UNIVERSITY OF THESSALY

MASTER OF ENGINEERING

Cryptocurrency Analysis & Predictions Utilizing Machine Learning

---

Author:
Athanasios ZOUMPEKAS

Supervisors:
Emmanouil VAVALIS
Michael VASSILAKOPOULOS

A thesis submitted in fulfillment of the requirements for the degree of Master of Engineering

in the

Department of Electrical & Computer Engineering

June 20, 2018

Declaration of Authorship

I, Athanasios ZOUMPEKAS, declare that this thesis titled, "Cryptocurrency Analysis & Predictions Utilizing Machine Learning" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"I have been impressed with the urgency of doing. Knowing is not enough; we must apply. Being willing is not enough; we must do."

Leonardo da Vinci

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

Περίληψη

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Προπτυχιακό Δίπλωμα

Ανάλυση Κρυπτονομισμάτων **&** Προβλέψεις με τη χρήση Μηχανικής Μάθησης

Αθανάσιος Ζουμπέκας

Η αγορά κρυπτονομισμάτων αναπτύχθηκε ταχέως και ασυνεπώς κατά τη διάρκεια της σύντομης διάρκειας ζωής της. Μετά την άφιξη του πρωτοποριακού κρυπτονομίσματος, Bitcoin, τον Ιανουάριο του 2009, έχουν δημιουργηθεί πάνω από 1200 κρυπτονομίσματα, με την πλειοψηφία τους να παρουσιάζει μέτρια επιτυχία. Η μεταβλητότητα αυτής της αγοράς παρουσιάζει σημαντικές ερευνητικές προκλήσεις και δικαιολογεί την εντατική αξιολόγηση της συμπεριφοράς τους. Αυτή η εργασία προσπαθεί να παρέχει μια ολοκληρωμένη ανάλυση δεδομένων της αγοράς κρυπτονομισμάτων. Για το λόγο αυτό, αναπτύσσουμε και εφαρμόζουμε τεχνικές στατιστικής και μηχανικής μάθησης για να αναλύουμε την κίνηση των τιμών συγκεκριμένων κρυπτονομισμάτων και να εξάγουμε συμπεράσματα. Συγκεκριμένα, χρησιμοποιούμε μια ανάλυση συσχετισμού των δεδομένων και εφαρμόζουμε αλγόριθμους μηχανικής μάθησης για να προβλέψουμε την τιμή κλεισίματος του κρυπτονομίσματος του Ethereum σε σύντομο χρονικό διάστημα. Τα δεδομένα τιμών συγκεντρώνονται από το ανταλλακτήριο Poloniex και την πλατφόρμα Quandl. Για την ανάλυση των δεδομένων που λήφθηκαν, δημιουργήσαμε και εφαρμόσαμε ένα Convolutional Neural Network (CNN) με ποικίλο αριθμό στρώσεων και δύο τύπους Recurrent Neural Network (RNN), το δίκτυο LSTM (Long Short-Term Memory) και το Gated Recurrent Unit (GRU). Τα προαναφερθέντα μοντέλα βαθιάς μάθησης συγκρίνονται χρησιμοποιώντας διάφορες μετρήσεις. Παρατηρήσαμε ότι το καλύτερο από τα παραπάνω μοντέλα μπορεί να χρησιμοποιηθεί για την πρόβλεψη της τιμής κλεισίματος του Ethereum σε πραγματικό χρόνο.

UNIVERSITY OF THESSALY

Abstract

Department of Electrical & Computer Engineering

Master of Engineering

Cryptocurrency Analysis & Predictions Utilizing Machine Learning

Athanasios ZOUMPEKAS

The cryptocurrency market has been developed rapidly and inconsistently with exceptional speed during its short lifespan. After the arrival of the avant-garde lawless cryptocurrency, Bitcoin, in January 2009, more than 1200 cryptocurrencies have been introduced with the majority of them exhibiting modest success. The volatility of this market presents significant research challenges and justifies intensive evaluation of its behavior. This thesis attempts to provide a comprehensive data analysis of the cryptocurrency market. For this, we develop and apply statistical and machine learning techniques to analyze the price movement of specific cryptocurrencies and generate inferences. Specifically, we utilize a correlation analysis of the data and apply machine learning algorithms to predict the closing price of the cryptocurrency Ethereum in a short period. The price data is accumulated from Poloniex exchange and Quandl platform. For the analysis of the data obtained, we have implemented and applied a Convolutional Neural Network (CNN), with varying numbers of layers and two types of Recurrent Neural Network (RNN), the Long Short Term Memory (LSTM) network and Gated Recurrent Unit (GRU) network. The above deep learning models are benchmarked and compared utilizing various metrics. We have observed that the best of the above models can be used to predict the Ethereum closing price in real time.

Acknowledgements

I would first like to thank my thesis supervisor and my mentor Professor Emeritus Elias N. Houstis of the Department of Electrical & Computer Engineering at University of Thessaly and Department of Computer Science at Purdue University. The door to Prof. Houstis office was always open whenever I ran into a trouble spot or had a question about my research or writing. I have been extremely lucky to have a supervisor who cared so much about my work, and who responded to my questions and queries so promptly. He consistently allowed this thesis to be my own work, but steered me in the right the direction whenever he thought I needed it.

I would also like to thank the rest project supervisors who were involved in the validation survey for this research project: Professor of the Department of Electrical & Computer Engineering at University of Thessaly, Emmanouil Vavalis and Associate Professor, Michael Vassilakopoulos. Without their participation and input, the validation survey could not have been successfully conducted.

I would also like to personally acknowledge Professor Emmanouil Vavalis of the Department of Electrical & Computer Engineering at University of Thessaly as the second reader of this thesis, despite his many occupations, and I am gratefully indebted to his for his very valuable comments on this thesis.

Finally, I must express my very profound gratitude to my parents, to my brother and to my friends for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

Thank you all !

Author
Athanasios ZOUMPEKAS

Contents

List of Figures

List of Tables

List of Abbreviations

OTC	**O**ver **T**he **C**ounter
BTC	**Bitc**oin
DApps	**D**istributed **Applications**
XRP	**R**i**p**ple
ANN	**A**rtificial **N**eural **N**etwork
SVM	**S**upport **V**ector **M**achine
ARMA	**A**uto **R**egressive **M**oving **A**verage
ARIMA	**A**uto **R**egressive **I**ntegrated **M**oving **A**verage
AC	**A**uto**c**orrelation
PAC	**P**artial **A**uto**c**orrelation
ACF	**A**uto**c**orrelation **F**unction
PACF	**P**artial **A**uto**c**orrelation **F**unction
RNN	**R**ecurrent **N**eural **N**etwork
FFNN	**Feed**-**F**orward **N**eural **N**etwork
LSTM	**L**ong **S**hort **T**erm **M**emory
GRU	**G**ated **R**ecurrent **U**nit
API	**A**pplication **P**rogramming **I**nterface
USD	**U**nited **S**tates **D**ollar
ETH	**Eth**ereum
CNN	**C**onvolutional **N**eural **N**etwork
CNTK	Microsoft **C**og**n**itive **T**ool**k**it
CPU	**C**entral **P**rocessing **U**nit
GPU	**G**raphics **P**rocessing **U**nit
IDE	**I**ntegrated **D**evelopment **E**nvironment
DDoS	**D**istributed **D**enial **o**f **S**ervice
mln	**million**
LTC	**Lit**e**c**oin
ETC	**Et**hereum **C**lassic
STR	**St**ella**r**
DASH	**Dash**
SC	**S**ia**c**oin
XMR	**M**one**r**o
ZEC	**Z**cash
LSK	**Lisk**
STRAT	**Strat**is
IQR	**I**nter**q**uartile **r**ange
ReLU	**Re**ctified **L**inear **U**nit
LeakyReLU	**Leaky** **Re**ctified **L**inear **U**nit
TanH	**H**yperbolic **Tan**gent
MSE	**M**ean **S**quared **E**rror
MAE	**M**ean **A**bsolute **E**rror
RMSE	**R**oot **M**ean **S**quared **E**rror
RMSD	**R**oot-**M**ean-**S**quare **D**eviation

Dedicated to my beloved family & friends . . .

Chapter 1

Introduction

1.1 Digital Currency

Digital currency (digital money, electronic money, electronic currency) is a kind of currency accessible just in digital form, not in physical (for example, banknotes and coins). It shows properties similar to physical currencies, but allows for instantaneous transactions and border-less transfer-of-ownership. Examples include virtual currencies and cryptocurrencies [1] or even central bank issued "digital base money". Like traditional money, these currencies may be used to buy physical goods and services, but may also be restricted to certain communities such as for use inside an on-line game or social network [2].

Digital currency is a money balance recorded electronically on a stored-value card or other device. Another form of electronic money is network money, allowing the transfer of value on computer networks, particularly the Internet. Electronic money is also a claim on a private bank or other financial institution such as bank deposits [3]. Digital money can either be centralized, where there is a central point of control over the money supply, or decentralized, where the control over the money supply can come from various sources.

1.2 Cryptocurrency

Cryptocurrency is a form of digital money that is designed to be secure and, in many cases, anonymous. It is a currency associated with the internet that uses cryptography, the process of converting legible information into an almost unbreakable code, to track purchases and transfers. Cryptography was born out of the need for secure communication in the Second World War. It has evolved in the digital era with elements of mathematical theory and computer science to become a way to secure communications, information and money on-line [4].

Cryptocurrencies use decentralized technology to let users make secure payments and store money without the need to use their name or go through a bank. They run on a distributed public ledger called blockchain, which is a record of all transactions updated and held by currency holders. Cryptocurrencies and applications of blockchain technology are still nascent in financial terms and more uses should be expected. Transactions including bonds, stocks and other financial assets could eventually be traded using the technology.

Units of cryptocurrency are created through a process called mining, which involves using computer power to solve complicated maths problems that generate coins. Users can also buy the currencies from brokers, then store and spend them using cryptographic wallets. The first cryptocurrency was bitcoin, which was created in 2009 and is still the best known. There has been a proliferation of cryptocurrencies in the past decade and there are now more than 1,200 available on the Internet. Bitcoin soared as high as $20,000 at the end of last year before crashing back to less than $8000 now.

1.2.1 Distributed Ledger

A distributed ledger (also called a shared ledger, or referred to as distributed ledger technology) is a consensus of replicated, shared, and synchronized digital data geographically spread across multiple sites, countries, or institutions [5]. There is no central administrator or centralized data storage.

A peer-to-peer network is required as well as consensus algorithms to ensure replication across nodes is undertaken [6]. One form of distributed ledger design is the blockchain system, which can be either public or private. But not all distributed ledgers have to necessarily employ a chain of blocks to successfully provide secure and valid achievement of distributed consensus: a blockchain is only one type of data structure considered to be a distributed ledger [7]. In 2016, numerous banks tested distributed ledgers for international payments.

1.2.2 Blockchain

The blockchain is a public ledger that records bitcoin transactions [8]. A novel solution accomplishes this without any trusted central authority: the maintenance of the blockchain is performed by a network of communicating nodes running bitcoin software [9]. Transactions of the form payer X sends Y bitcoins to payee Z are broadcast to this network using readily available software applications. Network nodes can validate transactions, add them to their copy of the ledger, and then broadcast these ledger additions to other nodes. The blockchain is a distributed database – to achieve independent verification of the chain of ownership of any and every bitcoin amount, each network node stores its own copy of the blockchain [10]. Approximately six times per hour, a new group of accepted transactions, a block, is created, added to the blockchain, and quickly published to all nodes. This allows bitcoin software to determine when a particular bitcoin amount has been spent, which is necessary in order to prevent double-spending in an environment without central oversight. Whereas a conventional ledger records the transfers of actual bills or promissory notes that exist apart from it, the blockchain is the only place that bitcoins can be said to exist in the form of unspent outputs of transactions.

1.2.3 Cryptocurrency Exchange

Cryptocurrency exchanges are online platforms where you can exchange one cryptocurrency for another cryptocurrency. In other words, depending on the exchange, it is either like a stock exchange or a currency exchange [16].

Traditional Cryptocurrency Exchanges

These are the exchanges that are like the traditional stock exchanges where buyers and sellers trade based on the current market price of cryptocurrencies. These type of trading platforms generally charge a fee for each transaction. Some of these types of exchanges deal only in cryptocurrency, others allow users to trade fiat currencies like the U.S. dollar for cryptocurrencies like Bitcoin [16]. Coinbase's GDAX is an example of this type of exchange, as is Kraken. Of exchanges, there are those run by third parties and decentralized exchanges and peer-to-peer exchanges.

Cryptocurrency Brokers

These are website-based exchanges, which allow customers to buy and sell cryptocurrencies at a price set by the broker [16]. Coinbase is an example of this type of exchange. Shapeshift provides a similar service as well. This is the simplest solution for new users, since it is simple and easy, but someone will pay slightly higher prices than on the exchanges [16].

Direct Trading Platforms

These platforms offer direct peer-to-peer trading between buyers and sellers [17], but do not use an exchange platform like GDAX . Direct trading platforms of this type do not use a fixed market price. Sellers set their own exchange rate and buyers either find sellers via the platform and preform an Over the Counter (OTC) exchange [16], or they denote the rates they are willing to buy for and the platform matches buyers and sellers. This solution is hardly ever the best one, but it can be the only solution in some regions.

Cryptocurrency Funds

Funds are pools of professionally managed cryptocurrency assets which allows public buy and hold cryptocurrency via the fund [17]. One such fund is GBTC. Using a fund you can invest in cryptocurrency without having to purchase or store it directly.

1.3 Bitcoin

FIGURE 1.1: Bitcoin Logo

Bitcoin is a digital currency created in 2009. It follows the ideas set out in a white paper by the mysterious Satoshi Nakamoto, whose true identity has yet to be verified. Bitcoin offers the promise of lower transaction fees than traditional online payment mechanisms and is operated by a decentralized authority, unlike government-issued currencies. Today's market cap for all bitcoin (abbreviated BTC or, less frequently, XBT) in circulation exceeds $135 billion [11].

There are no physical bitcoins, only balances kept on a public ledger in the cloud, that – along with all Bitcoin transactions – is verified by a massive amount of computing power. Bitcoins are not issued or backed by any banks or governments, nor are individual bitcoins valuable as a commodity. Despite its not being legal tender, Bitcoin charts high on popularity, and has triggered the launch of other virtual currencies collectively referred to as Altcoins.

Bitcoin is one of the first digital currencies to use peer-to-peer technology to facilitate instant payments. The independent individuals and companies who own the governing computing power and participate in the Bitcoin network, also known as "miners", are motivated by rewards (the release of new bitcoin) and transaction fees paid in bitcoin. These miners can be thought of as the decentralized authority enforcing the credibility of the Bitcoin network. New bitcoin is being released to the miners at a fixed, but periodically declining rate, such that the total supply of bitcoins approaches 21 million. One bitcoin is divisible to eight decimal places (100 millionth of one bitcoin), and this smallest unit is referred to as a Satoshi. If necessary, and if the participating miners accept the change, Bitcoin could eventually be made divisible to even more decimal places [11].

Bitcoin mining is the process through which bitcoins are released to come into circulation. Basically, it involves solving a computationally difficult puzzle to discover a new block, which is added to the blockchain, and receiving a reward in the form of few bitcoins. The block reward decreases every four years. As more and more bitcoins are created, the difficulty of the mining process – that is, the amount of computing power involved – increases.

1.4 Altcoins

FIGURE 1.2: Some Altcoins Logos

Altcoins are the alternative cryptocurrencies launched after the success of Bitcoin. Generally, they project themselves as better substitutes to Bitcoin. The success of Bitcoin as the first peer-to-peer digital currency paved the way for many to follow. Many altcoins are trying to target any perceived limitations that Bitcoin has and come up with newer versions with competitive advantages. There is a great variety of altcoins. This research will focus on Ethereum and Ripple.

1.4.1 Ethereum

FIGURE 1.3: Ethereum Logo

Launched in 2015, Ethereum is a decentralized software platform that enables SmartContracts and Distributed Applications (DApps) to be built and run without any downtime, fraud, control or interference from a third party. Ethereum is not just a platform but also a programming language (Turing complete) running on a blockchain, helping developers to build and publish distributed applications. The potential applications of Ethereum are wide ranging [12].

Smart contract is just a phrase used to describe computer code that can facilitate the exchange of money, content, property, shares, or anything of value. When running on the blockchain a smart contract becomes like a self-operating computer program that automatically executes when specific conditions are met. Because smart contracts run on the blockchain, they run exactly as programmed without any possibility of censorship, downtime, fraud or third party interference [13].

Like Bitcoin, Ethereum is a distributed public blockchain network. Although there are some significant technical differences between the two, the most important distinction to note is that Bitcoin and Ethereum differ substantially in purpose and capability. Bitcoin offers one particular application of blockchain technology, a peer to peer electronic cash system that enables online Bitcoin payments. While the Bitcoin blockchain is used to track ownership of digital currency (bitcoins), the

Ethereum blockchain focuses on running the programming code of any decentralized application.

In the Ethereum blockchain, instead of mining for bitcoin, miners work to earn Ether, a type of crypto token that fuels the network. Beyond a tradeable cryptocurrency, Ether is also used by application developers to pay for transaction fees and services on the Ethereum network [14].

1.4.2 Ripple

FIGURE 1.4: Ripple Logo

One of the most intriguing in its diversion from several crypto-norms is Ripple, a much more centralized cryptocurrency in a very decentralized space. Ripple is still classed as a cryptocurrency, but the way it was founded and the way it operates are very different from some of the others out there [15]. That's why if you're thinking of investing in Ripple, you need to do your research first and what better place to start, than right here.

Ripple is the catchall name for the cryptocurrency platform, the transactional protocol for which is actually XRP, in the same fashion as Ethereum is the name for the platform that facilitates trades in Ether. Like other cryptocurrencies, Ripple is built atop the idea of a distributed ledger network which requires various parties to participate in validating transactions, rather than any singular centralized authority. That facilitates transactions all over the world, and transfer fees are far cheaper than the likes of bitcoin. Unlike other cryptocurrencies, XRP transfers are effectively immediate, requiring no typical confirmation time [15].

Ripple was originally founded by a single company, Ripple Labs, and continues to be backed by it, rather than the larger network of developers that continue bitcoin's development. It also doesn't have a fluctuating amount of its currency in existence. Where bitcoin has a continually growing pool with an eventual maximum, and Ethereum theoretically has no limit, Ripple was created with all of its 100 billion XRP tokens right out of the gate. That number is maintained with no mining and most of the tokens are owned and held by Ripple Labs itself — around 60 billion at the latest count [15].

Chapter 2

Literature review

2.1 Time Series

Time series prediction is not a new phenomenon. Forecast of financial markets such as the stock market has been researched at length. Speculators, stock market traders, market participants or simply traders are for the most part terms used to portray people and associations who endeavor to bring home the bacon from purchasing and offering various financial assets in an immense scope of markets around the globe. Plainly the ability to forecast the direction of market movements, up or down, is vital to these individuals and entities. To this end a wide assortment of strategies and techniques have been attempted and utilized by the members in the market. Further, finished the most recent couple of decades academics have demonstrated an enthusiasm for this field and endeavored to evaluate and justify the wide assortment of methods utilized [18].

Cryptocurrencies present an interesting parallel to this as they are time series prediction problems in a market still in its transient stage. As a result, there is high volatility in the market [19] and this provides an opportunity in terms of prediction.

