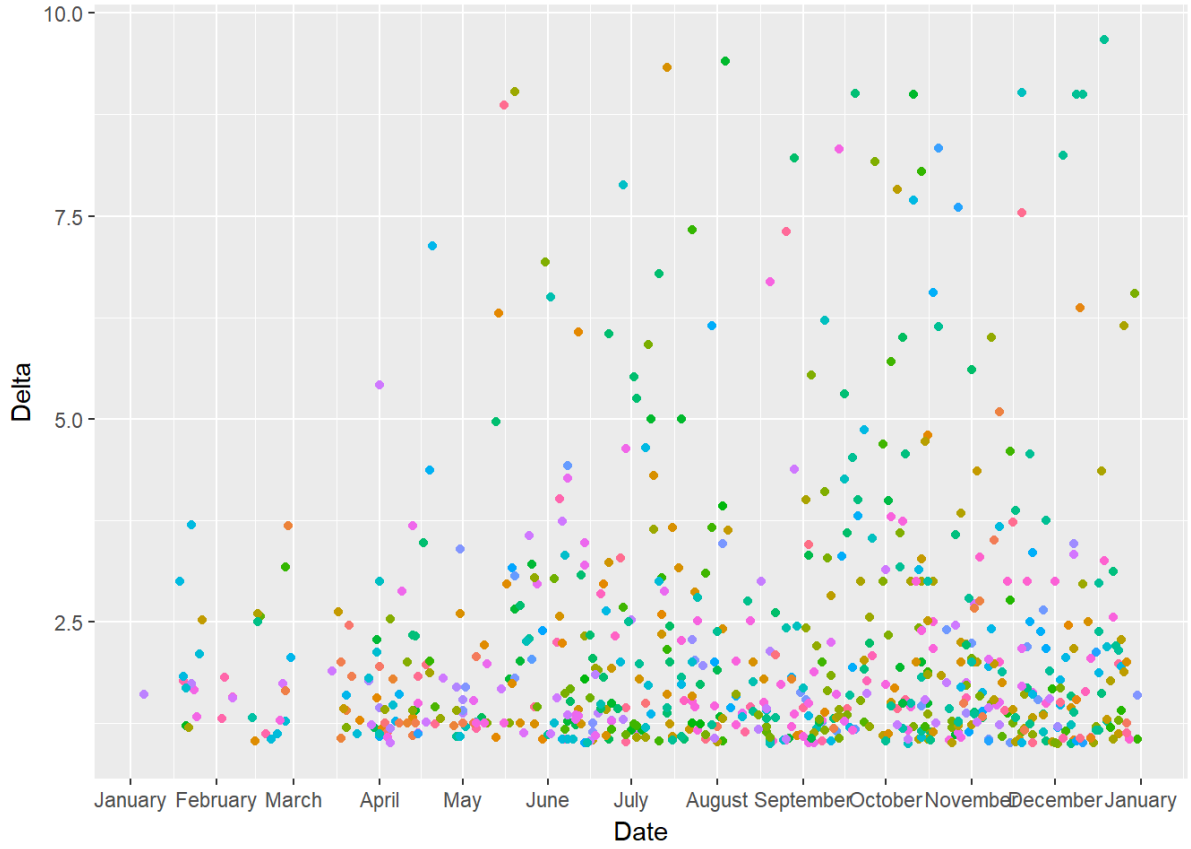

Figure 7.4 Delta for 2014

Delta is a column that we have added ourselves to the data. Delta, for a day, is the difference of Close and Open divided by Open. So, if delta for a particular coin is greater than 1, it indicates that investing in the coin is profitable. While if it is less than one, that indicates that the coin is not profitable. In the above graph, each point represents a coin. As we can see, the number of points are a bit scarce here.

Scatter Plot that shows Delta for 2015

Figure 7.5 Delta for 2015

As it is evident from the scatter plot, the number of dots has increased from 2014 indicating that new coins came in the market.