2.1.1 Analysis

The study of analyzing and forecasting time series data has been an active area of study for several decades [20]. Series data is ordered such that the ordering is an important if not critical aspect of the data and the requirement to maintain this ordering enforces certain constraints on its processing. Series data can be ordered by factors such as distance or height but typically time is the ordering encountered, and thus such collections are referred to as time series [18]. Analysis of time series data is found in a wide range of areas including, Sales Forecasting, Speech Recognition, Economic Forecasting, Stock Market Analysis, Process and Quality Control and Seismic Recordings.

In general, with non-series data we are interested in the relationships between the attributes of any particular row of data and perhaps how they affect the parameter we are interested in. Frequently, some kind of regression technique is used in this kind of analysis in order to answer questions such as how is rainfall in an area affected by altitude or how does fuel consumption vary with car engine size [21]. However with time series data there is an additional consideration, the relationship between the attribute's current value to that of its previous or later values. This is known as auto-correlation [22]. Typically, interest in financial data is found in their previous values. This is because it is interesting to be able to understand and answer how the present value is affected by past pre-day values.

A time series can contain some or all of the following components:

1. Trend - the overall direction of the series, is it increasing or decreasing over time?
2. Seasonality - regular variations in the time series that is caused by re-occurring events, for example a spike in sales during the Christmas period [23].
3. Random component - additional fluctuations in the series that may be attributed to noise or other random events.

There are three primary types of time series, stationary, additive and multiplicative. Stationary series have constant amplitude without a trend element. Often stationary time series are repetitive, in other words showing constant auto-correlation and are considered the easiest type to model [18]. The second type of time series is the additive type. In this type all three components of the series are present, trend, seasonality and noise. The distinguishing feature here is the amplitude of the seasonal component in that it is quite regular being static over time. This time series is trending upwards overall but there is a clear repetitive pattern of peaks and troughs caused by the seasonality, with the heights of the peaks all being similar. Multiplicative time series are similar to the additive version except the amplitude of the seasonality increases over time. Financial time series can be considered as containing all three elements of a time series [18].

According to the research developed in this field, we can classify the techniques used to solve the cryptocurrency market prediction problems to two folds:

- Econometric Models: These are statistical based approaches such as auto - regression family of models. There are number of assumptions need to be considered while using these models such as linearity and stationary of the financial time-series data. These non-realistic assumptions can degrade the quality of prediction results [24] [25].
- The computational models used include Artificial Neural Networks (ANN) and their variants [26], Support Vector Machines (SVM) [27], and many others.

2.2 Algorithms

Time series modeling exists in two classes, namely parametric and nonparametric [28]. A comprehensive classification of various time series prediction methods are available in [28] and a summary is given in Figure 2.1 below. Many early time series prediction methods were parametric approaches where the goal was to fit a statistical model to time series data. However with the identification of severe limitations in simple parametric approaches, people were more interested on nonparametric approaches in which no structural assumptions are made about the underlying structure of the process [29].

Artificial intelligence has become a perfect candidate for time series prediction in situations where manual data analysis is not applicable and relationships among different entities are not obvious. Machine learning is the branch in Artificial Intelligence where different algorithms are developed in order to allow computers to learn. Machine learning is said to be the domain of computational intelligence that develops algorithms which automatically improves themselves with experience. Various machine learning algorithms have been developed to perform time series prediction.

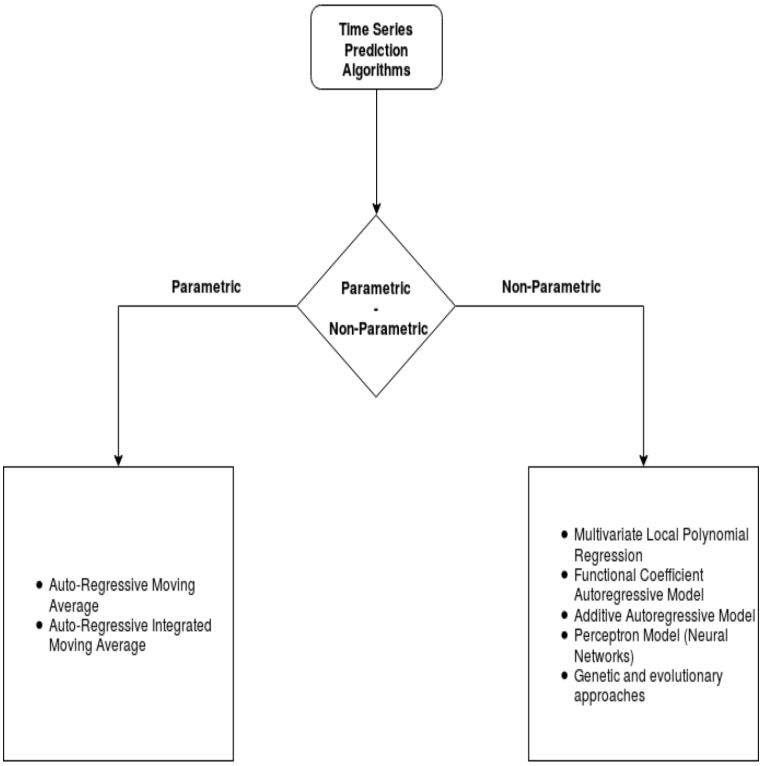

FIGURE 2.1: Classes of Time Series Prediction Methods

2.2.1 Regression Models

Regression is the study of the impact of known variables (independent) on an unknown (dependent) variable and addresses questions such as how does a person's income vary with their years of education and working experience. The general equation for linear regression is given by:

$$y = \alpha + \beta x + \epsilon$$

Interpretation:

- α is the intercept.
- β is the coefficient.
- x is the independent variable.
- ϵ is the error term.

In reality, there are often a large number of independent variables that affect the unknown under study and thus multiple regression, shown below, is usually of interest.

$$y_i = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_n x_{ni} + \epsilon$$

alternatively :

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_n x_{ni} + \epsilon$$

Auto-Regression (AR)

In a time series the preceding values often have a bearing on the current data point, and this is especially important in financial time series data. Thus auto-regression (AR) is the prediction of the current point from the use of previous values of the data point itself [18], and is given by:

$$y_t = \alpha_0 + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + ... + \alpha_p y_{t-p} + \epsilon$$

Interpretation:

- α_0 is the intercept, is often zero and the mean of the time series.
- $\alpha_1 - \alpha_p$ are the coefficients of model.
- $y_{t-1} - y_{t-p}$ are the previous values.
- ϵ is the random noise.

An auto-regressive model is often represented as $AR(p)$. This means that current values depend on its own p-previous values. The p variable denotes the order of AR process.

Moving-Average (MA)

In addition to auto-regressive models, in the time series, there are also moving-average (MA) models, where often the current deviation from mean depends on previous deviations. The moving-average (MA) model's equation is :

$$m_t = \beta_0 + \beta_1 m_{t-1} + \beta_2 m_{t-2} + ... + \beta_p m_{t-p} + \epsilon$$

Interpretation:

- β_0 is the intercept, is often zero and the mean of the time series.
- $\beta_1 - \beta_p$ are the coefficients of model.
- $m_{t-1} - m_{t-p}$ are the previous deviations.
- ϵ is the random noise.

Auto-Regressive Moving Average (ARMA)

The auto-regressive moving average (ARMA) model, also known as Box-Jenkins [30], combines moving-averages with auto-regression. $ARMA$ combines the moving-average process with auto-regressive process to generate a model utilizing both terms:

$$y_t = c + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + ... + \alpha_p y_{t-p}$$
$$-\beta_1 m_{t-1} - \beta_2 m_{t-2} - ... - \beta_p m_{t-p} + \epsilon$$

Interpretation:

- c is the intercept, is often zero and the mean of the time series.
- $\alpha_1 - \alpha_p$ are the coefficients of the auto-regression process.
- $\beta_1 - \beta_p$ are the coefficients of the moving average process.
- $y_{t-1} - y_{t-p}$ are the previous values.
- $m_{t-1} - m_{t-p}$ are the previous deviations.
- ϵ is the random noise.

Here, the MA parameters are defined so that their signs are negative in the formula, following the convention introduced by Box and Jenkins. Some authors and software define them so that they have plus signs instead. However, when actual numbers are plugged into the equation, there is no ambiguity.

An ARMA(p,q) model uses the previous p values in the auto-regression term and the moving averages derived from the last q values. There are therefore three steps in developing an ARMA model [18] :

1. identification step in which the order of AR and MA components is determined
2. parameters / coefficients estimation
3. forecasting

ARMA models have certain inborn properties that might be thought about downsides, namely the requirement for the time series to be stationary with no trend and also linear and the difficulty in deriving the correct parameters to use in the model. To beat these confinements researchers have attempted various ways to deal with upgrade the effectiveness of ARMA models [18].

Auto-Regressive Integrated Moving Average (ARIMA)

One limitation with the ARMA model and indeed other approaches is that it is assumed that the time series is stationary, it does not have trend and has constant variance and mean [31]. In reality of course many time series data sets have trend, and in the world of financial data this is also true. With a specific end goal to represent trend in a time series it is frequently changed into a stationary data set, modeling is then performed on this adjusted data after which it is come back to its original state. In actuality the trend aspect is removed, modeling is done, at that point the trend component is included back into the data.

One such technique for eliminating trend is differencing [22]. Differencing is the technique of replacing the actual values of the observations with the values of the differences between them. This is represented as:

$$\Delta_t^{(1)} = y_t - y_{t-1}$$

Differencing is the same as considering the derivative of the series, thus a time series that has under gone differencing is considered “integrated” [18]. On the off chance that taking this purported first difference does not expel the trend one can go further and utilize the second difference:

$$\Delta_t^{(2)} = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2})$$

Addition of an integration step to the ARMA model results in an auto-regressive integrated moving average (ARIMA) model. $ARIMA$ can handle any series, with or without seasonal elements.

$ARIMA$ mathematical formulation:

$$\begin{gathered} y_t = c + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + ... + \alpha_p y_{t-p} + \\ + \beta_1 m_{t-1} + \beta_2 m_{t-2} + ... + \beta_p m_{t-p} + \\ d_1 \Delta_{t-1}^{(1)} + d_2 \Delta_{t-2}^{(2)} + ... + d_d \Delta_{t-d}^{(d)} + \epsilon \end{gathered}$$

Interpretation:

- c is the intercept, is often zero and the mean of the time series.
- $\alpha_1 - \alpha_p$ are the coefficients of the auto-regression process.
- $\beta_1 - \beta_p$ are the coefficients of the moving average process.
- $d_1 - d_p$ are the coefficients of the differencing term.
- $y_{t-1} - y_{t-p}$ are the previous values.
- $m_{t-1} - m_{t-p}$ are the previous deviations.
- ϵ is the random noise.

ARIMA models are typically referenced as ARIMA(p,d,q) with p the quantity of terms utilized as a part of the auto-regression process, d the quantity of differencing terms and q the quantity of terms utilized as a part of the moving-average [18].

An important aspect of building time series models with ARIMA techniques is the choice of parameters to use. Autocorrelation (AC) and partial autocorrelation (PAC) are important measures in the selection process of these parameters [22]. Autocorrelation (AC) is the linear dependence of a variable with itself at two points in time. For stationary processes, autocorrelation between any two observations only depends on the time lag h between them. Define $Cov(y_t, y_{t-h}) = \gamma_h$. Lag-$h$ autocorrelation is given by

$$p_h = Corr(y_t, y_{t-h}) = \frac{\gamma_h}{\gamma_0}$$

The denominator γ_0 is the lag 0 covariance, which means the unconditional variance of the process. Correlation between two variables can result from a mutual linear dependence on other variables. Partial autocorrelation (PAC) is the autocorrelation between y_t and y_{t-h} after removing any linear dependence on $y_1, y_2, ..., y_{t-h+1}$. The partial lag-h autocorrelation is denoted $\phi_{h,h}$.

The autocorrelation function (ACF) for a time series y_t, $t = 1, ..., N$ is the sequence p_h, $h = 1, 2, ..., N-1$. The partial autocorrelation function (PACF) is the sequence

$\phi_h, h, h = 1, 2, ..., N - 1$. The theoretical ACF and PACF for the AR, MA, and ARMA conditional mean models are known, and quite different for each model. The differences in ACF and PACF among models are useful when selecting models. The following table 2.1 summarizes the ACF and PACF behavior for these models.

Conditional Mean Model	ACF	PACF
AR(p)	Tails of gradually	Cuts off after p lags
MA(q)	Cuts off after q lags	Tails off gradually
ARMA(p, q)	Tails off gradually	Tails off gradually

TABLE 2.1: ACF & PACF

2.2.2 Support Vector Machines

Support vector machine is a powerful supervised learning model for prediction and classification. SVM was first introduced by Vladimir Vapnik and his co-workers at AT&T Bell Laboratories [32]. The fundamental idea of SVM is to map the training data into higher dimensional space using a nonlinear mapping function and then perform "linear" regression in higher dimensional space in order to separate the data [33]. For data mapping purposes is utilized a predetermined kernel function. Data separation is done by finding the optimal hyperplane. This optimal hyperplane is called the Support Vector with the maximum margin from the separated classes.

Training phase

Suppose we have a data set $\{x_i, y_j\}_{i=1,..,n}$ where the input vector $x_i \in \Re^d$ and the actual $y_i \in \Re$ [34]. The modeling goal of SVM is to find the linear decision function represented in the following equation :

$$f(x) \leq w, \quad \phi_i(x) > +b$$

Interpretation:

- w is the weight vector, which have to be estimated from the dataset.
- b is a constant, which have to be estimated from the dataset.
- ϕ is a nonlinear mapping function.

The regression problem can be formulated as to minimize the following regularized risk function:

$$R(C) = \frac{C}{n} \sum_{i=1}^{n} L_\epsilon(f(x_i), y_i) + \frac{1}{2} \|w\|^2$$

$L_\epsilon(f(x_i), y_i)$ is known as ϵ-intensive loss function and given by the following equation:

$$L_\epsilon(f(x_i), y_i) = \begin{cases} |f(x) - y| - \epsilon & \text{if } |f(x) - y| \geq \epsilon \\ 0 & \text{otherwise} \end{cases}$$

Slack variables can be utilized, ζ_i and ζ_i^*, in order to accomplish an acceptable degree of miss classification error. So now, there seems to be a constrained minimum optimization problem, as this addition has occurred.

$$minR(w, \zeta_i^*) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} (\zeta_i + \zeta_i^*)$$

The prediction error might have a penalty, denoted as C, that is greater than ϵ. The slack variables, ζ_i and ζ_i^*, form the separation from actual values to the corresponding boundary values of ϵ [34]. The objective of SVM is to minimize ζ_i , ζ_i^* and $\|w\|^2$. The above optimization with constraint can be changed over by methods for Lagrangian multipliers to a quadratic programming problem. Therefore, the form of the solution can be given by the following equation:

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(x_i, x) + b$$

K is the kernel function and its values is an inner product of two vectors x_i and x_j in the feature space $\phi(x_i)$ and $\phi(x_j)$ and satisfies the Mercer's condition. Therefore,

$$K(x_i, x_j) = \phi(x_i)\phi(x_j)$$

In Table 2.2 there are some common kernels used with SVM.

Kernel	formula
Polynomial	$K(x_i, x_j) = (x_i x_j + 1)^d$
Gaussian	$K(x, y) = exp(-\frac{\|x-y\|^2}{2\sigma^2})$
Gaussian radial basis function	$K(x_i, x_j) = exp(-\gamma \left\|x_i - x_j\right\|^2)$
Laplace radial basis function	$K(x, y) = exp(-\frac{\|x-y\|}{\sigma})$
Hyperbolic tangent	$K(x_i, x_j) = tanh(k x_i x_j + c)$
Anova radial basis	$K(x, y) = \sum_{k=1}^{n} exp(-\sigma(x^k - y^k)^2)^d$

TABLE 2.2: Common SVM kernels

Support vector machines can be efficiently used in the field of time series prediction. A few researchers say that as a rule, SVMs have numerous advantages over classical classification approaches like artificial neural networks, decision trees and others. Good performance in high dimensional spaces can be considered as an advantage. Moreover, the support vectors depend on a little subset of the training data which gives SVM an awesome computational advantage.

2.2.3 Neural Networks

Artificial Neural Networks

Artificial Neural Networks (ANN) or neural networks are computational algorithms. They intended to simulate the behavior of biological systems composed of "neurons". ANNs are computational models inspired by an animal's central nervous systems. It is capable of machine learning as well as pattern recognition. These presented as systems of interconnected "neurons" which can compute values from inputs.

A neural network is an oriented graph. It consists of nodes which in the biological analogy represent neurons, connected by arcs. It corresponds to dendrites and synapses. Each arc associated with a weight while at each node. Apply the values received as input by the node and define activation function along the incoming arcs, adjusted by the weights of the arcs. The activation function of a node defines the output of that node given an input or set of inputs.

A neural network is a machine learning algorithm based on the model of a human neuron. The human brain consists of millions of neurons. It sends and process signals in the form of electrical and chemical signals. These neurons are connected with a special structure known as synapses. Synapses allow neurons to pass signals. From large numbers of simulated neurons neural networks forms.

An Artificial Neural Network is an information processing technique. It works like the way human brain processes information. ANN includes a large number of connected processing units that work together to process information. They also generate meaningful results from it. We can apply Neural network not only for classification. It can also apply for regression of continuous target attributes.

Neural networks find great application in data mining used in sectors. For example economics, forensics, etc and for pattern recognition. It can be also used for data classification in a large amount of data after careful training. A neural network may contain the following 3 layers:

- Input layer – The activity of the input units represents the raw information that can feed into the network.
- Hidden layer – To determine the activity of each hidden unit. The activities of the input units and the weights on the connections between the input and the hidden units. There may be one or more hidden layers.
- Output layer – The behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units.

Neural Network Layers

An Artificial Neural Network is typically organized in layers. Layers are being made up of many interconnected "nodes" which contain an "activation function". A neural network may contain the following 3 layers:

1. Input layer

   The purpose of the input layer is to receive as input the values of the explanatory attributes for each observation. Usually, the number of input nodes in an input layer is equal to the number of explanatory variables. The input layer presents the patterns to the network, which communicates to one or more hidden layers. The nodes of the input layer are passive, meaning they do not change the data. They receive a single value on their input and duplicate the value to their many outputs. From the input layer, it duplicates each value and sent to all the hidden nodes.

2. Hidden layer

   The Hidden layers apply given transformations to the input values inside the network. In this, incoming arcs that go from other hidden nodes or from input nodes connected to each node. It connects with outgoing arcs to output nodes or to other hidden nodes. In hidden layer, the actual processing is done via a system of weighted connections. There may be one or more hidden layers. The values entering a hidden node multiplied by weights, a set of predetermined numbers stored in the program. The weighted inputs are then added to produce a single number.

3. Output layer

   The hidden layers then link to an output layer. Output layer receives connections from hidden layers or from input layer. It returns an output value that corresponds to the prediction of the response variable. In classification problems, there is usually only one output node. The active nodes of the output layer combine and change the data to produce the output values. The ability of the neural network to provide useful data manipulation lies in the proper selection of the weights. This is different from conventional information processing.

Artificial Neural Networks are one of the dominant application in intelligent time series prediction [35]. The attempts to use neural networks for time series forecasting go back to 1970's. But at that time with the limited knowledge on neural networks the results from the experiments were not very promising. But with formation of back-propagation algorithm in 1986, research interests in neural networks had grown rapidly and successfully adapted for the problem of time series forecasting. Some of those results had shown that the neural networks are capable of outperforming statistical forecasting methods such as regression analysis and Box-Jenkins forecasting [36]. With the rise of cryptocurrencies, many researchers use the various types of neural networks to predict the price as well as other stock indices that concern Bitcoin, Ethereum and the various altcoins that exist.