Scatter Plot that shows Delta for 2016

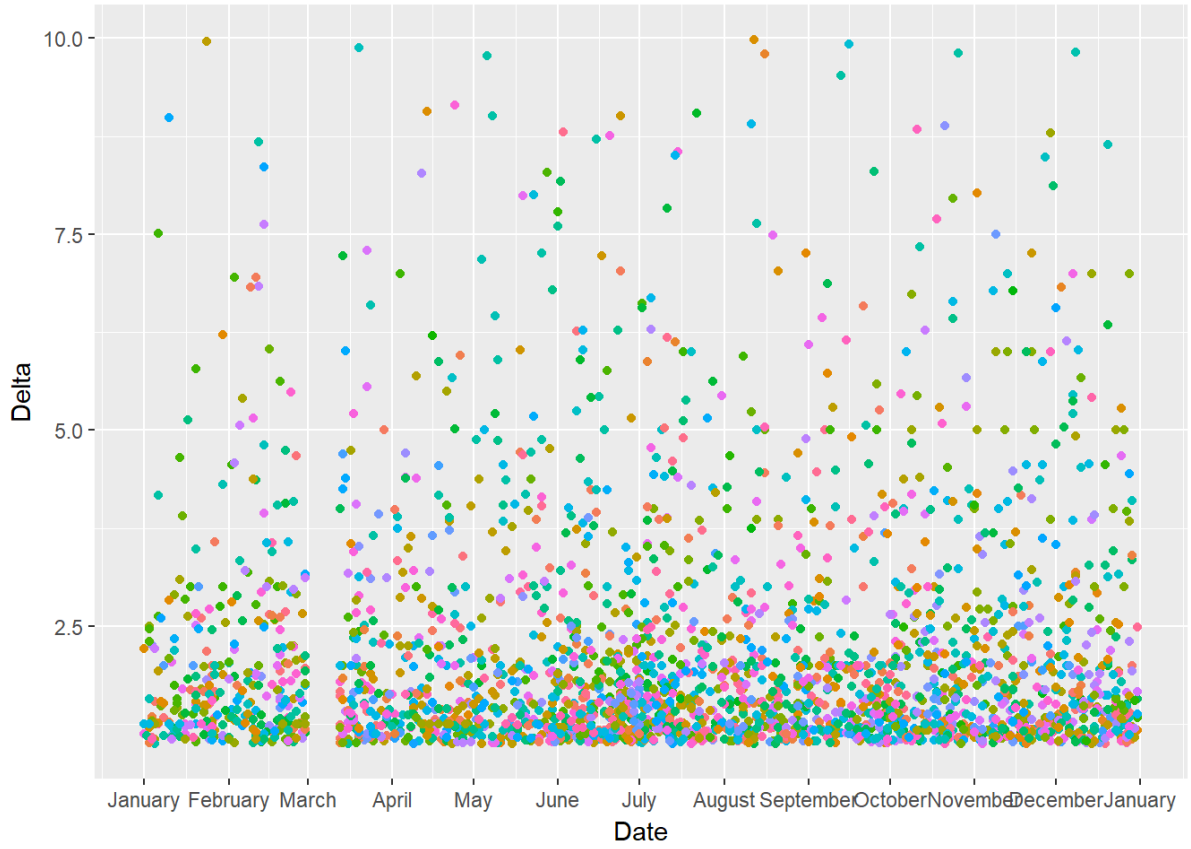

Figure 7.6 Delta for 2016

Many more new coins came in the market in 2016. This indicates that the Cryptocurrency market is a growing market.

Scatter Plot that shows Delta for 2017

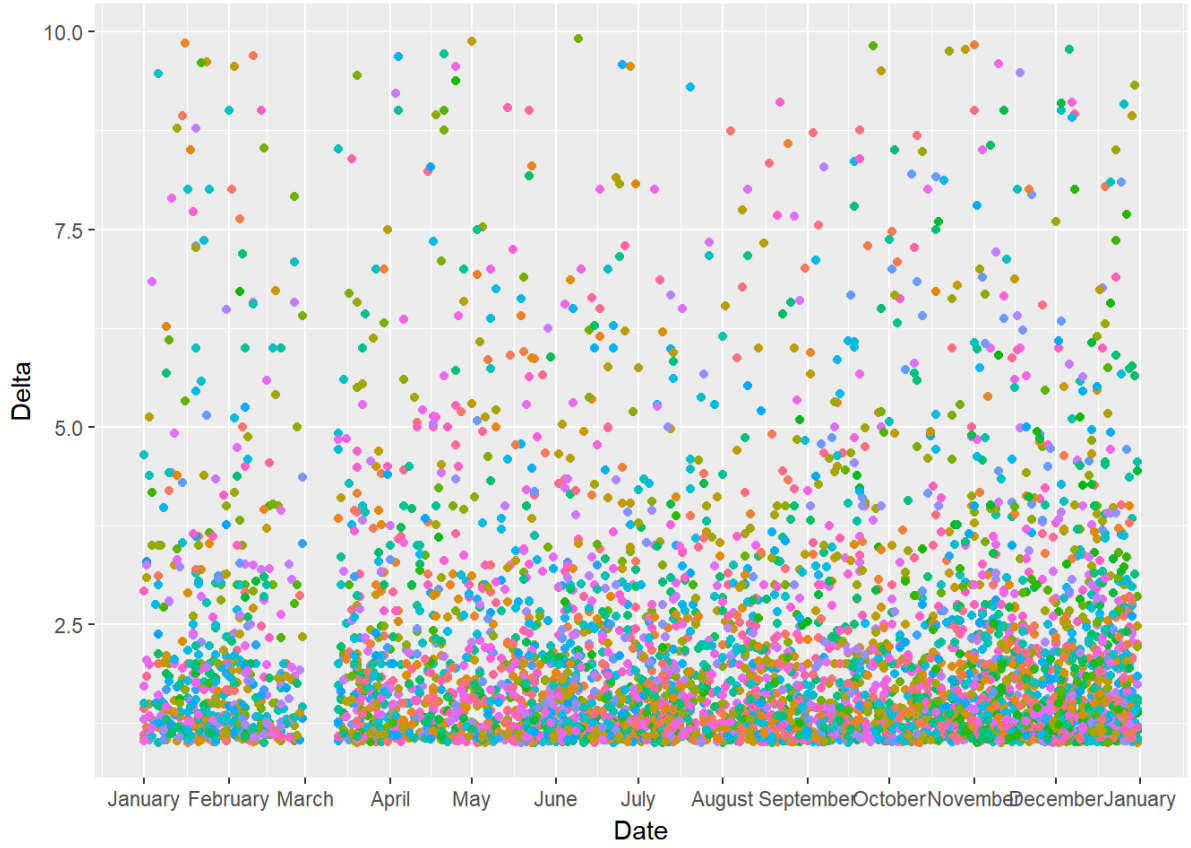

Figure 7.7 Delta for 2017

The graph has become more condensed indicating a significant increase in the number of coins from 2014 to 2017.