Recurrent Neural Networks

Recurrent neural networks (RNNs) were developed in the 1980s. Hopfield networks were invented by John Hopfield in 1982. In 1993, a neural history compressor system solved a "Very Deep Learning" task that required more than 1000 subsequent layers in an RNN unfolded in time [37]. A recurrent neural network (RNN) is a class of artificial neural network where connections between units form a directed cycle. This allows it to exhibit dynamic temporal behavior. Unlike feedforward neural networks (FFNN), RNNs can use their internal memory to process arbitrary sequences of inputs.

In graph presented in figure 2.2, a chunk of neural network, A, looks at some input x_t and outputs a value h_t. A loop allows information to be passed from one step of the network to the next. A RNN can be thought of as multiple copies of the same network, each passing a message to a successor [38].

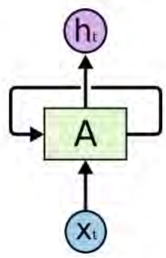

FIGURE 2.2: RNN-rolled [38]

The graph in figure 2.2 above can be expanded to :

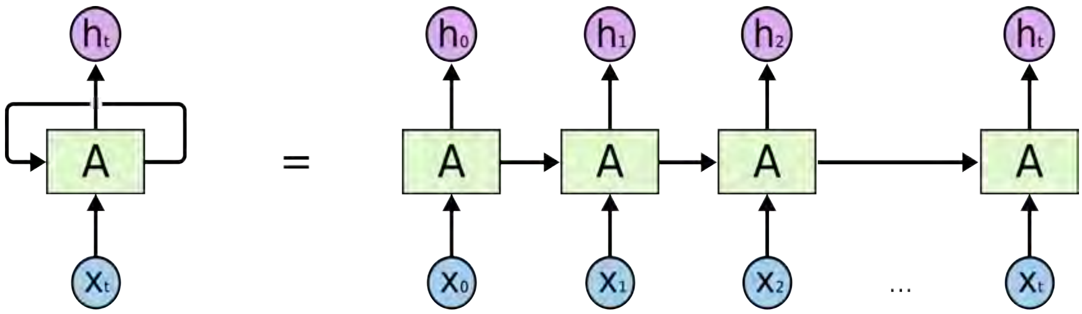

FIGURE 2.3: RNN-unrolled [38]

This chain-like nature reveals that RNNs are intimately related to sequences and lists. They are the natural architecture of neural network to use for such data. All RNNs have the form of a chain of repeating modules of neural network. The most basic RNN cell is a single layer neural network, the output of which is used as both the RNN cell's current (external) output and the RNN cell's current state:

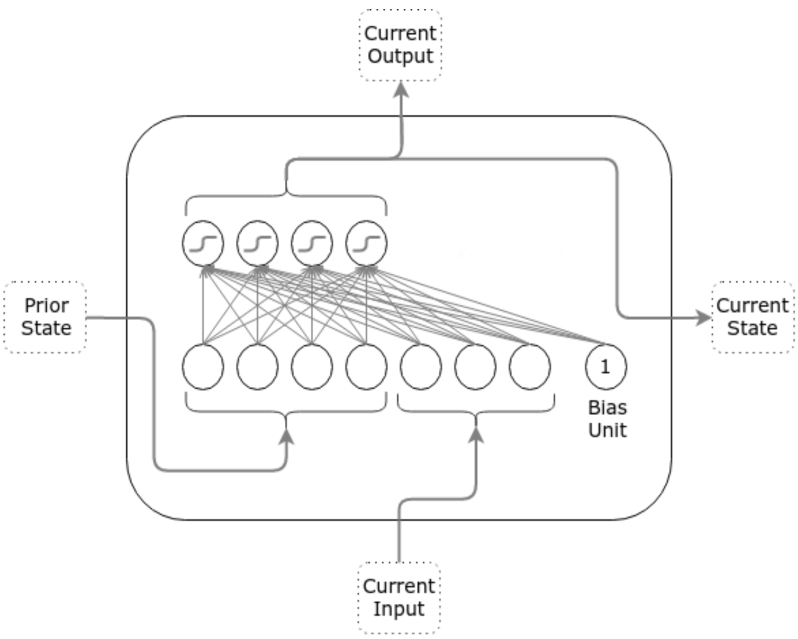

FIGURE 2.4: RNN-Cell [39]

Mathematical formulation of a normal RNN cell [39] :

$$s_t = \phi(Ws_{t-1} + Ux_t + b)$$

Interpretation:

- ϕ is the activation function.
- $s_t \in \Re^n$ is the current state and current output.
- $s_{t-1} \in \Re^n$ is the prior state.
- $x_t \in \Re^m$ is the current input.
- $W \in \Re^{n\times n}$,$U \in \Re^{m\times n}$ are the weights.
- $b \in \Re^n$ are the biases.
- n and m are the state and input sizes.

Long Short-Term Memory

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter & Schmidhuber in 1997 [40], and were refined and popularized by many people in following work. They work tremendously well on a large variety of problems, and are now widely used. LSTMs are explicitly designed to avoid the long-term dependency problem. It was invented to solve the vanishing gradient problem created by normal RNN. It is claimed that LSTMs are capable of remembering inputs with longer time steps. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn [38].

In LSTMs, everything is "written down" and, assuming no interference from other state units or external inputs, carries its prior state forward. This means that any state changes are incremental, so that

$$s_{t+1} = s_t + \Delta s_{t+1}$$

The fundamental challenge of LSTMs is that uncontrolled and uncoordinated writes, particularly at the start of training when writes are completely random, create a chaotic state that leads to bad results and from which it can be difficult to recover. According to the early literature on LSTMs, the key to overcoming the fundamental challenge of LSTMs and keeping the state under control is to be selective in three things: what they write, what they read, and what they forget (because obselete information is a distraction and should be forgotten). As a result LSTMs , need to have 3 mechanisms for selective writing, reading, and forgetting.

Selective reading, writing and forgetting involves separate read, write and forget decisions for each element of the state. These decisions will be made by taking advantage of state-sized read, write and forget vectors with values between 0 and 1 specifying the percentage of reading, writing and forgetting, for each state element [39]. These read, write and forget vectors are called "gates", and can be computed using the single layer neural network. The three gates at time step t are denoted i_t, the input gate (for writing), o_t, the output gate (for reading) and f_t, the forget gate (for remembering).

Mathematical formulation of gates :

$$i_t = \sigma(W_i s_{t-1} + U_i x_t + b_i)$$
$$o_t = \sigma(W_o s_{t-1} + U_o x_t + b_o)$$
$$f_t = \sigma(W_f s_{t-1} + U_f x_t + b_f)$$

Below there is a basic LSTM cell :

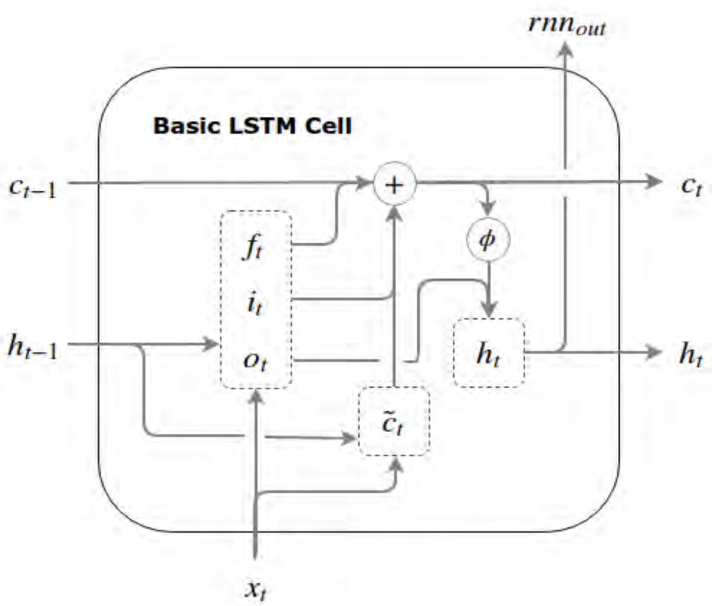

FIGURE 2.5: Basic LSTM cell [39]

Mathematical formulation of the cell :

$$i_t = \sigma(W_i s_{t-1} + U_i x_t + b_i)$$
$$o_t = \sigma(W_o s_{t-1} + U_o x_t + b_o)$$
$$f_t = \sigma(W_f s_{t-1} + U_f x_t + b_f)$$
$$\bar{c}_t = \phi(W h_{t-1} + U x_t + b)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \bar{c}_t$$
$$h_t = o_t \odot \phi(c_t)$$
$$rnn_{out} = h_t$$

Interpretation:

- ϕ is the activation function.
- i_t, o_t, f_t are the gate functions.
- $s_{t-1} \in \Re^n$ is the prior state.

- c_t is cell or constant error.
- $\bar{c}_t$ is the candidate cell to write.
- h_t is the hidden state.

The hidden state has the same size as the regular state. In addition , h_{t-1} is analogous to the gated prior state $o_t \odot s_{t-1}$, except that it is squashed by a non-linearity.Thus the prior state LSTM receives at time step t is a tuple of closely-related vectors: (c_{t-1}, h_{t-1}), where $h_{t-1} = o_{t-1} \odot \phi(c_{t-1})$. Gates are computed using the gated shadow state, $h_{t-1} = o_{t-1} \odot \phi(c_{t-1})$, instead of a squashed main state, $\phi(c_{t-1})$. The LSTM's external output is a gated shadow state, $h_t = o_t \odot \phi(c_t)$, instead of a squashed main state, $\phi(c_t)$. The above is a basic architecture of a LSTM recurrent neural network. But not all LSTMs are the same as the above. In fact, it seems like almost every paper involving LSTMs uses a slightly different version. The differences are minor [38].

Gated Recurrent Unit

Gated recurrent units (GRUs) are a gating mechanism in recurrent neural networks, introduced in 2014 [41]. It is a slightly more dramatic variation on the LSTM. It combines the forget and input gates into a single "update gate." It also merges the cell state and hidden state, and makes some other changes. The resulting model is simpler than standard LSTM models, and has been growing increasingly popular. One way to impose a hard bound on the state and coordinate the writes and forgets of each state is to explicitly link them. In other words, instead of doing selective writes and selective forgets, selective overwrites by setting the forget gate equal to 1 minus the write gate are done, so that:

$$s_t = (1 - i_t) \odot s_{t-1} + i_t \odot \bar{s}_t$$

The basic idea of using a gating mechanism to learn long-term dependencies is the same as in a LSTM, but there are a few key differences:

- A GRU has two gates, an LSTM has three gates.
- GRUs do not possess and internal memory (c_t) that is different from the exposed hidden state. They do not have the output gate that is present in LSTMs.
- The input and forget gates are coupled by an update gate z and the reset gate r is applied directly to the previous hidden state. Thus, the responsibility of the reset gate in a LSTM is really split up into both r and z.

To conform for the GRU terminology used in the literature, the overwrite gate called an update gate and labeled z_t. The update gate, z_t, is the same as the forget gate from LSTM, f_t, and the write gate is calculated by $1 - z_t$. The read gate (or reset gate) is denoted r_t.

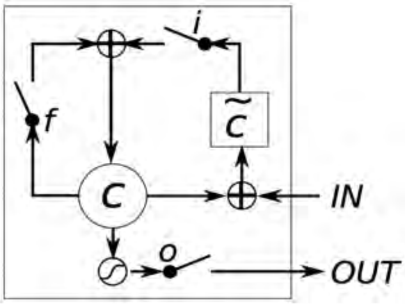

FIGURE 2.6: LSTM Gating [42]

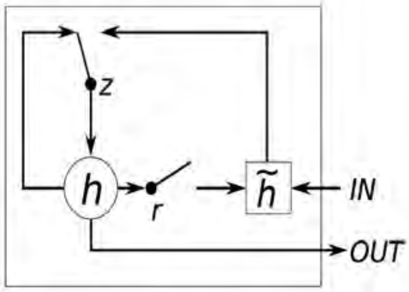

FIGURE 2.7: GRU Gating [42]

The above two figures show the gates of the LSTM and the GRU by contrasting them for a better understanding.

Below there is a GRU cell:

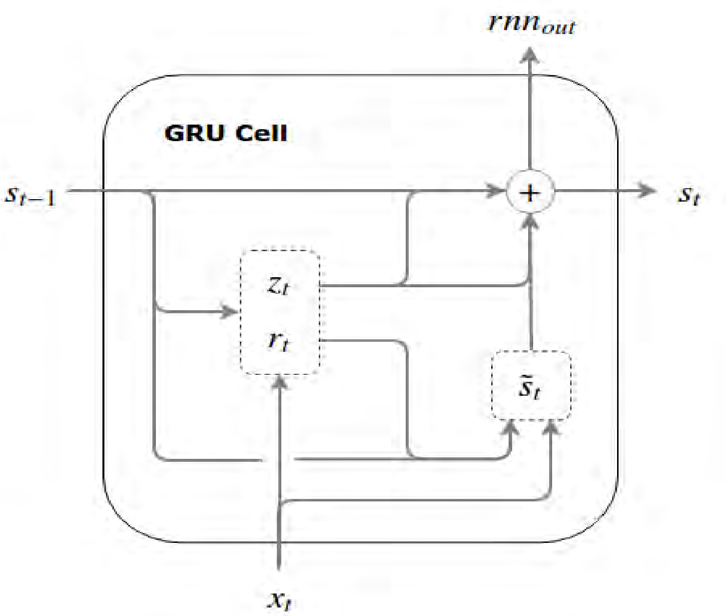

FIGURE 2.8: GRU cell [39]

Mathematical formulation of the GRU cell :

$$r_t = \sigma(W_r s_{t-1} + U_r x_t + b_r)$$
$$z_t = \sigma(W_z s_{t-1} + U_z x_t + b_z)$$
$$\bar{s}_t = \phi(W(r_t \odot s_{t-1}) + U x_t + b)$$
$$s_t = z_t \odot s_{t-1} + (1 - z_t) \odot \bar{s}_t$$

Chapter 3

Methodology

3.1 Data Collection

This research required data to draw conclusions as well as for subsequent price prediction models. These data had to be stock data with features like High, Low, Open, Close, Volume, Weighted Average. The search for these was done in Poloniex and Quandl. Poloniex is a US-based cryptocurrency exchange platform. Quandl is a platform for financial, economic, and alternative data that serves investment professionals. Quandl sources data from over 500 publishers. All Quandl's as well as Poloniex's data are accessible via an API. The data used in this study was freely collected from Poloniex and Quandl. It is of high quality with no missing values and represents the opening, high, low, closing, weighted average prices for each day that the particular market indice was open for trading. More specifically, data from Poloniex was received with a period of 300 seconds.

3.2 Data Description

In this subsection we describe the data used in the survey. In particular, we indicate which crypto-coins were used as well as from which exchanges the data was taken, explaining a little more.

3.2.1 Crypto-coins

Stock data was received for Bitcoin from Kraken exchange, as well as Bitstamp, Itbit, Coinbase. With respect to the remaining altcoins data obtained, it was data for Ethereum, Litecoin, Ripple, Ethereum Classic, Stellar, Dash, Siacoin, Monero, Zcash, Lisk, Stratis, NXT and NEM.

Exchanges

Kraken is a feature loaded trading platform with fast deposits and withdrawals for professional and advanced traders. It offers advanced order and trading tools - stop loss, stop loss limit, trailing stop, trailing stop limit, take profit limit, stop loss take profit limit, leverage, margin, etc. The exchange caters well to the needs of more sophisticated traders but that doesn't turn it unsuitable for beginners.

Coinbase has emerged as one of the high profile names in the Bitcoin world placing itself on the top as a one stop solution for Bitcoins - it is a wallet for storing, spending, buying and accepting Bitcoins, it acts as a Bitcoin processor (merchant tool) for many merchants and businesses, and is among the top Bitcoin exchanges.

Coinbase is a San Francisco, California based company and its initial offering was primarily restricted to the U.S. but now it has entered Europe providing, buying and selling services to around 18 countries.

Bitstamp, a Slovenia based Bitcoin exchange, is one of toppers in terms of trading volume. It facilitates instant buy-sell orders for USD/BTC pair with withdrawals and deposits available in currencies other than dollars as well. The exchange offers trading using limit orders where a pre-determined price can be set for buy and sell orders. Bitstamp has earned a strong reputation in the Bitcoin world. The two youngsters, Nejc Kodrič and Damian Merlak, who are behind Bitstamp, are to be commended for running the exchange very professionally with sound audits and regulatory compliance.

Itbit provides bitcoin trading services for financial institutions and active traders. Unlike other exchanges, it does not cater towards small-time traders or those who just make a small number of cryptocurrency transactions. Itbit is based in New York City. It is a regulated US financial services company. It is a professional bitcoin exchange designed for professional traders and institutions. The regulatory status of Itbit allows it to work with large financial institutions around the world.

3.2.2 Data Explanation

In this subsection we explain the data used in the survey. More specifically, the focus is on what data were used to explain each of them. These are basically the features used below in the data analysis section as well as in the machine learning section.

Opening Price

In the stock markets, open refers to the beginning of the trading day or the price of a security at the beginning of the trading day. The open is the start of a new day, though it is important to note that does not necessarily mean trading has not been going on right before the open. After-hours markets remain open as do other exchanges in other countries and time zones, which provides opportunity for the price to change right up until the open in many cases [43]. In the stock markets, opening price refers to the price of a security at the beginning of the trading day.

Closing Price

A closing price is the trading price of a security at the end of the trading day [44]. In cryptocurrency, it is the price at which a crypto-coin of property sells. The closing price does not necessarily mean the end of all trading on that security for the day. In fact, it simply means the floor of the exchange is closed. After-hours markets remain open, as do other exchanges in other countries and time zones, which provides opportunity for the price to change.

High & Low Price

A security's intraday low trading price. Today's low is the lowest price at which a stock trades over the course of a trading day. Today's low is typically lower than the opening or closing price [45].

A security's intraday high trading price. Today's high is the highest price at which a stock traded during the course of the day. Today's high is typically higher than the closing or opening price [46]. More often than not this is higher than the closing price.

Volume

Volume is the number of shares or contracts traded in a security or an entire market during a given period of time. For every buyer, there is a seller, and each transaction contributes to the count of total volume. That is, when buyers and sellers agree to make a transaction at a certain price, it is considered one transaction. Volume is an important indicator in technical analysis as it is used to measure the relative worth of a market move. If the markets make a strong price movement, then the strength of that movement depends on the volume for that period. The higher the volume during the price move, the more significant the move [47].

Base volume is the volume in terms of the first currency pair. Quote volume is the volume in terms of the second currency pair. For example, for BTC/USD, base volume would be BTC and quote volume would be USD.

Weighted Average

Weighted average is a mean calculated by giving values in a data set more influence according to some attribute of the data. It is an average in which each quantity to be averaged is assigned a weight, and these weightings determine the relative importance of each quantity on the average. Weightings are the equivalent of having that many like items with the same value involved in the average.

3.3 Methodology

At an early stage, data from Quandl and specifically from KRAKEN exchange for Bitcoin in USD (BTC / USD) was obtained through the python module quandl using its API. Then data from other exchanges were also obtained from ITBIT, COINBASE, BITSTAMP. Visualizations of the BTC/USD data and price comparisons between the different data sources were made. The data for Altcoins were then taken and visualizations of the weighted price and close price were made for Ethereum and Ripple cryptocurrencies. After this initial stage, there is an idea of how data values fluctuate. All these will be shown and explained below in the Data analysis section. The values of the data will be analyzed, will be compared and the predictive effort will be made.

At a later stage, the data from Altcoins, mainly the weighted average price and closing price, were compared with Bitcoin, through time series visualizations. In this way it was understood what position the Bitcoin holds in relation to all other cryptocurrencies. Statistical analysis then begins, starting with a correlation analysis. Links and correlations between the various crypto-coins with respect to their price and its fluctuations for the years 2015 to 2018, as they are formed, have been found and analyzed. Visualizations of correlation matrices have been made with heatmaps for better interpretation and results extraction. Focusing on 2017, bitcoin, ethereum and ripple with boxplots, histograms, and some trading strategies for this year were analyzed.

Proceeding at the machine learning stage, predictive models were built to predict the "Close" price of Ethereum (ETH/USD). The three main models tested and compared were based on Convolutional Neural Network (CNN), Long Short-Term

Memory (LSTM) and Gated Recurrent Unit (GRU) neural networks. These models were constructed using Keras, with Tensorflow backend. These models have been evaluated according to statistical measures and performance metrics. Finally, results and conclusions were drawn, according to which a Ethereum real-time forecast API was made. The previous one predicts the closing price of ETH / USD in 80 minutes in the future, accurately a forecast every 5 minutes.

Predictive Analytics Work-flow

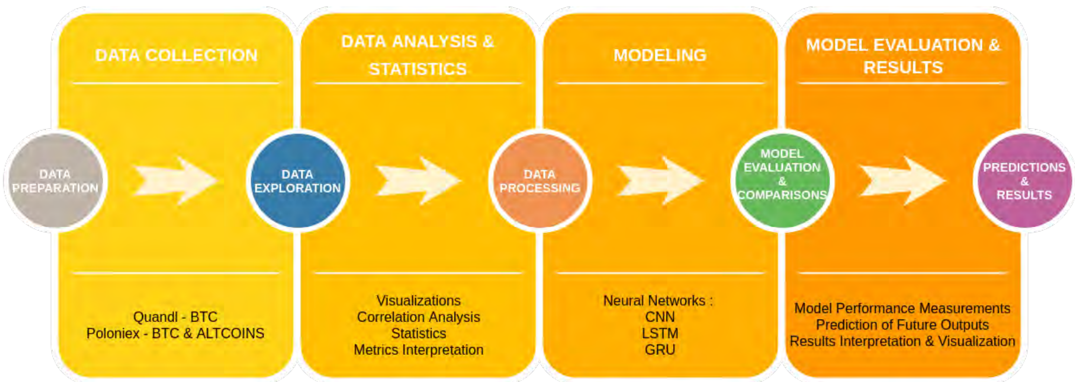

FIGURE 3.1: Predictive Analytics

3.4 Software Tools

This section describes and analyzes the software tools used to carry out this research work.

3.4.1 Python & Anaconda Environment

Python is a very powerful programming language used for many different applications. Over time, the huge community around this open source language has created quite a few tools to efficiently work with Python. In recent years, a number of tools have been built specifically for data science. As a result, analyzing data with Python has never been easier.

Anaconda is a free open source distribution of the Python and R programming languages for large-scale data processing, predictive analytics, and scientific computing, that aims to simplify package management and deployment. The main Python libraries used were NumPy and SciPy, which are present in the Anaconda installation package. SciPy is an open source Python library used for scientific computing and technical computing. NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. SciPy builds on the NumPy array object and is part of the NumPy stack which includes tools like Matplotlib, pandas and SymPy, and an expanding set of scientific computing libraries.

Most of the implementation was done in python and specifically in Jupyter Notebook, in the anaconda environment.

Keras & Tensorflow

Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result with the least possible delay is key to doing good research. Keras is compatible with: Python versions 2.7-3.6.

Keras is a deep learning library that:

- Allows for easy and fast prototyping (through user friendliness, modularity, and extensibility).
- Supports both convolutional networks and recurrent networks, as well as combinations of the two.
- Runs seamlessly on CPU and GPU.

TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API. TensorFlow was originally developed by researchers and engineers working on the Google Brain Team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well.

Keras with backend TensorFlow, was used to create CNN, LSTM, GRU neural networks, which were used for predictive modeling.

3.4.2 R & Rstudio

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

Experimental results and graphs were produced with the open source programming language R. For help in the creation and organisation of the R code for this thesis the open-source development environment R Studio was used .

3.4.3 RapidMiner

RapidMiner is a data science software platform developed by the company of the same name that provides an integrated environment for data preparation, machine learning, deep learning, text mining, and predictive analytics. Rapid Miner, a market leading open-source data mining and predictive analytic platform, was used for building and testing multiple predictive models.

Chapter 4

Data Analysis

In this section, we analyze the stock data of bitcoin, ethereum, ripple, and more altcoins. In addition, a statistical analysis of the data and comparisons will be made between these data. The analysis focused on the price of cryptocurrencies.

The price of a cryptocurrency is a reflection of its value. The more useful it is, the higher the demand for it will be. That request is what drives the value about whatever one coin dependent upon. However, internal factors are not the only ones affecting the price. Speculative pressure, exerted by traders who buy cryptocurrencies only to sell them later, is an external factor which may affect the price of a coin regardless of its actual usefulness. Oftentimes, this pressure is negative - when the traders start selling, the price goes down.

4.1 Bitcoin (BTC)

In 2009, the year Bitcoin was created, no one could have thought it would gain such reputation. The initial price was about $0.001. Over the subsequent five years there had been no tremendous activities, so the price rose slowly with little fluctuations. In 2013, Bitcoin attracted quite a few attention due to the Cyprus banking crisis. In November 2013, the Chinese began to buy BTC in massive portions and the price increased nearly 1000 percent. However, it did not last long. In February 2014, there was a DDoS attack on the servers of the exchange Mt. Gox. At that time, it managed approximately 60 percentage of Bitcoin transactions. This occasion struck Bitcoin hard and the price fell by 40 percent. Throughout the year, until January 2015, the price continued to fall down.

At some stage in 2015, Bitcoin received popularity slowly but firmly and at some point of that 12 months, the price started to upward thrust little by little. Since May 2016, Bitcoin has earned the trust of more and more people, and its price has persisted to rise. In June 2017, the market saw a dramatic fall and the price decreased by 14 percent. After rallying for most of the second half of 2016 Bitcoin breaches the $1000 mark for the first time in 3 years, in January 2017. Continuing, there is a sharp rise until August 2017, but it did not last much, as in September 2017 there was an almost vertical fall in price. From this point onwards, in general, there is a tremendous upward trend in the price, reaching $20,000 in December 2017. This did not last long, as despite the high price fluctuation seen in the transition from December 2017 to January 2018, Bitcoin's price is plummeting several thousand dollars. During this time the BTC/USD exchange is trying to keep the price, but this is not the case as the price is constantly decreasing until February 2018. Its price as it stands at the end of February 2018 is about $9.500. So, within a period of almost two months from December, 2017 to February 2018, an exceptional reduction of about 52.5 percent is omitted.

4.1.1 Price influencing factors

There are a number of factors that impact Bitcoin price over the course of time.

- **Market demand and supply** - This factor is major. Nowadays, Bitcoin does not have any physical equivalent in the real world, so BTC are sold on exchanges. The main principle of economics says that if people buy a currency, its price rises and if people sell the currency, its price falls. Bitcoin is no exception. In the fall of 2013, the price went up by 10 times as a result of Chinese demand.
- **Total amount of Bitcoins and Bitcoin holders** - The total amount of Bitcoins is 21 mln, but they are produced with time. Currently, there are about 17 mln BTC and more than 14 mln people have wallets with BTC. This number is growing rapidly and since the number of Bitcoins is fixed, the price will continue to rise.
- **News in mass media** - There is always a human factor involved - i.e. the way people can react to news.
- **Technical Issues** - Bitcoin has an open source code, so everyone can examine it. New updates for fixing some bugs and weak points in code can give an impetus for price growth. Meanwhile, successful account hacks or server attacks can bring down the exchange rate.
- **Political and economic events worldwide** - In the age of globalization, decisions in just one country can have an influence on the entire world.
- **High volatility** - Volatility is the degree of trading price variation over time. Volatility refers to the amount of uncertainty or risk in a security's value. Higher volatility means that a security's value can potentially be spread out over a larger range of values. In other words, the price of the security can change dramatically over a short time period in either direction.

All of the above are metaphorically summarized in the chart below, which shows the average weighted price of BTC / USD as it is formed by the price of four exchanges of Kraken, Itbit, Coinbase, Bitstamp.

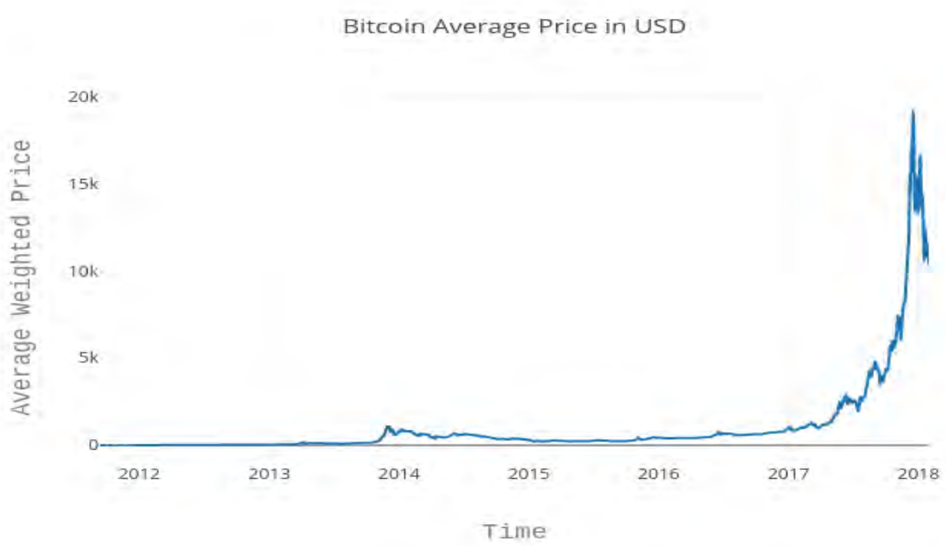

FIGURE 4.1: BTC/USD Weighted Price

4.2 Altcoins

Most altcoins cannot be bought directly with USD; to acquire these coins individuals often buy Bitcoins and then trade the Bitcoins for altcoins on cryptocurrency exchanges. For this reason, we downloaded the exchange rate to BTC for each coin, and then the existing BTC pricing data used to convert this value to USD. Below we made visualizations of price in USD for altcoins. The altcoins used in this analysis are Ethereum (ETH), Litecoin (LTC), Ripple (XRP), Ethereum Classic (ETC), Stellar (STR), Dash (DASH), Siacoin (SC), Monero (XMR), Zcash (ZEC), Lisk (LSK), Stratis (STRAT), NXT and NEM. The graph below provides a pretty solid "big picture" view of how the exchange rates for each currency have varied over the past few years. Initially, a logarithmic y-axis scale was used to compare all of the currencies on the same plot.

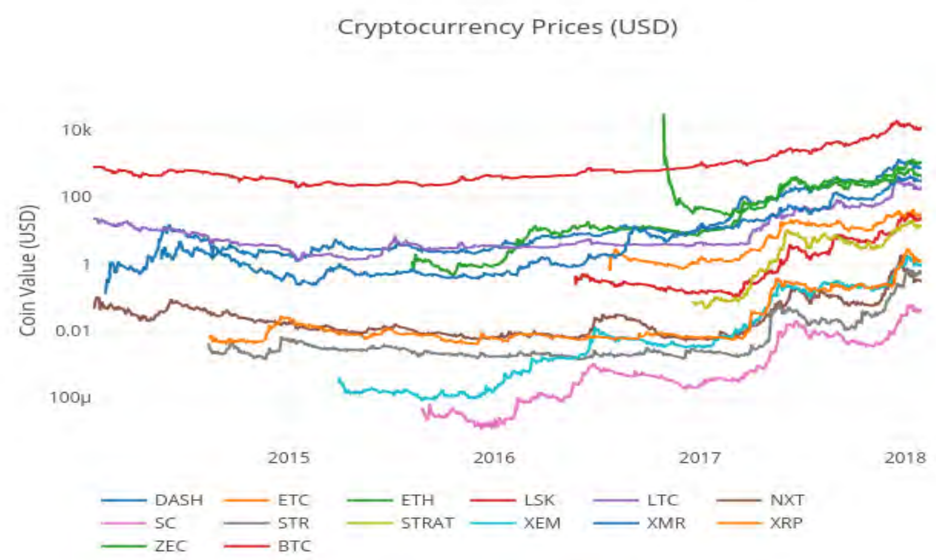

FIGURE 4.2: Cryptocurrencies Price(log-scale)

Then it appears in the linear scale the price of cryptocurrencies, this was mainly done to make the price of BTC clear and distinct in relation to the others. It is understood that Bitcoin is properly called the king of cryptocurrencies. Beyond that, something strange appears worthwhile explaining. The reason is for Zcash (ZEC) which for some time was at the top of all cryptocurrencies, precisely the initial first hour that entered the Poloniex trading.

Within 40 minutes of trading on Poloniex, a high of 3,299.99 bitcoins to one ZEC was recorded, or about US$2.3 million for each Zcash coin. In spite of the fact that the mania was uncommon, even in the cryptocurrency space, the explanation behind this grand level of significant worth was the extraordinarily decreased supply of coins because of a month-long, "Slow start", discharge plan. The volume of trading on Poloniex in the historic five-minute window was only 0.117 ZEC, worth about $268,383. The price of ZEC fell steadily after the peak of the first day. In almost two days its price dropped below the bitcoin price. Soon afterward it hit a low of 0.705 BTC ($510), where it has found some support, making it one of the most valuable cryptocurrencies.

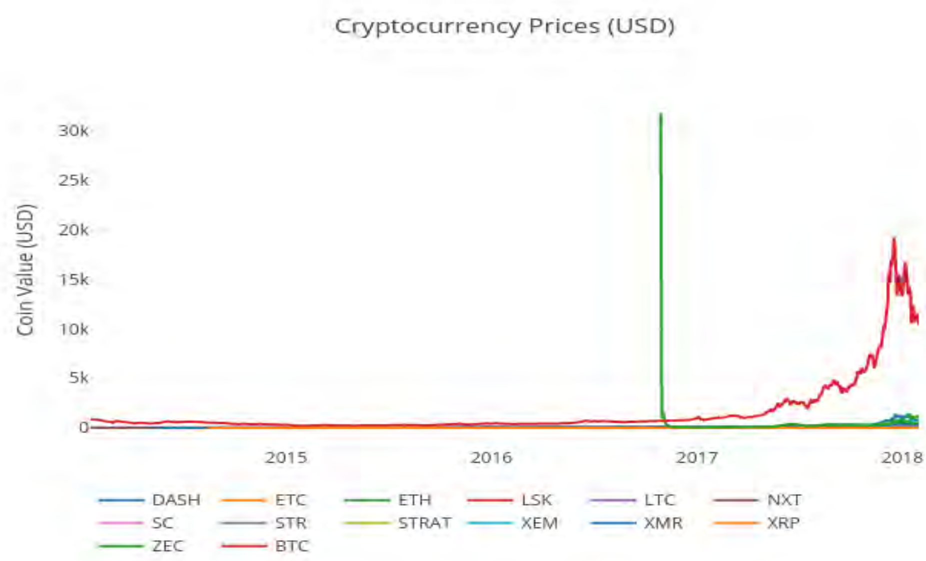

FIGURE 4.3: Cryptocurrencies Price(linear-scale)

4.2.1 Ethereum (ETH)

In 30th July 2015 the initial version of Ethereum is launched. Since the early months, Ethereum has shown a steady growth path in its price, without major steps until March 2016. In March 2016 it reached an increase of more than 105 percent. From now on we see a price fluctuation at the same level as some events happen. In June 2016 DAO, a digital decentralized autonomous organization and a form of investor-directed venture capital fund, was hacked. That event sparked a significant debate in the community market resulting in disputes. In 2nd July 2016, the network split into two groups, the Ethereum(ETH) and Ethereum Classic(ETC). Towards the end of July, Ethereum foundations, developers, business partners, miners and users of the Ethereum ecosystem disassociated themselves with ETC.

The hub is considered the 23rd February 2017 as eToro, one of the largest social trading and multi asset brokerage company, adds Ethereum trading. Since then, Ethereum has been on the upward trend until June 2017, with an important factor in this being AVATrade as it added it to its tradings. In 3rd June 2017 ,a surge in bullish momentum noticed as Russia endorses Ethereum. In June 2017, Ethereum rallies above 380$, recording an about 4500 percent rise since 1st January 2017. From now on we have some big fluctuations in the price with several peaks, but in general in 2017 we are seeing an outstanding 10,000% increase since 1 January. The new year 2018, namely in January, a fairly large negative peak happened, as its price falls almost vertically. Quite a lot of fluctuations happen in this new year. All of the above are summarized in the chart below, which deserves further study as the price depends on many factors. The chart below depicts the Weighted Price of Ethereum in dollars as it was formed from 2015 to today.

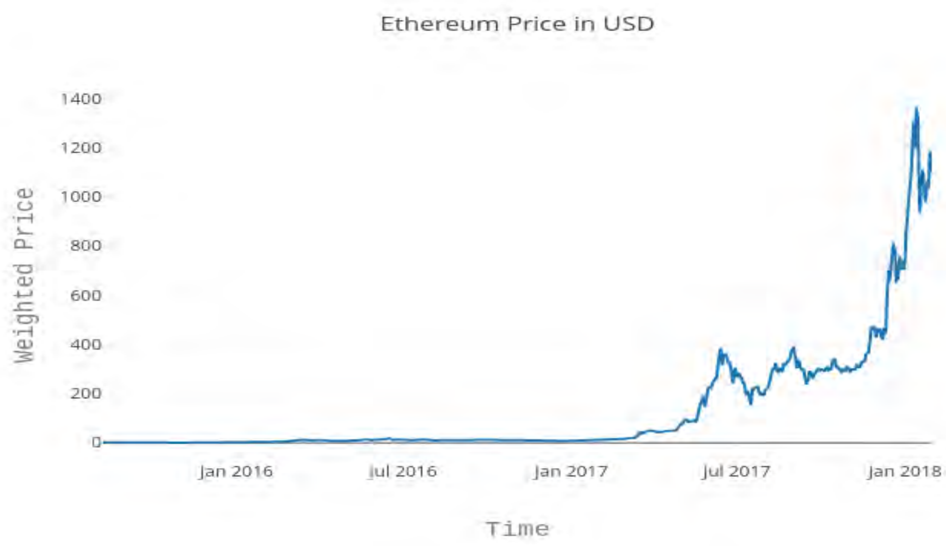

FIGURE 4.4: ETH/USD Weighted Price

4.2.2 Ripple (XRP)

While 2017 was major year for cryptocurrencies, Ripple appears to have flown under the radar to a specific degree. Surely, the digital currency has its offer of speculators and fans. As of this written work, with a worldwide market cap of more than $43 billion, it remains the fourth-largest virtual currency in the world.

Ripple(XRP) has been in presence for quite a long while, however the start of 2017 saw the cryptocurrency coasting at a low rate for each token. As of January 1 2017, one Ripple token could be obtained for a pitiful $0.0063. The price of XRP did not by any means take off until halfway during that time 2017. After a concise hop to simply finished $0.02 per token in late March 2017, the currency made rapid gains throughout late April and early May.