Scatter Plot that shows Delta for 2018

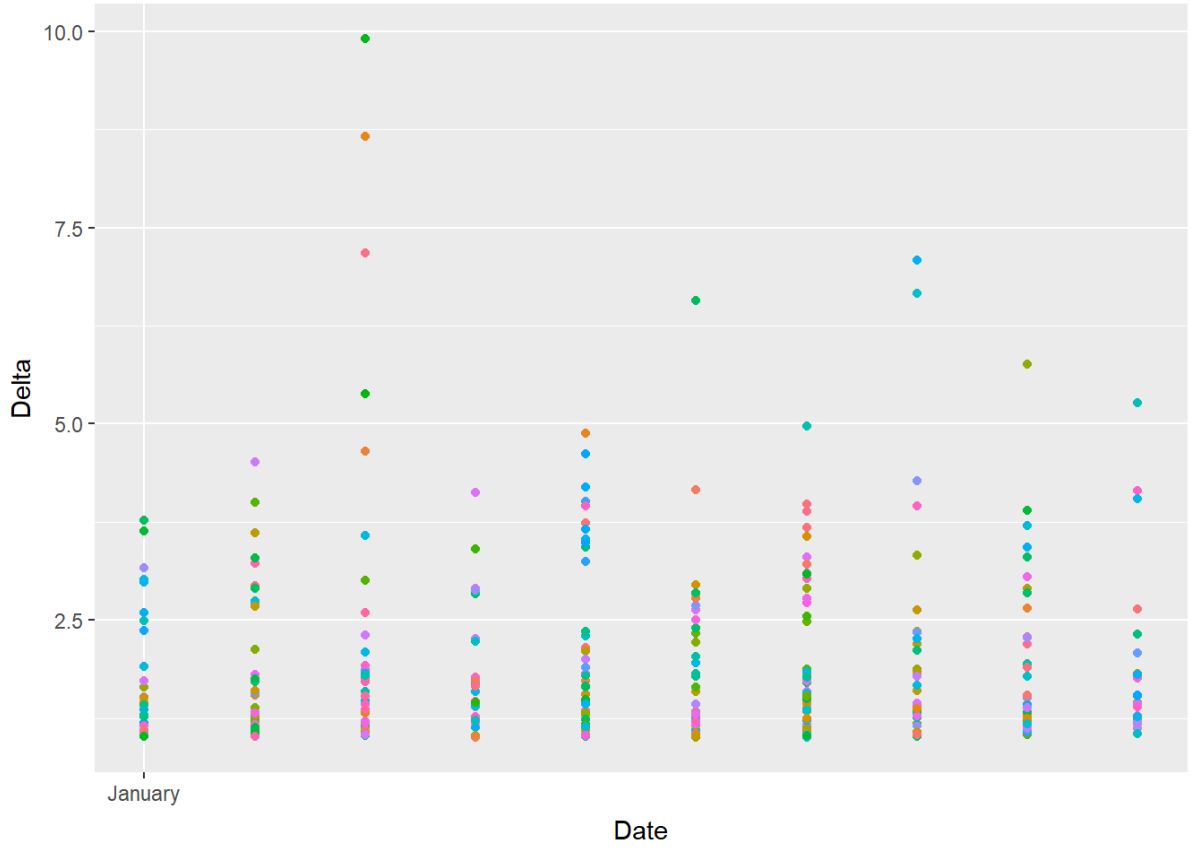

Figure 7.8 Delta for 2018

Since our data did not contain much information about all the coins for the year 2018, the graph here is scarce. If we plot the scatter plot based on the real time data, this plot will become even more condense.

Histogram that shows the number of times delta exceeded 1 for every coin

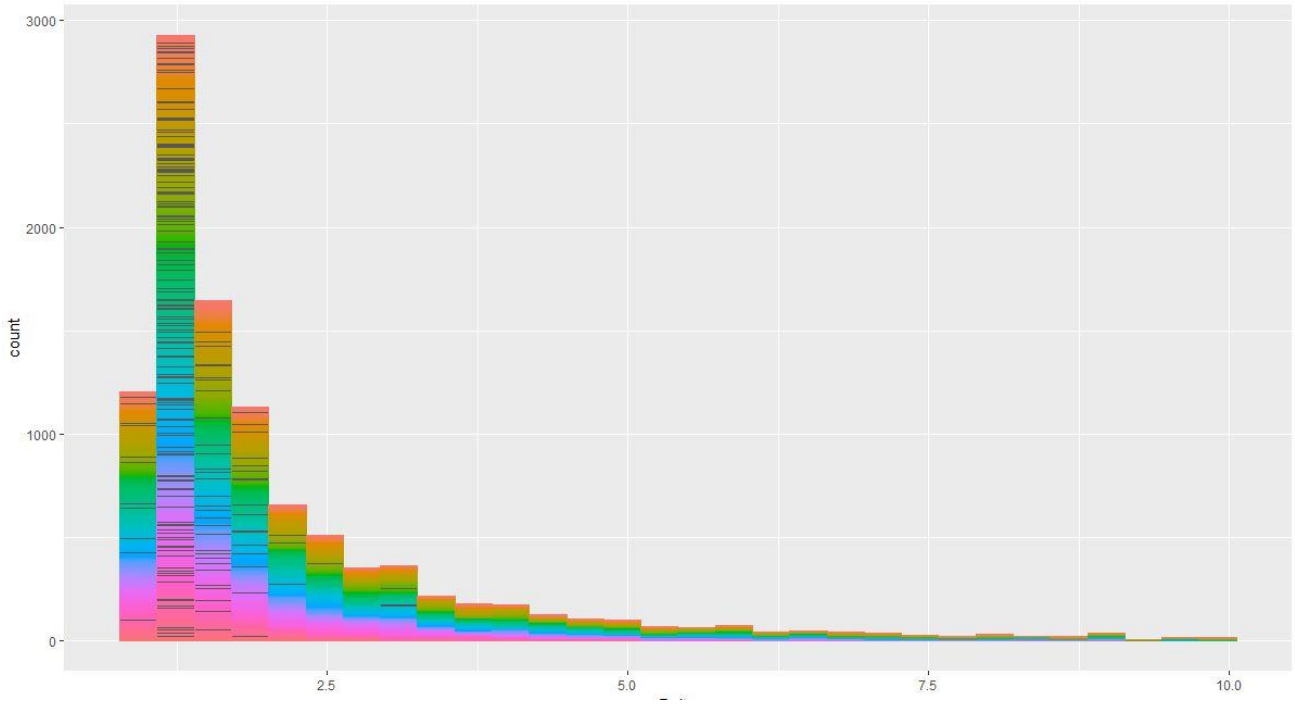

Figure 7.9 Number of Coins whose Delta exceed 1

As the intuition says, most number of coins have delta less than 2. A very few coins have seen their delta going more than 5.