XRP reached a local high point on May 16, 2017, when it topped out at just over $0.40 per token. As is common for cryptocurrencies that have just reached new record prices, Ripple tumbled from that zenith in the days to take after. However, it settled at a significantly higher price point than it had been earlier in the year. From late May through early December, the currency floated in the region of $0.20 per token. Obviously, inside this period, there were numerous peaks and valleys, yet XRP did not come back to its past high amid that time.

After the primary seven day stretch of December 2017, the price of Ripple started to take off. On December 13, XRP climbed to a new record of more than $0.86 per token. As of this writing, XRP has taken off higher than ever, breaking the $1 per token price hindrance on December 21. With a few days left in the year, it stays to be seen whether this is a transitory price peculiarity or indications of more noteworthy development to come. Ripple's price taking off may have been connected to the declaration that both American Express and Santander would utilize the currency's blockchain to encourage certain kinds of cross-border transactions. The marching rally of XRP continued until 7 January 2018. It should be noted here that from 11 December 2017 to 7 January 2018, the percentage increase is more than 1250 percent.

From now on, however, the price begins and falls at a fast pace. There are multiple reasons related to this fast price fall. Some of them are firmly identified with online networking and social media. Ripple authorities have pointed the finger at

CoinMarketCap's choice to exclude data from South Korean markets for the current crash. Ripple's chief cryptographer David Schwartz bemoaned the move on Twitter, blaming CoinMarketCap for starting "frenzy selling". Some other reasons are that below:

- There is a major Lawsuit pending against Ripple.
- Ripple can control XRP to markets since it is centralized currency.
- There is no necessity for "Cheap payment processing".
- XRP's capitalization is falling rapidly.
- No big players want to invest in it.

All of the above are summarized in the chart below, which deserves further study as the price depends on many factors. The chart below depicts the Weighted Price of XRP in USD.

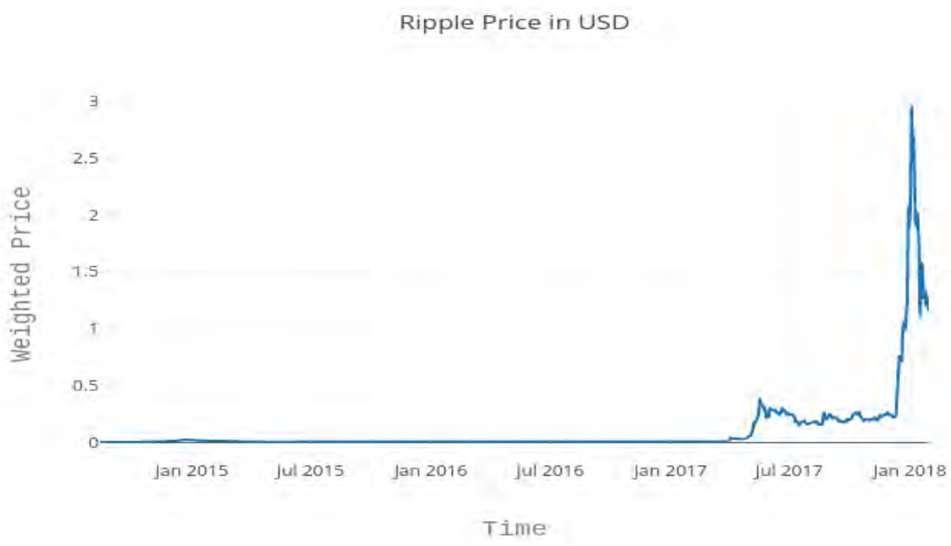

FIGURE 4.5: XRP/USD Weighted Price

4.3 Statistical Analysis

Descriptive statistics are used to describe the basic features of the data in a study. They provide simple summaries about the sample and the measures. Together with simple graphics analysis, they form the basis of virtually every quantitative analysis of data. Descriptive statistics are typically distinguished from inferential statistics [48]. With descriptive statistics you are basically portraying what is or what the information appears. We utilize descriptive statistics just to depict what is happening in our information.

With inferential statistics, you are endeavoring to achieve conclusions that extend beyond the immediate data alone. For instance, we utilize inferential statistics to attempt to gather from the sample data what the population might think. Or on the other hand, we utilize inferential statistics to make judgments of the likelihood that an observed difference between groups is a dependable one or one that might have happened by chance in this study [48]. In this way, we use inferential statistics to make inferences from our data to more broad conditions.

Below, we used both descriptive and inferential statistics in data analysis.

4.3.1 Correlation Analysis

Correlation analysis is a technique of statistical evaluation. It is used to consider the strength of a relationship between two, numerically measured, continuous variables. This specific kind of analysis is helpful when a researcher wants to establish if there are conceivable associations between variables. Correlation analysis does not decides circumstances and end results. However, this is not the situation on the grounds that different factors that are absent in the exploration may have affected on the outcomes.

If correlation is found between two variables it implies that when there is an efficient change in one variable, there is additionally an orderly change in the other; the variables alter together over a certain period of time. If there is correlation found, depending upon the numerical values measured, this can be either positive or negative. Positive correlation exists if one variable increases simultaneously with the other, i.e. the high numerical values of one variable relate to the high numerical values of the other. Negative correlation exists if one variable decreases when the other increases, i.e. the high numerical values of one variable relate to the low numerical values of the other.

Pearson's product-moment coefficient is the measurement of correlation. It ranges, (depending on the correlation), between $+1$ and -1.

- $+1$ indicates the strongest positive correlation possible
- -1 indicates the strongest negative correlation possible

In this manner the nearer the coefficient to both of these numbers the stronger the correlation of the data it represents. On this scale 0 indicates no correlation, hence values closer to zero highlight weaker/poorer correlation than those closer to $+1/-1$. Coefficients close to 1 or -1 mean that the series' are strongly correlated or inversely correlated respectively, and coefficients close to zero mean that the values are not correlated, and fluctuate independently of each other.

Four datasets were created one for each year from 2015 to 2018 including price data ,more specifically the weighted price in USD, from all the altcoins used in the study and the bitcoin. Then correlation matrices were created for each year to see the correlation between altcoins and bitcoin in relation to their value. Visualizations of correlation matrices have been made with heatmaps for better interpretation and results extraction.

In heatmaps below, the dark red values represent strong correlations (note that each currency is, obviously, strongly correlated with itself), and the dark blue values represent strong inverse correlations. All of the light blue/orange/gray/tan colors in-between represent varying degrees of weak/non-existent correlations. What is worth to be observed in the following correlation heatmaps is mainly the transition of colors, that is to say the correlations from 2015 to 2018, and not so much each heatmap separately.

In the correlation heatmap of 2015 there are some loopholes, mainly derived from missing values, as there were no such cryptocurrencies at that time. In heatmaps of 2015 and 2016, it can be observed that there was little statistically significant linkage between how the prices of different cryptocurrencies fluctuated during 2015. This perhaps worth mentioning is that from 2015 to 2016 the correlation of the BTC seems to have decreased with most cryptocurrencies but increased with the LTC.

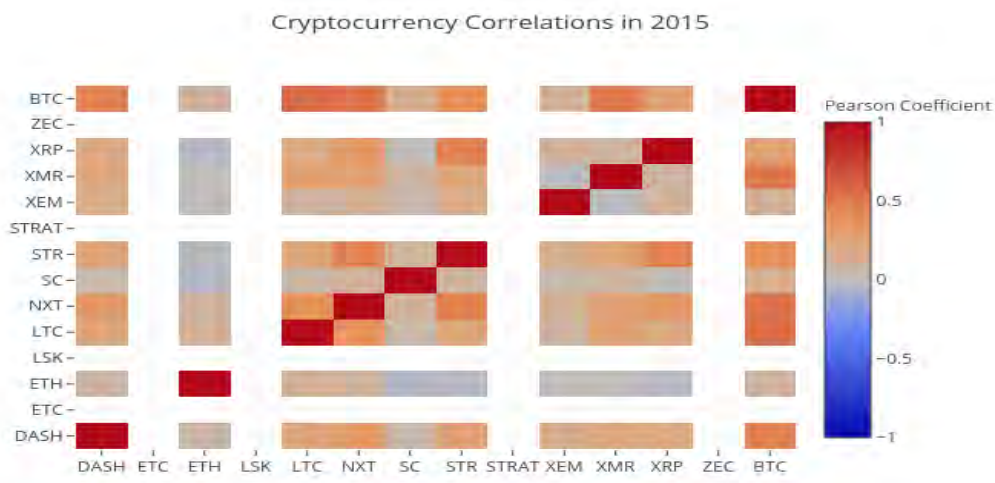

FIGURE 4.6: Correlation Heatmap 2015

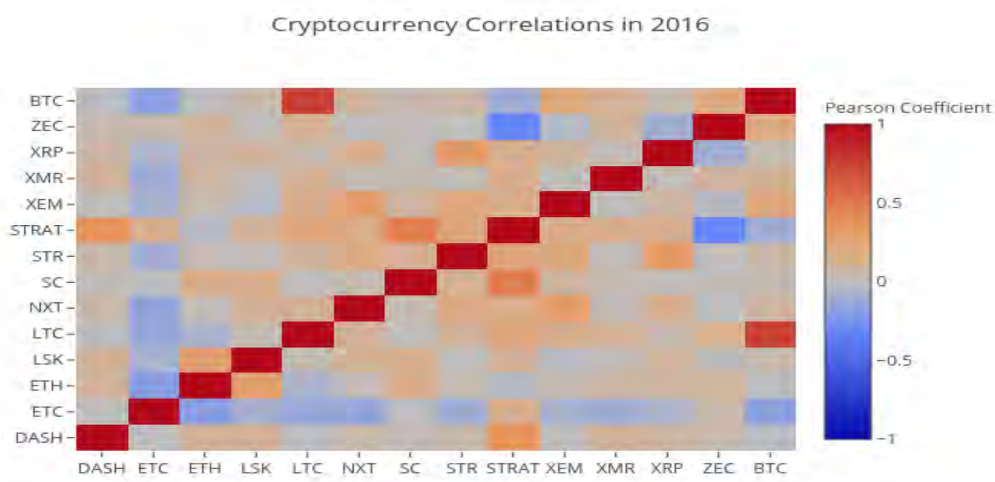

FIGURE 4.7: Correlation Heatmap 2016

In the correlation heatmap of 2017 there are more significant correlation coefficients. Almost all of the cryptocurrencies have become more correlated with each other across the board. The most immediate explanation is that hedge funds have recently begun publicly trading in cryptocurrency markets. These funds have vastly more capital to play with than the average trader, so if a fund is hedging their bets across multiple cryptocurrencies, and using similar trading strategies for each based on independent variables, it could make sense that this trend of increasing correlations would emerge.

One noticeable trait of the above chart is that XRP, is the least correlated cryptocurrency. The notable exception here is with STR, which has a stronger correlation with XRP. What is interesting here is that STR and XRP are both fairly similar fintech platforms aimed at reducing the friction of international money transfers between banks. It is conceivable that some big-money players and hedge funds might be using similar trading strategies for their investments in STR and XRP, due to the similarity of the blockchain services that use each token. This could explain why XRP is so much more heavily correlated with STR than with the other cryptocurrencies.

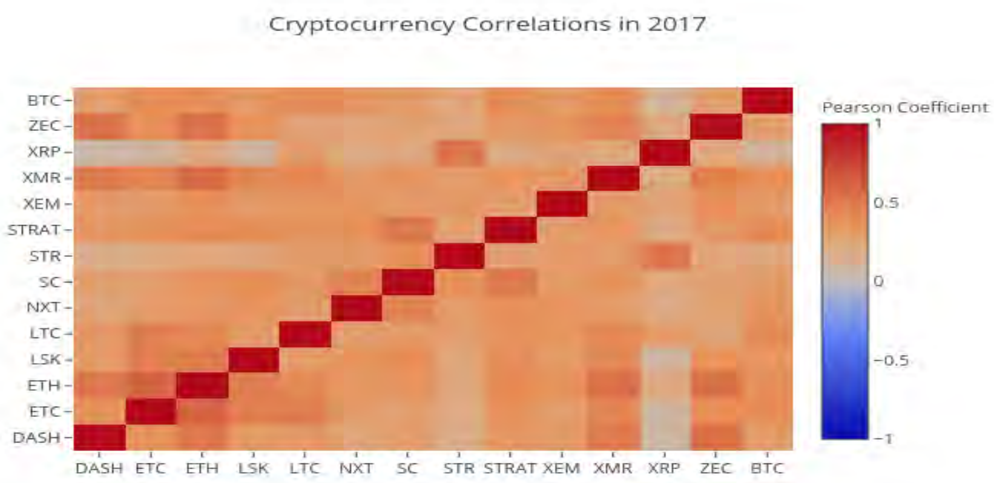

FIGURE 4.8: Correlation Heatmap 2017

Observing the 2018 correlation heatmap seems something that is unrealistic, as almost all cryptocurrencies are much more related to each other, some cells in the table even occur to have a perfect correlation. This may be because it is still the beginning of the year, and the data is relatively little to come to a conclusion. However, what seems to be for this little bit of time that passed in 2018 seems that there is enough correlation among all cryptocurrencies.

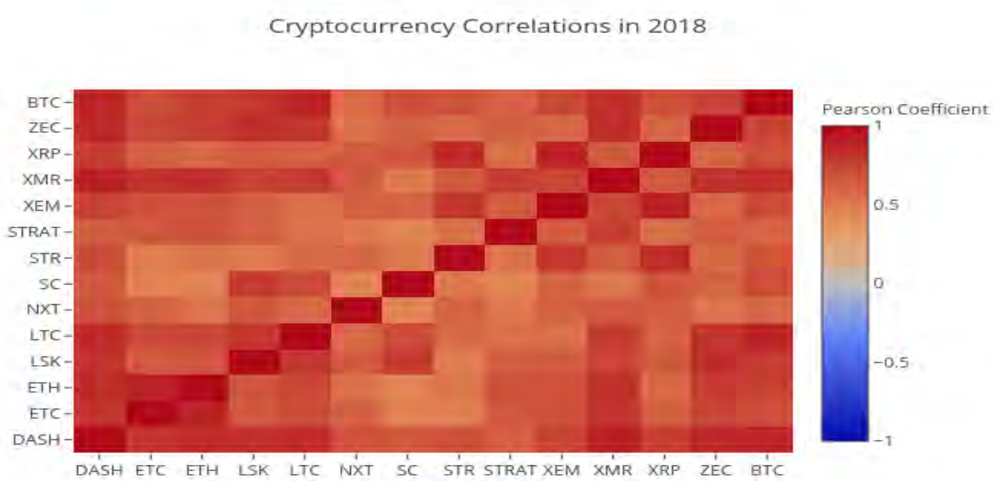

FIGURE 4.9: Correlation Heatmap 2018

In a general context, it seems that between 2015 and 2018 cryptocurrencies are becoming more and more correlated. The cryptocurrencies examined have highly dynamic price relationships. Sometimes they move in tandem, while other times they move in opposite directions. In certain cases, their correlation is very low. In certain instances, the markets develop clear trends where many cryptocurrencies move either higher or lower over specific periods of time. During others, these digital currencies may display little or no price correlation. As the cryptocurrency space becomes more mature, however, the individual digital assets that compose this market may very well progress in terms of carving out their own individual niches, a development that could cause their price determinants to change over time.

4.3.2 Distribution Analysis of 2017

Distributions (or generalized functions) are objects that generalize the classical notion of functions in mathematical analysis. Distributions make it possible to differentiate functions whose derivatives do not exist in the classical sense. In particular, any locally integrable function has a distributional derivative. Distribution analysis provides information about the distribution of numeric variables. A variety of plots such as histograms, probability plots, and quantile-quantile plots can be used in this analysis.

Boxplots

Boxplot is a simple way of representing statistical data on a plot in which a rectangle is drawn to represent the second and third quartiles, usually with a vertical line inside to indicate the median value. The lower and upper quartiles are shown as horizontal lines either side of the rectangle. Outliers may be plotted as individual points.

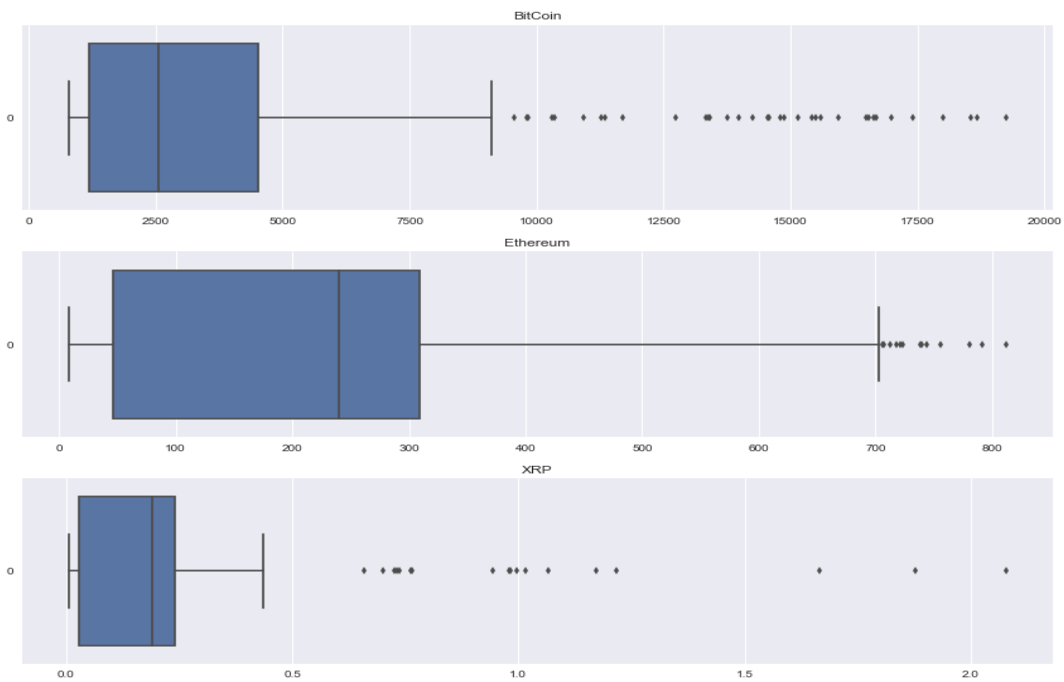

FIGURE 4.10: Boxplots 2017

On the box plots above:

1. BTC price was most of the time between $1250 and $4750 approximately in the last year. All values over $9000 are outliers (using interquartile range (IQR)). Note that outliers are specific to this data sample.
2. ETH price was most of the time between $50 and $310 approximately in the last year. All values over $700 are outliers (using interquartile range (IQR)). Note that outliers are specific to this data sample.
3. XRP price was most of the time between $0.05 and $0.25 approximately in the last year. All values over $0.4 are outliers (using interquartile range (IQR)). Note that outliers are specific to this data sample.

Histograms

Histogram is a diagram consisting of rectangles whose area is proportional to the frequency of a variable and whose width is equal to the class interval. A histogram is an accurate representation of the distribution of numerical data. It is an estimate of the probability distribution of a continuous variable .