Histogram that shows the number of times delta remained less than 1 for every coin

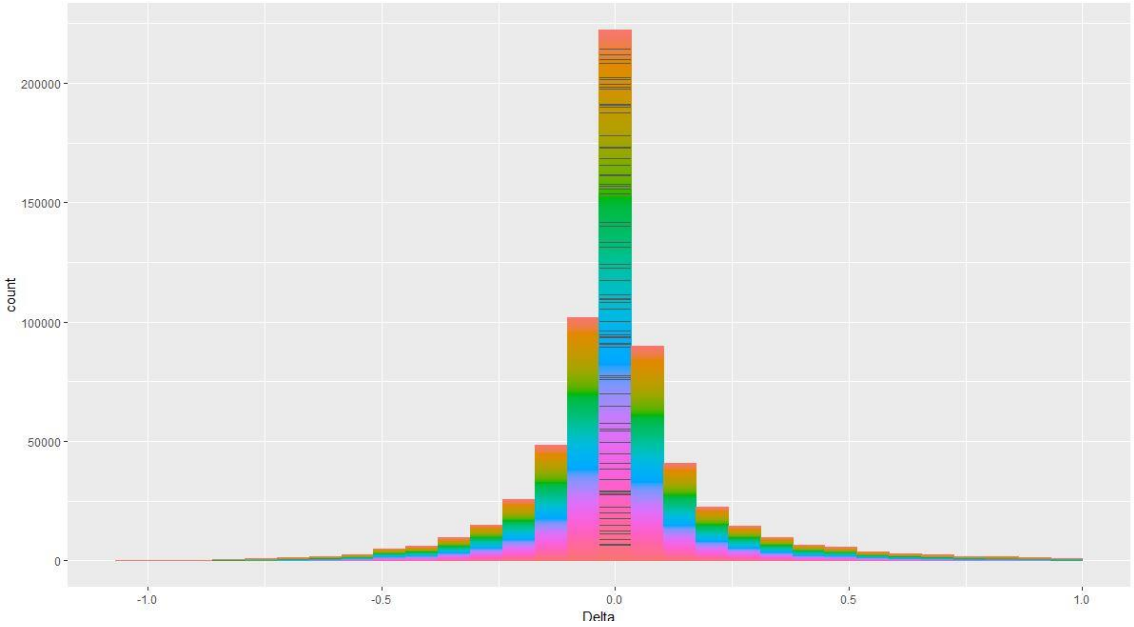

Figure 7.10 Number of Coins whose Delta is less than 1

Since there are coins whose delta is less than 1, it shows that there are coins which are not safe to incest in. Also, maximum number of coins in this histogram has delta between 0 and 1. It shows that there many coins in which investing doesn't return high profits but it also doesn't let the investor suffer loss.