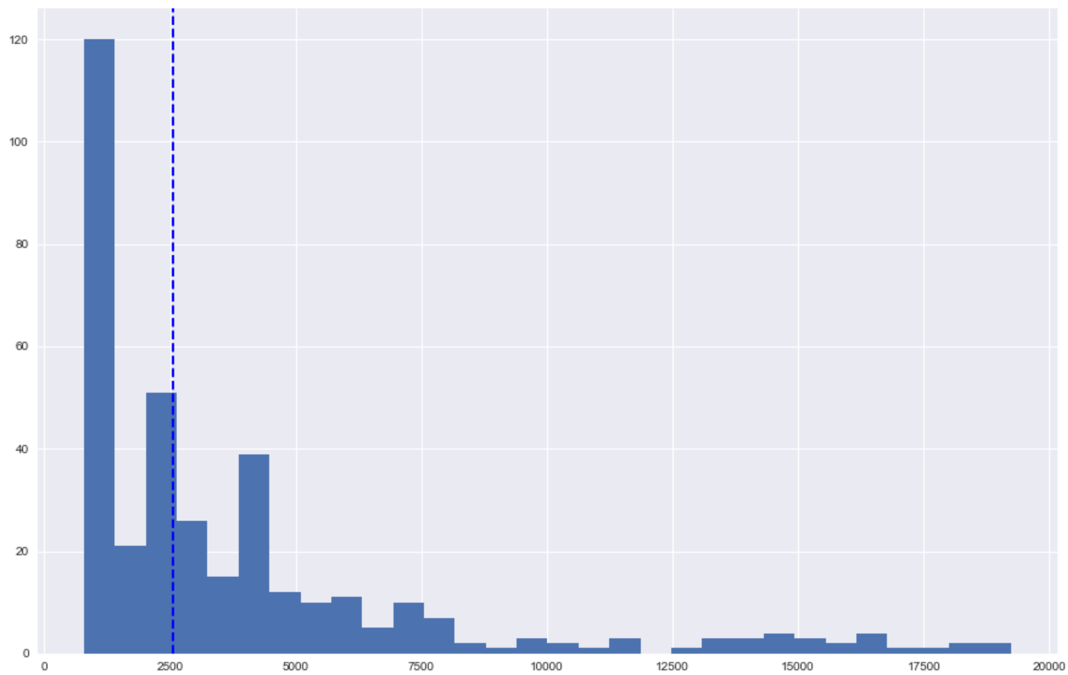

FIGURE 4.11: Histogram - BTC/USD 2017

Interpretation :

- It shows the number of days BTC had a certain value. For example, it can be observed that BTC price was not over $5000 for many days.
- It has right-skewed distribution because a natural limit prevents outcomes on one side.
- Blue dotted line (median) shows that half of the prices were under $2500.

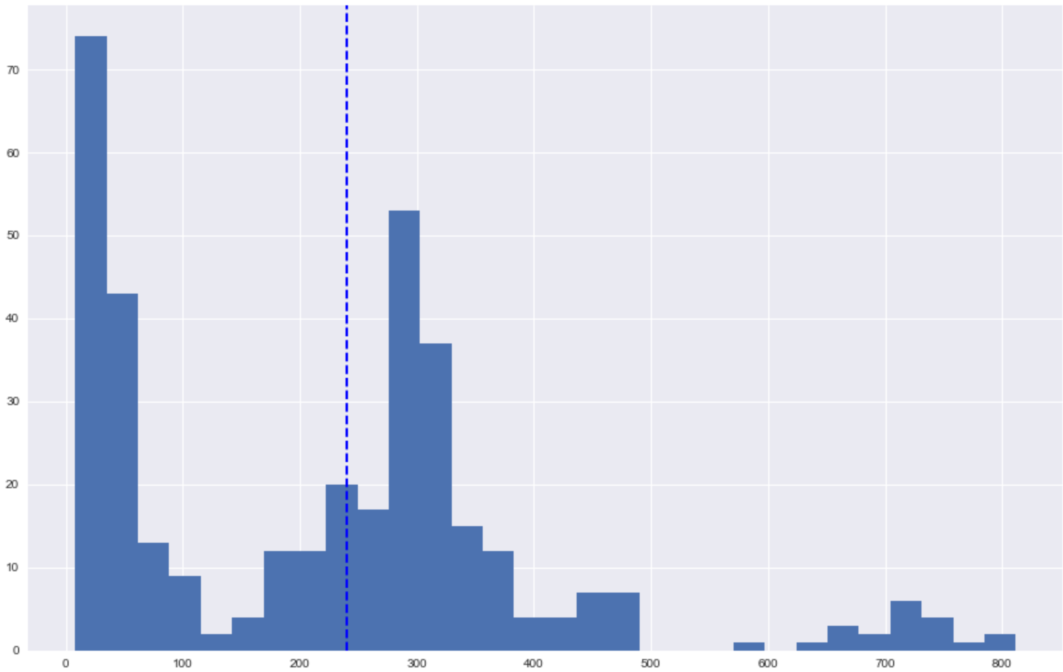

FIGURE 4.12: Histogram - ETH/USD 2017

Interpretation :

- It shows the number of days ETH had a certain value.
- It has approximately bimodal distribution, we can observe 2 modes one the natural limit of 0 and one in $300.
- Blue dotted line (median) shows that half of the prices were under $230.

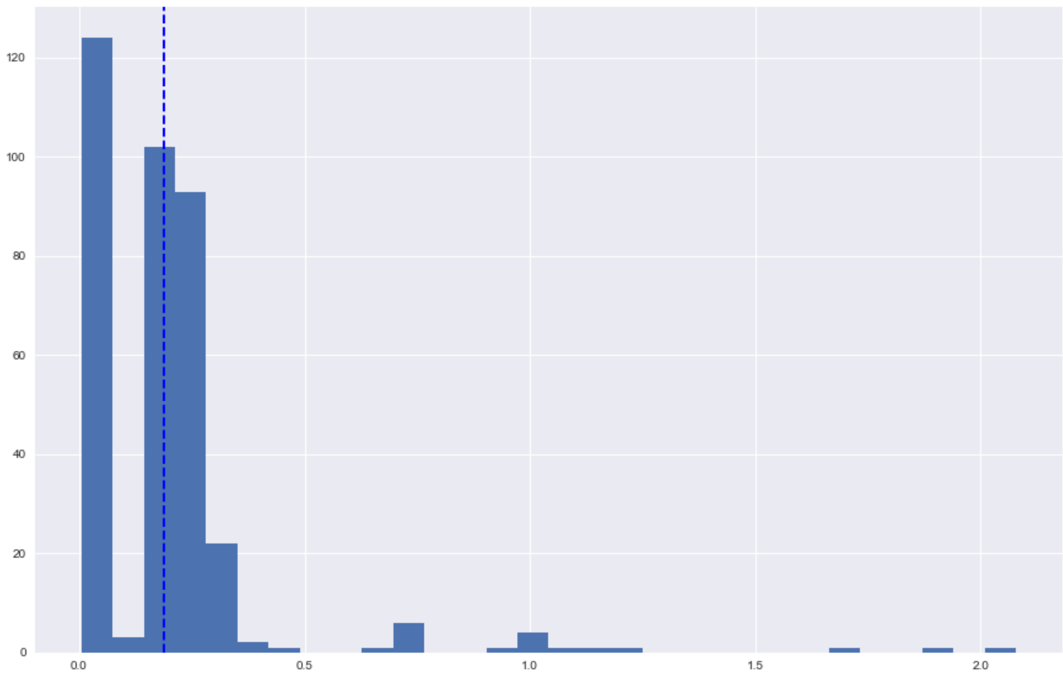

FIGURE 4.13: Histogram - XRP/USD 2017

Interpretation :

- It shows the number of days XRP had a certain value. We can observe that price was not over \$0.4 for many days.
- It has right-skewed distribution.
- Blue dotted line (median) shows that half of the prices were under \$0.2.

4.4 Trading Strategies

In finance, a trading strategy is a fixed plan that is designed to achieve a profitable return by going long or short in markets. Trading strategy is a set of objective rules defining the conditions that must be met for a trade entry and exit to occur. Trading strategies include specifications for trade entries, including trade filters and triggers, as well as rules for trade exits, money management, time frames, order types, and other relevant information. A trading strategy, if based on quantifiably specifications, can be analyzed based on historical data to project future performance.

4.4.1 Buy & Hold

Buy and hold is a passive investment strategy in which a speculator or an investor purchases stocks and holds them for a drawn out stretch of time, paying little heed to changes and fluctuations in the market. An investor who utilizes a buy and hold methodology actively selects stocks, but once in a position, is not concerned with short-term price movements and technical indicators.

The return percentage is calculated by :

$$r_{t,0} = \frac{p_t}{p_0}$$

Interpretation :

- r represents return.
- p represents price and p_0 is initial price.
- t represents a certain time period.

Below the chart depicts the return of cryptocurrencies for the year 2017 according to the trading strategy, buy and hold. More specifically, focus is on cryptocurrencies such as BTC, ETH, XRP. Explaining the chart could be said that if the above strategy had been applied then the XRP at the end of the year would be extremely profitable in relation to the BTC and ETH. Second most profitable in one year, 2017, comes ETH and third is BTC. This is reasonable if one thinks that the percent incrase from the beginning of the year to the end of the year 2017 is very different between these three cryptocurrencies. The percentage increase is shown in the table 4.1 below. So, very simply, if someone had invested 1,000 in each of the above cryptocurrencies, by the end of the year he would have been richer by about \$290,000 if he had invested in XRP, \$86,000 had he invested in the ETH and only \$13500 had invested in BTC.

Cryptocurrency	Percentage Increase
BTC	1353 %
ETH	8613 %
XRP	29014 %

TABLE 4.1: Percentage Increase of BTC,ETH,XRP - 2017

FIGURE 4.14: Buy & Hold Strategy 2017

4.4.2 Log Returns

Advantage of utilizing returns, versus prices, is normalization: estimating all factors in a similar metric, consequently empowering assessment of analytic relationships among at least two factors regardless of beginning from price arrangement of unequal esteems. This is a requirement for many multidimensional statistical analysis and machine learning techniques. There are several benefits of using log returns. Some of them are log-normality assumption, raw-log equality approximation, time-additivity. The reason individuals utilize log returns is that they are roughly invariant and subsequently less demanding to work with in estimating distributions. The fundamental idea is that the distribution of security prices is log-normal, so the arithmetic returns will likewise be. However, making a log transformation results in approximately normal returns, which are less demanding to work with. Additionally, if the supposition is made to be normally distributed, at that point there are convenient results for the convolution of multivariate normal series. This is the thing that takes into account less demanding time-aggregation.

Mathematical formulation of returns :

$$r_i = \frac{p_i - p_j}{p_j} = \frac{p_i}{p_j} - \frac{p_j}{p_j} \quad \Leftrightarrow \quad 1 + r_i = \frac{p_i}{p_j}$$

Interpretation :

- r_i denotes the return at time i.
- p_i denotes the price at time i.
- $j = i - 1$.

Continuing with the above equation one can log in so we have the following new equation representing log-returns.

Mathematical formulation of log-returns :

$$\log(1 + r_i) = \log(\frac{p_i}{p_j}) \quad \Leftrightarrow \quad \log(1 + r_i) = \log(p_i) - \log(p_j)$$

The log returns technique was applied to the cryptocurrencies BTC, ETH, XRP. This results in a better comparison between the three. The following graph shows for the year 2017 their log returns.

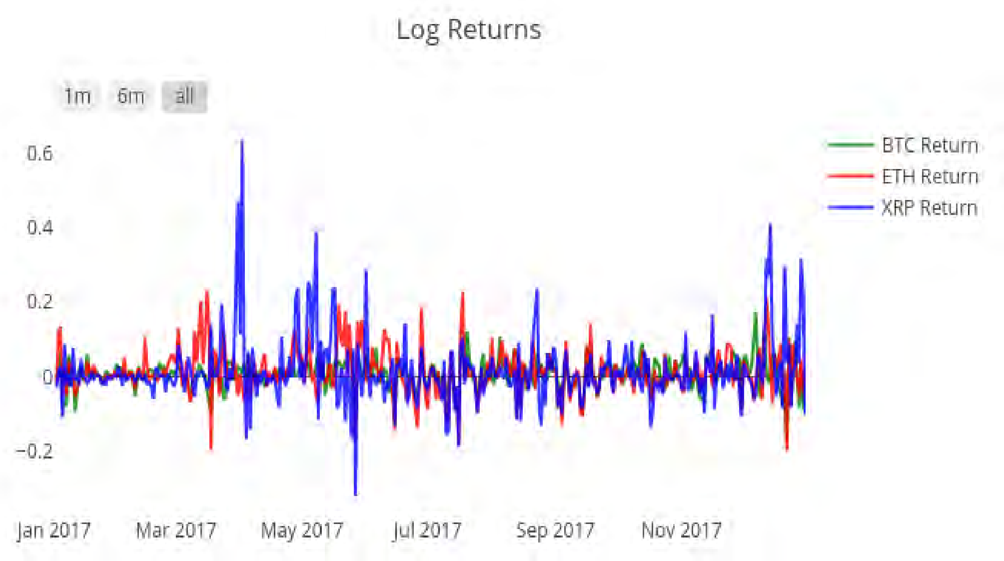

FIGURE 4.15: Log-Returns Strategy 2017

Finally, by applying the log returns' trading strategy, one can more easily model the behavior of these time series in addition to better and more accurate comparison of data with different scales. This is on account of it diminishes the variation of the time series making it easier to fit the model being referred to.

Chapter 5

Machine Learning

In this section, we present models for predicting the closing price of Ethereum (ETH). The price is in USD. The ETH / USD data used for the training of neural networks as well as time series data were sampled and separated into training and validation set in accordance with the following technique. New samples are constructed that pair sequences of N samples with the subsequent K samples. In this way, a regression model can be fit which predicts K time periods into the future given data from the past M. More specifically, the past 256 samples are transformed into a prediction about the next 16 samples. Samples of close and weighted average prices used as features to predict the closing price. These samples were split for training the following models in 80 percent training and 20 percent test set.

The following subsections show the neural networks used in this research, as well as the various architectures of those tested. It is worth mentioning that we used various activation functions, such as Rectified Linear Unit (ReLU), Leaky Rectified Linear Unit (LeakyReLU), Hyperbolic Tangent (TanH). In the neural networks below, the mean squared error (MSE) was chosen as the loss function and as an optimizer, adam.

Rectified Linear Unit (ReLU) is believed to be one purpose behind the rebirth of neural networks in the recent years. It computes the function $f(x) = max(0, x)$. Especially, the activation is commonly thresholded at zero. More specifically, below is the mathematical formula of the function as well as the image below (figure 5.1).

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

The derivative with respect to x:

$$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$$

There are several pros and cons to using the ReLUs:

- Pros:
  - It was found to incredibly quicken the convergence of stochastic gradient descent compared to the sigmoid/tanh functions. It is contended this is because of its linear, non-saturating form.
  - Compared to tanh/sigmoid neurons that involve expensive operations, the ReLU can be implemented by simply thresholding a matrix of activations at zero.

- Cons:
  - Unfortunately, ReLU units can be fragile during training and can "die". If the dot product of the input to a ReLU with its weights is negative, the output is 0. The gradient of $max(0, x)$ is 0 when the output is 0. If the output of a ReLU is consistently 0, then the gradient through it will consistently be 0. The error signal back-propagated from later layers gets multiplied by this 0, so no error signal ever passes to earlier layers. The ReLU has died.

ReLU's are unstable and can die when large gradients flowing throw them cause the update to happen in such a way that they never get activated again. Leaky Rectified Linear Units (LeakyReLUs) were introduced to beat the "dying ReLU" issue and as the name recommends takes into account small negative values. The reason one may maintain a strategic distance from a Leaky ReLU during training is to diminish small valued floating point computations when back-propagation when contrasted with the standard ReLU. The mathematical formula of the function as well as the image below (figure 5.1).

$$f(x) = \begin{cases} \alpha x & \text{for } x < 0\,, \quad \alpha \in [0, 1) \\ x & \text{for } x \geq 0 \end{cases}$$

The derivative with respect to x:

$$f'(x) = \begin{cases} \alpha & \text{for } x < 0\,, \quad \alpha \in [0, 1) \\ 1 & \text{for } x \geq 0 \end{cases}$$

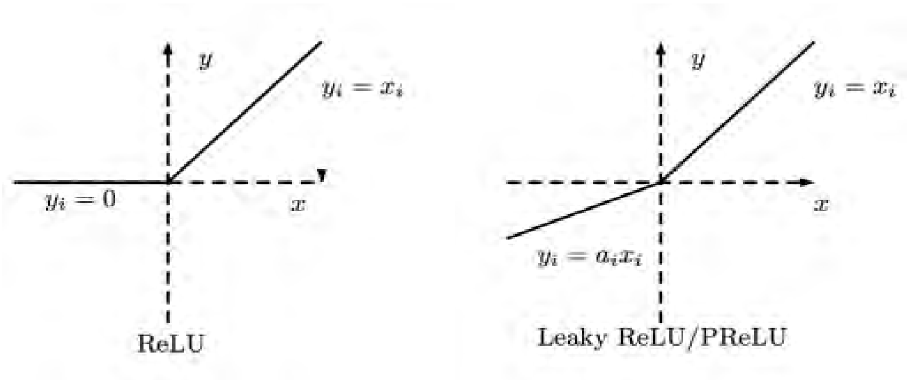

FIGURE 5.1: ReLU vs LeakyReLU [49]

The hyperbolic tangent function (TanH) is an old mathematical function. This function is easily defined as the ratio between the hyperbolic sine and the cosine functions. It squashes a real-valued number to the range $[-1, 1]$. Like the sigmoid neuron, its activations saturate, but unlike the sigmoid neuron its output is zero-centered. Therefore, in practice the tanh non-linearity is always preferred to the sigmoid nonlinearity. Also note that the tanh neuron is simply a scaled sigmoid neuron. The mathematical formula of the function is show below :

$$f(x) = tanh(x) = \frac{sinh(x)}{cosh(x)} = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$

The derivative with respect to x:

$$f'(x) = 1 - f(x)^2$$

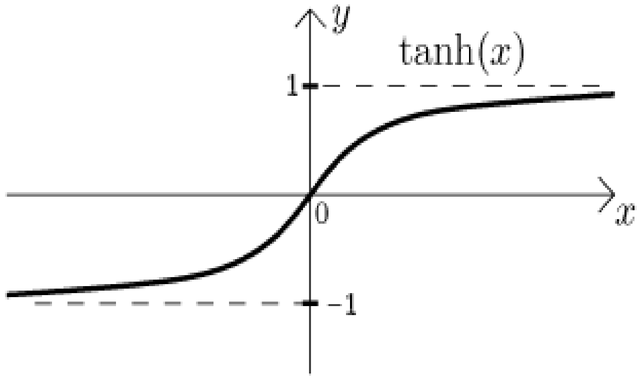

FIGURE 5.2: TanH [50]

In mathematical optimization, statistics, econometrics, decision theory, machine learning and computational neuroscience, a loss function or cost function is a function that maps an event or values of one or more variables onto a real number intuitively representing some "cost" associated with the event. Mean Squared Error (MSE), or quadratic, loss function is widely used in regression tasks. The target of MSE loss function is to minimize the residual sum of squares. Regression analysis is used when you want to predict a continuous dependent variable from a number of independent variables.
The standard form of MSE loss function is defined as :

$$L = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Interpretation :

- y_i is the actual values.
- $\hat{y}_i$ is the predicted values.
- $(y_i - \hat{y}_i)$ is named as residual.

Adam is an optimization algorithm that can used instead of the classical stochastic gradient descent procedure to update network weights iterative based in training data. Adam is different to classical stochastic gradient descent. Stochastic gradient descent maintains a single learning rate (termed alpha) for all weight updates and the learning rate does not change during training. By using adam, a learning rate is maintained for each network weight (parameter) and separately adapted as learning unfolds. This method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients. Adam is a popular algorithm in the field of deep learning because it achieves good results fast.