Box Plot to find Outliers for High column of Bitcoin

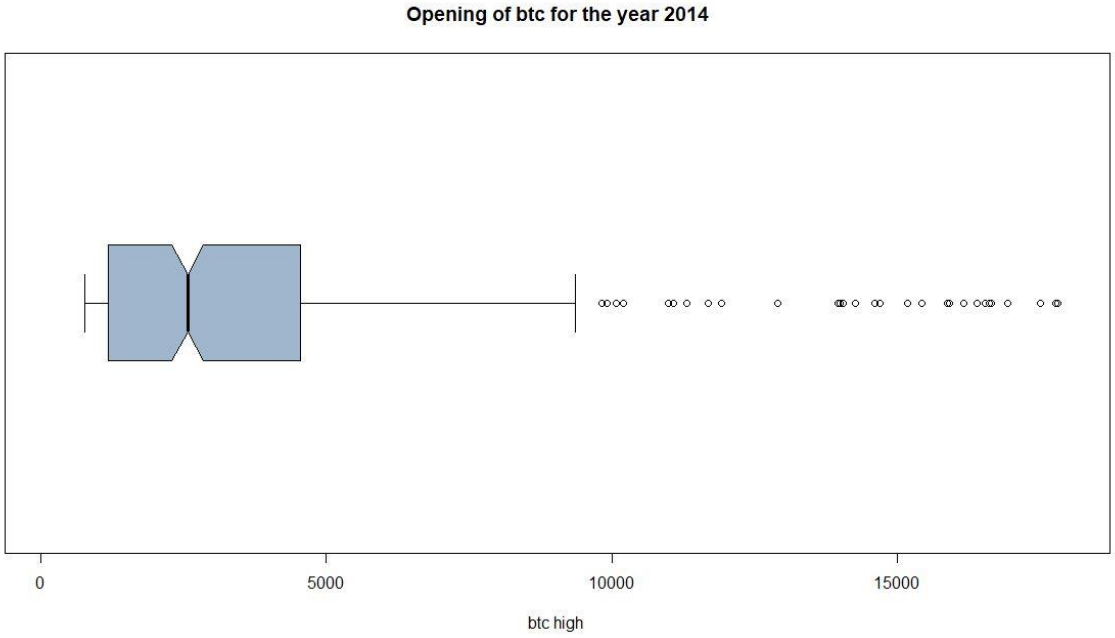

Figure 7.11 Box Plot Showing the Outliers for BTC

The first vertical line shows the minimum value, the last vertical line shows the maximum value and the middle most line shows the median. But as we see, there are a number of dots outside the maximum value which is not possible. This is because the price of Bitcoin increased after 2016 at a very high rate which is unpredictable. When such scenarios happen, the context of the Box Plot changes.