Regarding the input data and the following models, we used 256 past samples of the feature 'close price' and the 'weighted average price', with a period of time between two consecutive observations of 5 minutes. For the training of the neural networks, the train - test split technique was used, with a ratio of 0.75 training set and 0.25 validation set. With regard to the output of the following models, 16 predictions of closing price are made for each 256 past samples accepted as input, each forecast remaining in the previous 5-minute period. In plain numbers we had about 273900 instances of data, sampled as $\frac{1}{256+16} = \frac{1}{272}$, which means that the predictive models trained on about 755 samples and validated on about 252 samples.

5.1 Convolutional Neural Networks (CNN)

Convolutional Neural Network (CNN) is a special type of neural network that works in the same way of a regular neural network except that it has a convolution layer at the beginning. Convolutional Neural Networks (CNN) are widely used for classification, mainly for image classification. Image classification is the task of taking an input image and outputting a class or a probability of classes that best describes the image. Here, the CNN was used and tested in a continuous variable predictor. Below are the different layers that have been tested.

A 1D CNN is expected to capture the data locality well with the kernel sliding across the input data. The spatial size of the output volume is computed as a function of the input volume size (W), the receptive field size of the convolutional layer neurons (F), the stride with which they are applied (S), and the amount of zero padding used (P) on the border. The formula for calculating how many neurons fit is given by

$$O = \frac{(W - F + 2P)}{S} + 1$$

5.1.1 CNN 2-layers

Initially a simple convolutional neural network with 2 layers was created. Below is its architecture, as shown by the keras plotting function for models. It is worth noting that next to the layers are the data shapes that are passed to each layer. As an activation function of the first hidden convolutional layer, ReLU was chosen.

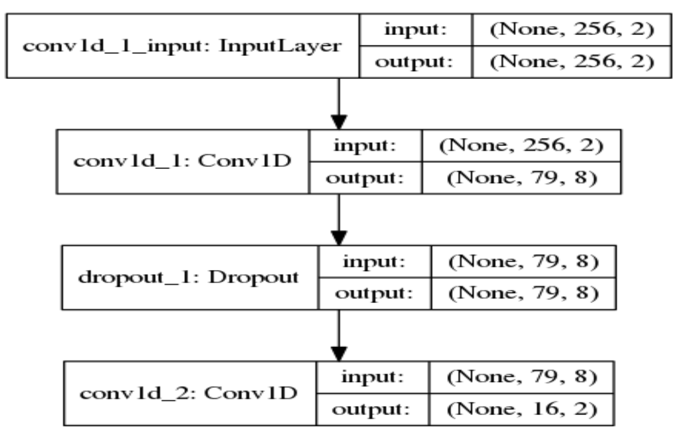

FIGURE 5.3: CNN - 2 Layers Architecture

Interpretation :

- Conv1D: This layer creates a convolution kernel that is convolved with the layer input over a single spatial (or temporal) dimension to produce a tensor of outputs.

- Dropout: This is a regularization technique for reducing overfitting in neural networks by preventing complex co-adaptations on training data. It is a very efficient way of performing model averaging with neural networks. The term "dropout" refers to dropping out units (both hidden and visible) in a neural network [51]. At each training stage, individual nodes are either "dropped out" of the net with probability $1 - p$ or kept with probability p , so that a reduced network is left; incoming and outgoing edges to a dropped-out node are also removed. Only the reduced network is trained on the data in that stage. The removed nodes are then reinserted into the network with their original weights. By avoiding training all nodes on all training data, dropout decreases overfitting.

Data fitting

Data fitting is the process of fitting models to data and analyzing the accuracy of the fit. Engineers and scientists use data fitting techniques, including mathematical equations and nonparametric methods, to model acquired data.

The graph below shows the approximation of the price curve predicted by the CNN, with 2 layers, in the actual price data over a time period covering the last days of 2017 up to date.

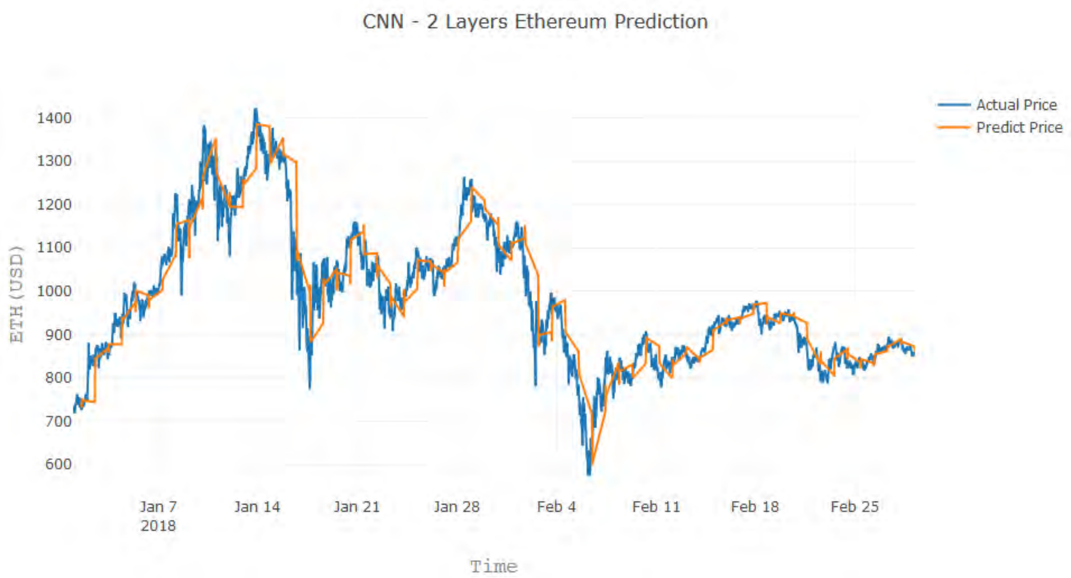

FIGURE 5.4: CNN - 2 Layers Data Fit

Training Loss

Training loss is simply the values of the loss function that was chosen and explained above (MSE), during the training stage of the algorithm, for 300 epochs. The best training loss is the minimum that was sustained. The following table shows the best training loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Training loss	CNN - 2 Layers
Normalized	7.7070e-05
Scale inverted	0.1146

TABLE 5.1: CNN - 2 Layers - Training Loss

Validation Loss

Validation loss is the values of the loss function that was chosen and explained above (MSE), during the validation stage of the algorithm, for 300 epochs. The best validation loss is the minimum that was sustained. The following table shows the best validation loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Validation loss	CNN - 2 Layers
Normalized	60.3988e-05
Scale inverted	0.8642

TABLE 5.2: CNN - 2 Layers - Validation Loss

Below the graphs show the training and validation loss for a CNN, with 2 layers, as calculated in each epoch.

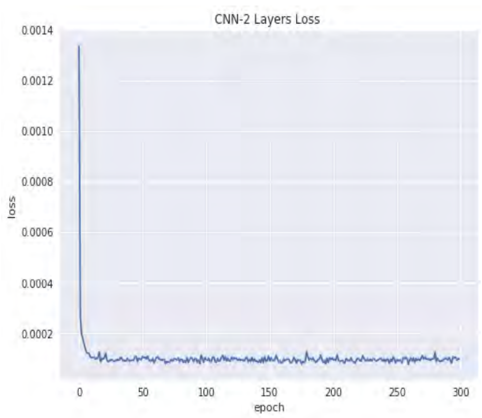

FIGURE 5.5: CNN - 2 Layers Training Loss

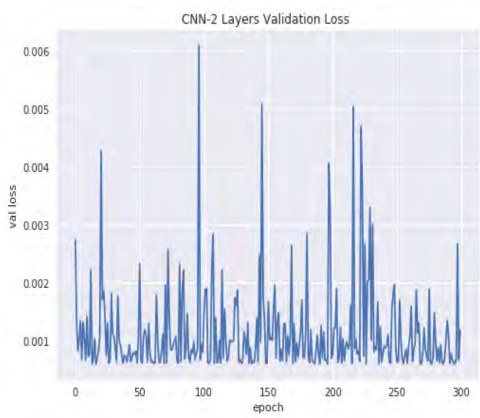

FIGURE 5.6: CNN - 2 Layers Validation Loss

The graphs show what is expected about the training loss, which decreases and remains stable at very low levels. However, with regard to validation loss, it seems that the CNN-2 layers model behaves very well, and quite badly. What is worth saying is that it does not seem to learn anything essential as validation loss is not gradually diminishing and there is a wide range of values .

5.1.2 CNN 3-layers

In addition, another neural network was CNN with 3 layers, its architecture shown below in the figure. The layers were explained in detail above. As an activation function of the hidden convolutional layers, ReLU was chosen.

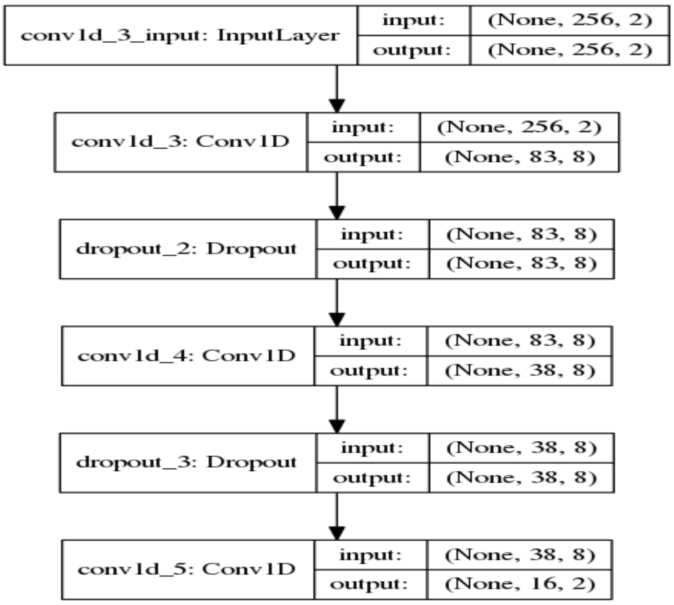

FIGURE 5.7: CNN - 3 Layers Architecture

Interpretation :

- Conv1D : This layer creates a convolution kernel that is convolved with the layer input over a single spatial (or temporal) dimension to produce a tensor of outputs.
- Dropout: This is a regularization technique for reducing overfitting in neural networks by preventing complex co-adaptations on training data. It is a very efficient way of performing model averaging with neural networks. The term "dropout" refers to dropping out units (both hidden and visible) in a neural network [51]. At each training stage, individual nodes are either "dropped out" of the net with probability $1 - p$ or kept with probability p , so that a reduced network is left; incoming and outgoing edges to a dropped-out node are also removed. Only the reduced network is trained on the data in that stage. The removed nodes are then reinserted into the network with their original weights. By avoiding training all nodes on all training data, dropout decreases overfitting.

Data fitting

The graph below shows the approximation of the price curve predicted by the CNN, with 3 layers, in the actual price data over a time period covering the last days of 2017 up to date.

FIGURE 5.8: CNN - 3 Layers Data Fit

Training Loss

The following table shows the best training loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Training loss	**CNN - 3 Layers**
Normalized	7.5831e-05
Scale inverted	0.1128

TABLE 5.3: CNN - 3 Layers - Training Loss

Validation Loss

The following table shows the best validation loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Validation loss	**CNN - 3 Layers**
Normalized	59.3210e-05
Scale inverted	0.8488

TABLE 5.4: CNN - 3 Layers - Validation Loss

Below the figures 5.9 and 5.10 show the training and validation loss for a CNN, with 3 layers, as calculated in each epoch. The graphs show what is expected about the training loss, which decreases and remains stable at very low levels. However, in this case, a better behavior of the network in terms of unknown network data appears. In the validation loss graph, although an initial reduction is observed and for a few epochs it remains quite low, it is later increased but maintained in fixed frames without these large peaks on the CNN-2 layers. It seems to be slightly better CNN-3 layers than CNN-2 layers in terms of learning.

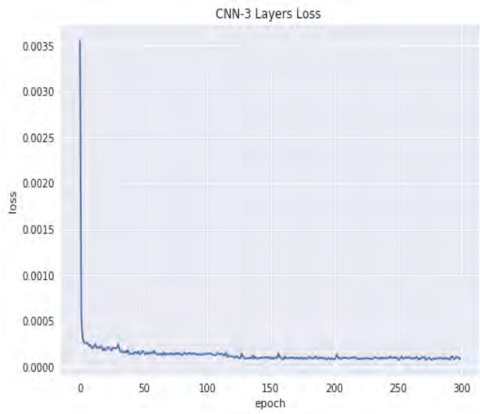

FIGURE 5.9: CNN - 3 Layers Training Loss

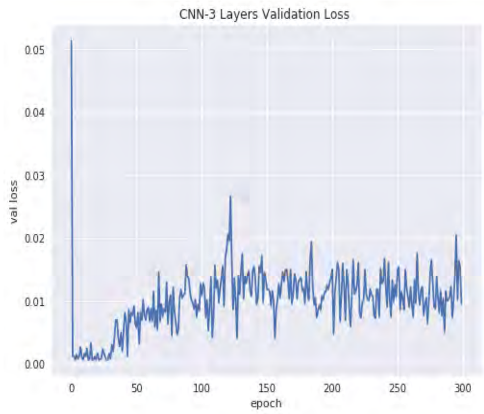

FIGURE 5.10: CNN - 3 Layers Validation Loss

5.1.3 CNN 4-layers

Furthermore, another neural network that was trained and tested was CNN with 4 layers. Its architecture shown below in the figure. The layers were explained in detail above. As an activation function of the hidden convolutional layers, ReLU was chosen.

Interpretation :

- Conv1D : This layer creates a convolution kernel that is convolved with the layer input over a single spatial (or temporal) dimension to produce a tensor of outputs.
- Dropout: This is a regularization technique for reducing overfitting in neural networks by preventing complex co-adaptations on training data. It is a very efficient way of performing model averaging with neural networks. The term "dropout" refers to dropping out units (both hidden and visible) in a neural network [51]. At each training stage, individual nodes are either "dropped out" of the net with probability $1 - p$ or kept with probability p , so that a reduced network is left; incoming and outgoing edges to a dropped-out node are also removed. Only the reduced network is trained on the data in that stage. The removed nodes are then reinserted into the network with their original weights. By avoiding training all nodes on all training data, dropout decreases overfitting.

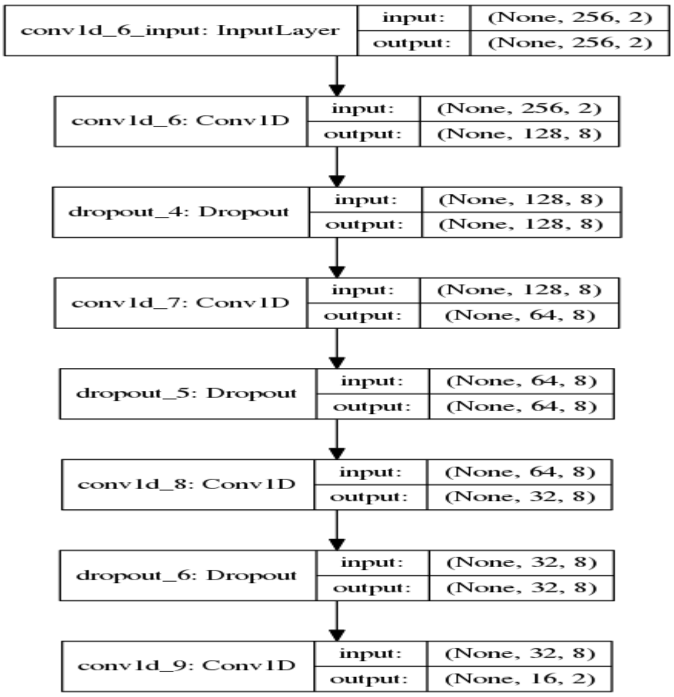

FIGURE 5.11: CNN - 4 Layers Architecture

Data fitting

The graph below shows the approximation of the price curve predicted by the CNN, with 4 layers, in the actual price data over a time period covering the last days of 2017 up to date.

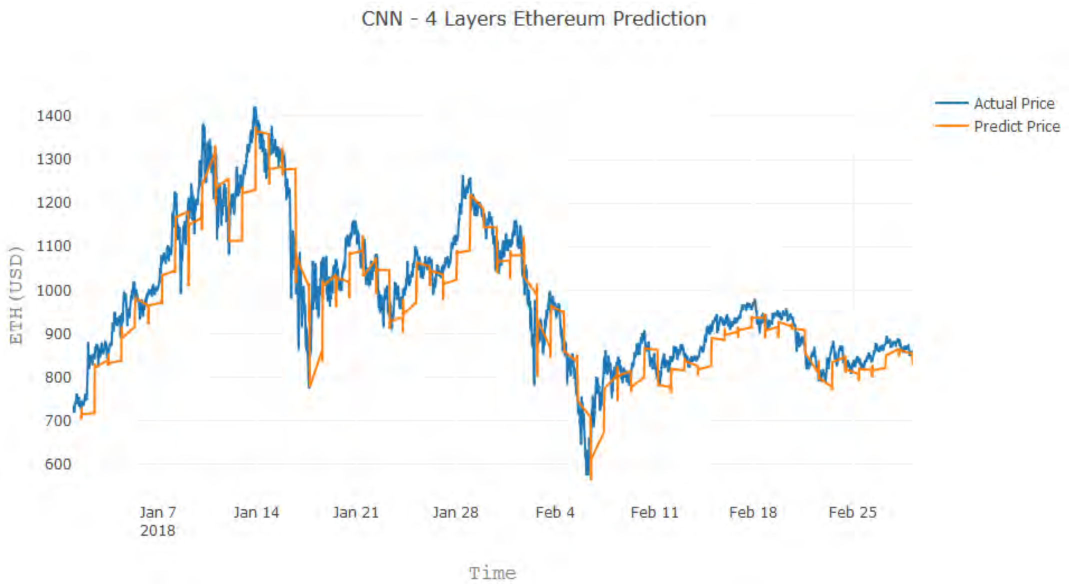

FIGURE 5.12: CNN - 4 Layers Data Fit

Training Loss

The following table shows the best training loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Training loss	**CNN - 4 Layers**
Normalized	22.4030e-05
Scale inverted	0.3237

TABLE 5.5: CNN - 4 Layers - Training Loss

Validation Loss

The following table shows the best validation loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Validation loss	**CNN - 4 Layers**
Normalized	89.8684e-05
Scale inverted	1.2833

TABLE 5.6: CNN - 4 Layers - Validation Loss

Below in figures 5.13 and 5.14, the graphs show the training and validation loss for a CNN, with 4 layers, as calculated in each epoch. In the training loss graph, it looks roughly the same behavior as the previous ones, but here it is higher. The validation loss chart shows an initial large increase. Its values are quite large and remain at a constant high level. In terms of learning, it is worse than the other 2 of its class.