Correlation between all the variables in the data

Figure 7.12 Correlation between different variables in the data

Prediction for REP

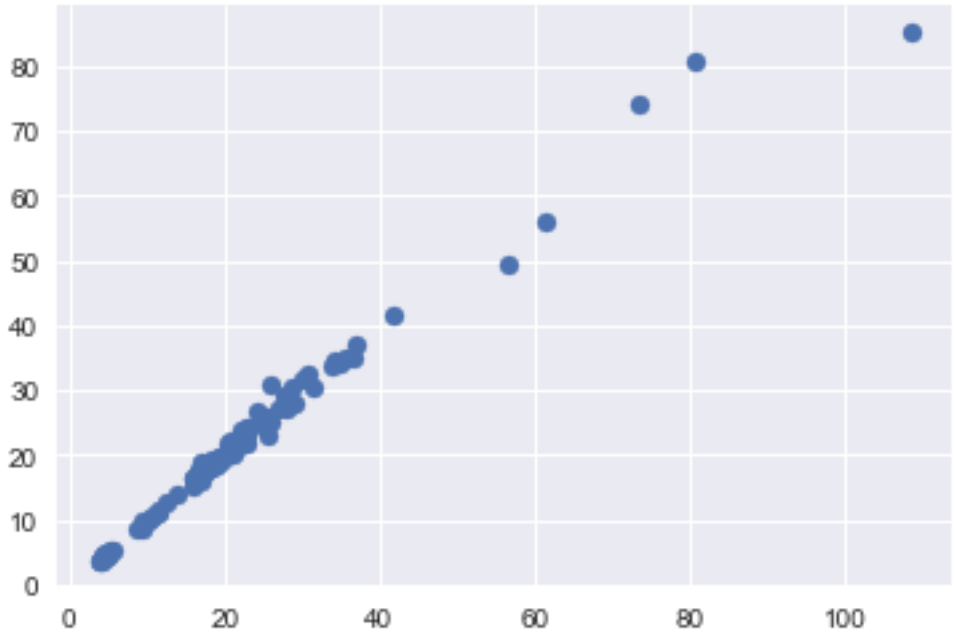

Figure 7.13 Linear Regression for REP coin

Screenshot of the Shiny WebApp

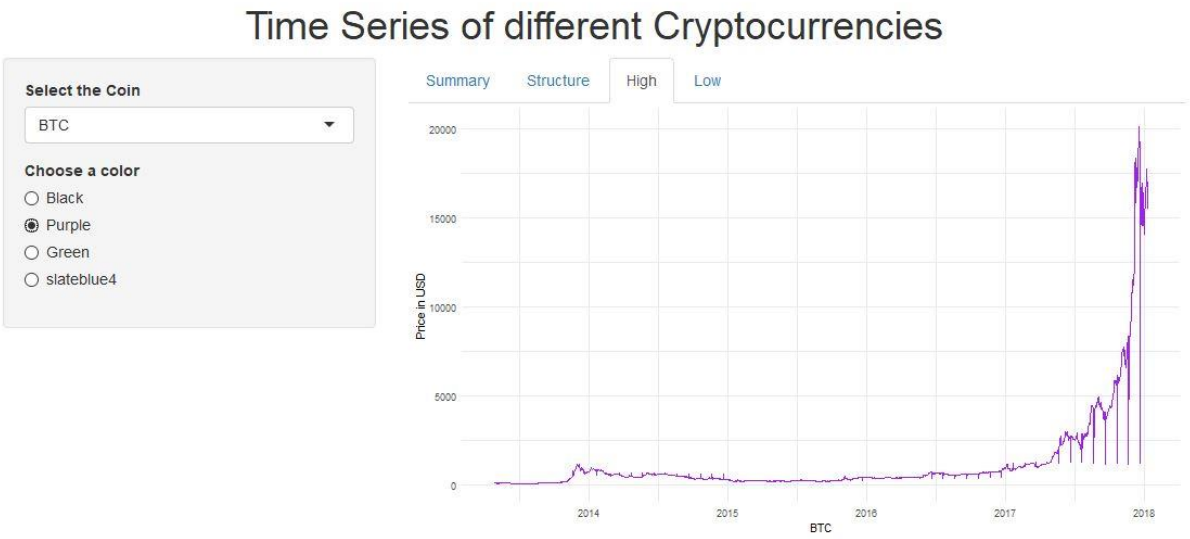

Figure7.14 Shiny App to View Time Series of Different Cryptocurrencies

Shiny is an R package that makes it easy to create dynamic web apps straight from R. The above screenshot is of a shiny WebApp that allows the user to select the coin of his/her choice and shows the time series of the selected coin as the output.

CHAPTER 8
Conclusion

CHAPTER 8

CONCLUSION

Exploratory Data Analysis is emerging field. The amount of data that is generated daily is huge. Combining it with Machine Learning forms an interesting combination. These are very vast fields and as time progresses, the implementation of these techniques in the project can be improved in order to produce more reliable outputs.

Cryptocurrency Market is an emerging market that attracts a lot of attention. This project gave us a lot of insights into how the Cryptocurrency Market works. We learned a lot about different packages in R and regression in Machine Learning. But after all this, Cryptocurrency Market is somewhat unpredictable. Nobody knows what exactly will happen in the next moment. Yet, we just gave a try at exploring and predicting the trends in these market.

CHAPTER 9

REFERENCES

CHAPTER 9

REFERENCES

[1] Introduction to R Software by NPTEL conducted by IIT Kharagpur.

[2] Introduction to Data Analysis by NPTEL conducted by IIT Madras.

[3] Exploratory Data Analysis on Udacity.

[4] R in Action: Data Analysis and Graphics with R, 2nd Edition, Robert L Kabacoff.

[5] THE VISUAL DISPLAY OF QUANTITATIVE INFORMATION, 2nd Edition, Edward R Tufte

[6] Kannan KS, Sekar PS, Sathik MM, Arumugam P. Financial stock market forecast using data mining techniques. InProceedings of the International Multiconference of Engineers and computer scientists 2010 Mar 17 (Vol. 1, p. 4).

[7] Olaniyi SA, Adewole KS, Jimoh RG. Stock trend prediction using regression analysis–a data mining approach. ARPN Journal of Systems and Software. 2011 Jul;1(4):154-7.

[8] For Obtaining Datasets - https://www.kaggle.com/rajanand/education-in-india

[9] For technical queries - https://stackoverflow.com/

[10]https://www.r-bloggers.com/

[11]http://www.sthda.com/english/wiki/wiki.php

[12] Plotly Package : https://plot.ly

[13] Deepika EP, Kaur ER. *Cryptocurrency: Trends, Perspectives and Challenges.*

[14] Bitcoin Wikipedia. Available:http://ru.wikipedia.org/wik

[15] Lamon C, Nielsen E, Redondo E. Cryptocurrency *Price Prediction Using News and Social Media Sentiment. Not Published*. 2016.