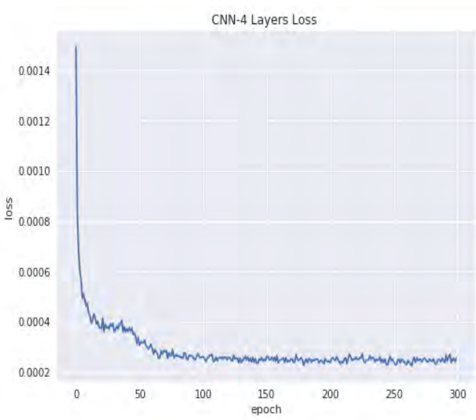

FIGURE 5.13: CNN - 4 Layers Training Loss

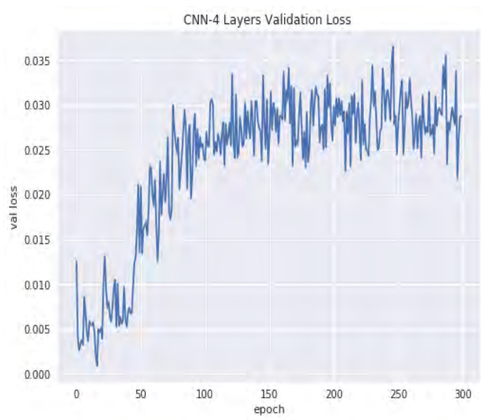

FIGURE 5.14: CNN - 4 Layers Validation Loss

5.2 Recurrent Neural Networks (RNN)

Recurrent neural networks are a type of neural network where the outputs from previous time steps are fed as input to the current time step. This creates a network graph or circuit diagram with cycles, which can make it difficult to understand how information moves through the network. Recurrent neural networks (RNN) are a widely used tool for the prediction of time series. RNNs are fit and make predictions over many time steps. Here in this subsection, the two types of RNNs used to predict the closing price of ETH, i.e. a continuous variable, are shown and tested. These two types are LSTM and GRU.

5.2.1 Long Short-Term Memory (LSTM)

Long Short Term Memory (LSTM) network is a variation of Recurrent Neural Network (RNN). It was invented to solve the vanishing gradient problem created by normal RNN. It is guaranteed that LSTMs are capable of remembering inputs with longer time steps. The LSTM network that has been created and used is shown below, where its architecture is shown. As an activation function in the hidden LSTM layer, *tanh* was used to find nonlinear relationships between the values. In the last layer, a LeakyReLU activation function is used as the task is nonlinear regression to predict a continuous non-negative variable.

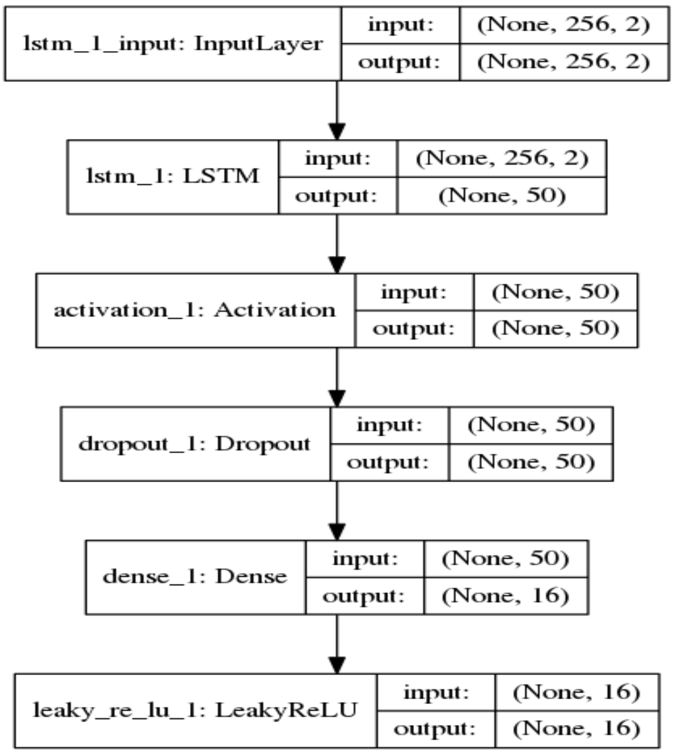

FIGURE 5.15: LSTM Architecture

Interpretation :

- LSTM : This layer creates a LSTM recurrent neural network cell.

- Dense: A dense layer represents a matrix vector multiplication. The values in the matrix are the trainable parameters which get updated during back-propagation.

$$u^T.W, \quad W \in \Re^{nxm}$$

The output is a m dimensional vector . A dense layer thus is used to change the dimensions of a vector. Mathematically speaking, it applies a rotation, scaling, translation transform to a vector. A dense layer is just a regular layer of neurons in a neural network. Each neuron receives input from all the neurons in the previous layer, thus densely connected .

Data fitting

The graph below shows the approximation of the price curve predicted by the LSTM recurrent neural network, in the actual price data over a time period covering the last days of 2017 up to date.

FIGURE 5.16: LSTM Data Fit

Training Loss

The following table shows the best training loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Training loss	**LSTM**
Normalized	2.6836e-05
Scale inverted	0.0509

TABLE 5.7: LSTM Training Loss

Validation Loss

The following table shows the best validation loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Validation loss	LSTM
Normalized	6.8012e-05
Scale inverted	0.1092

TABLE 5.8: LSTM Validation Loss

Below the graphs show the training and validation loss for a LSTM recurrent neural network, as calculated in each epoch.

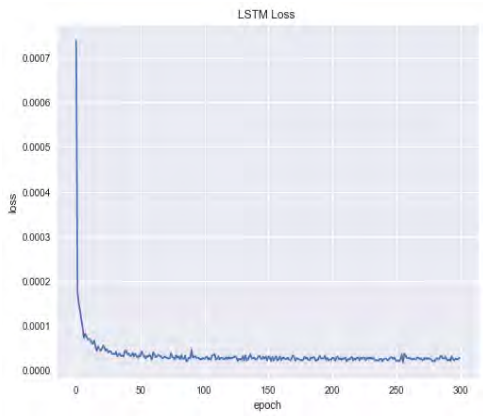

FIGURE 5.17: LSTM Training Loss

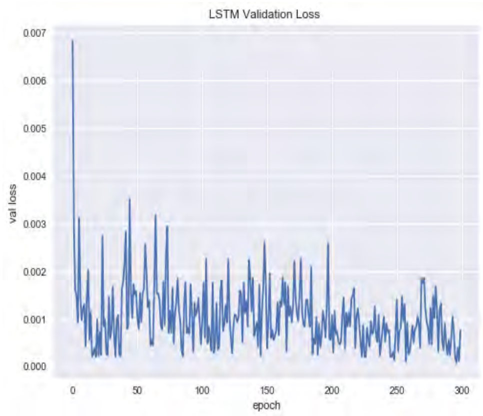

FIGURE 5.18: LSTM Validation Loss

The above graphs show a fairly low and stable training loss. In terms of validation loss, what appears to be is also at a much lower level than the models shown above. In general, there is a gradual small decrease even though there are some small local peaks.

5.2.2 Gated Recurrent Unit (GRU)

Gated Recurrent Units (GRU) is another variation of RNN. Its network structure is less sophisticated than LSTM with one reset and forget gate but getting rid of the memory unit. It is claimed that GRU's performance is on par with LSTM but more efficient. The GRU network that has been created and used is shown below, where its architecture is shown. As an activation function in the hidden GRU layer, *tanh* was used to find nonlinear relationships between the values. In the last layer, a LeakyReLU activation function is used as the task is nonlinear regression to predict a continuous non-negative variable.

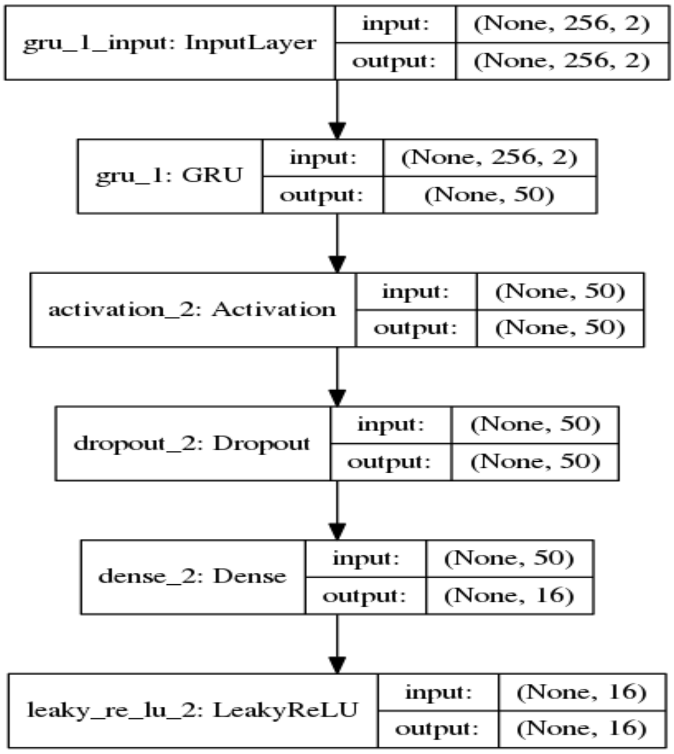

FIGURE 5.19: GRU Architecture

Interpretation :

- GRU : This layer creates a GRU recurrent neural network cell.
- Dense: A dense layer represents a matrix vector multiplication. The values in the matrix are the trainable parameters which get updated during back-propagation.

$$u^T.W, \quad W \in \Re^{nxm}$$

The output is a m dimensional vector . A dense layer thus is used to change the dimensions of a vector. Mathematically speaking, it applies a rotation, scaling, translation transform to a vector. A dense layer is just a regular layer of neurons in a neural network. Each neuron receives input from all the neurons in the previous layer, thus densely connected .

Data fitting

The graph below shows the approximation of the price curve predicted by the GRU recurrent neural network, in the actual price data over a time period covering the last days of 2017 up to date.

FIGURE 5.20: GRU Data Fit

Training Loss

The following table shows the best training loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Training loss	**GRU**
Normalized	2.7193e-05
Scale inverted	0.0527

TABLE 5.9: GRU Training Loss

Validation Loss

The following table shows the best validation loss as it was created by the normalized data entered in the neural network but also scale inverted to the real data scale.

Best Validation loss	**GRU**
Normalized	4.2915e-05
Scale inverted	0.0642

TABLE 5.10: GRU Validation Loss

The graphs, in figures 5.21 and 5.22, show the training and validation loss for a GRU recurrent neural network, as calculated in each epoch. Once again, what is expected is the training loss, which is almost the same as the LSTM. What is worth to say, however, is that GRU's validation loss is very low. Although peaks appear and there is no gradual decrease or increase, ie there is no specific behavior in these 300 epochs trained, it has the lowest validation loss of all the models tested.

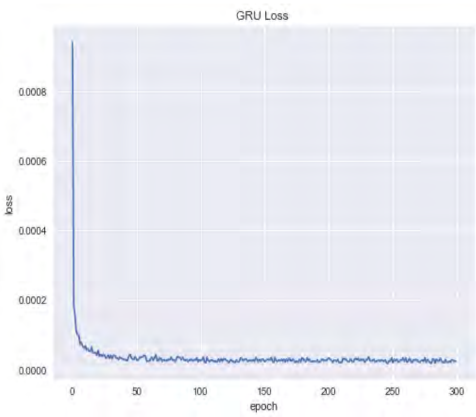

FIGURE 5.21: GRU Training Loss

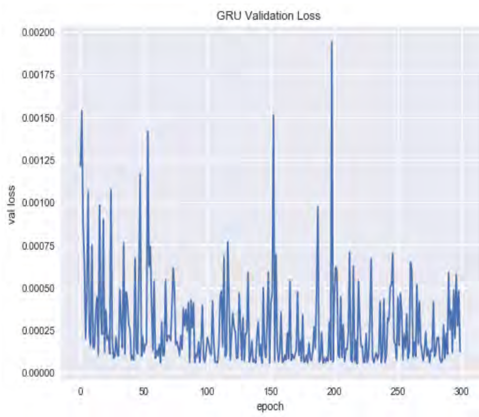

FIGURE 5.22: GRU Validation Loss

Chapter 6

Model Evaluation & Comparisons

In this section, comparisons will be made between the forecasting models, analyzed in the previous chapter. In time-series forecasting, the accuracy measure is obtained by the difference between the predicted value and the actual value where the subscript t denotes the values at time t (error of prediction at time t).

$$e_t = y_t - \hat{y}_t$$

6.1 Performance Metrics

Root mean squared error ($RMSE$) and Mean Absolute Error (MAE) are two of the most well-known metrics used to gauge accuracy for continuous variables. Both MAE and $RMSE$ express average model prediction error in units of the variable of interest. Both metrics can range from 0 to inf and are apathetic regarding the direction of errors. They are negatively-oriented scores, which implies lower values are better. We used $RMSE$ and MAE in order to measure the performance of the regression models show in the previous chapter, Machine Learning.

6.1.1 Root Mean Square Error

The root-mean-square deviation ($RMSD$) or root-mean-square error ($RMSE$) (or sometimes root-mean-squared error) is a much of the time utilized measure of the contrasts between values (sample and population values) predicted by a model or an estimator and the values actually observed. The $RMSE$ represents the sample standard deviation of the differences between predicted values and observed values. These individual differences are called residuals when the computations are performed over the data sample that was utilized for estimation, and are called prediction errors when computed out-of-sample. The RMSE serves to aggregate the magnitudes of the errors in predictions for various times into a single measure of predictive power. $RMSE$ is a measure of accuracy, to compare forecasting errors of different models for a particular data and not between datasets, as it is scale-dependent [52].

$RMSE$ is the square root of the average of squared errors. The impact of each error on $RMSE$ is relative to the measure of the squared error; accordingly bigger errors have a disproportionately large effect on $RMSE$. Thusly, $RMSE$ is delicate to outliers. $RMSE$ is a quadratic scoring rule that likewise measures the average magnitude of the error. It's the square root of the average of squared differences between prediction and actual observation.

The mathematical formula of $RMSE$ is :

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

The table 6.1 below shows the RMSE of the predictive models trained and tested. What is clear is that from the CNNs family of models, the best are the 2 and 3 layer networks with very small differences in the $RMSE$ value in the test set. Of the RNNs, the GRU neural network is the best one, achieving an error of about 9. The GRU was the model with the smallest $RMSE$.

Neural Network	*RMSE*
CNN-2 Layers	34.95
CNN-3 Layers	34.60
CNN-4 Layers	42.49
LSTM	15.30
GRU	9.38

TABLE 6.1: RMSE - Model Comparisons

6.1.2 Mean Absolute Error

In statistics, mean absolute error (MAE) is a measure of difference between two continuous variables. Assume X and Y are variables of paired observations that express the same phenomenon. Examples of Y versus X include comparisons of predicted versus observed, subsequent time versus initial time, and one technique of measurement versus an alternative technique of measurement. Consider a scatter plot of n points, where point i has coordinates (x_i, y_i).... Mean Absolute Error (MAE) is the average vertical distance between each point and the $Y = X$ line, which is also known as the One-to-One line. MAE is also the average horizontal distance between each point and the $Y = X$ line.

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It's the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

The mathematical formula of MAE is :

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

The table 6.2 shows the MAE of the predictive models trained and tested. What is clear is that from the CNNs family of models, the best are the 2 and 3 layer networks with very small differences in the MAE value in the test set. Of the RNNs, the GRU neural network is the better than LSTM , achieving an error of about 2.09 against 3.10 of LSTM. The GRU was the model with the smallest MAE.

Neural Network	MAE
CNN-2 Layers	20.79
CNN-3 Layers	20.62
CNN-4 Layers	30.88
LSTM	3.10
GRU	2.09

TABLE 6.2: MAE - Model Comparisons

Chapter 7

Conclusion

7.1 Results Summary

In this section, aggregated results of the models used to predict ETH / USD are presented. The following table shows the comparison in training and validation loss. The best behavior is shown by the LSTM and GRU recurrent neural networks, although there is a gap between training and validation loss. The best of the last two seems to be the GRU, as it has the smallest gap between training and validation loss, and also the smallest validation loss, which means it is better approaching the unknown data.

Neural Network	**Best Training Loss (e $-$ 05)**	**Best Validation Loss (e $-$ 05)**
CNN-2 Layers	7.71	60.40
CNN-3 Layers	7.58	59.32
CNN-4 Layers	22.40	89.87
LSTM	2.68	6.80
GRU	2.72	4.29

TABLE 7.1: Best Training & Validation Loss Results Summary

The $RMSE$ is a quadratic scoring rule which measures the average magnitude of the error. The difference between forecast and corresponding observed values are each squared and then averaged over the sample. Finally, the square root of the average is taken [53]. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. This means the $RMSE$ is most useful when large errors are particularly undesirable. The MAE and the $RMSE$ can be used together to diagnose the variation in the errors in a set of forecasts. The $RMSE$ will always be larger or equal to the MAE; the greater difference between them, the greater the variance in the individual errors in the sample. If the $RMSE = MAE$, then all the errors are of the same magnitude.

The performance metrics for prediction models are shown below.

Neural Network	**RMSE**	**MAE**
CNN-2 Layers	34.95	20.79
CNN-3 Layers	34.60	20.62
CNN-4 Layers	42.49	30.88
LSTM	15.30	3.10
GRU	9.38	2.09

TABLE 7.2: Performance Metrics Results Summary

As is clear, the model with the best performance metrics seems to be the GRU, achieving $RMSE$ equal to 9.38 and MAE equal to 2.09. There is some variation in the magnitude of errors and very large errors are unlikely to have occurred. The average difference between the forecast and the observed closing price of ETH/USD was 2.09.

7.2 Conclusion

In this research, an extensive analysis of cryptocurrencies was presented. Data charts were presented in the data analysis and the price fluctuation was explained. Correlation analysis has shown that starting from the year 2015 up to now in 2018 almost all cryptocurrencies tend to be fully correlated with each other in relation to fluctuations in their prices and the general movement of their price. Three of the most important cryptocurrencies for this era were analyzed, BTC, XRP, ETH.

In particular, we focused on ETH in order to predict its closing price utilizing machine learning algorithms. The machine learning models used are the neural networks CNN and RNN. For this analysis we used data from 2015 to today with a time period of five minutes between two different observations. From the predictive models developed and tested on the above data turned out to be the recurrent neural network, GRU, achieving the smallest and best RMSE as well as MAE.

The GRU neural network predictive model, based on its performance, we assert that can be used for real time prediction of the ETH value. The experiments have shown that in a run of the already trained model, using the 256 past samples, generates 16 predictions in a period of 80 minutes in the future. The issue of overfitting in the models introduced has to be studied further. The topic of cryptocurrencies is constantly on the news and has drawn significant interest from investors, researchers, and financial industry. It appears that cryptocurrencies is a promising research area that requires additional research and development.

7.3 Future Work

There is significant potential for further research regarding the prediction of cryptocurrencies prices in real time. The issue of data overfitting in the proposed models has to be studied further. In addition, the future work should include social network data and google trends for better calibration of the machine learning models used and assess their prediction accuracy. In the near future, we plan to combine sentiment analysis techniques with the GRU model studied in this thesis and analyze the prediction results.

Bibliography

[1] A. Wagner, *Digital vs. virtual currencies ? bitcoin magazine*, 2014. [Online]. Available: https://bitcoinmagazine.com/articles/digital-vs-virtual-currencies-1408735507/.

[2] N. Acheson, 2018. [Online]. Available: https://www.coindesk.com/information/what-is-bitcoin/.

[3] Al-Laham, Al-Tarawneh, and Abdallat, "Development of electronic money and its impact on the central bank role and monetary policy", *Issues in Informing Science and Information Technology*, vol. 6, 2009.

[4] C. McGoogan and M. Field, *What is cryptocurrency, how does it work and why do we use it?*, 2018. [Online]. Available: http://www.telegraph.co.uk/technology/0/cryptocurrency/.

[5] U. G. C. S. Adviser, *Distributed Ledger Technology: beyond block chain*. 2016.

[6] C. Scardovi, *Restructuring and Innovation in Banking*. 2016, p. 36.

[7] *Central banks look to the future of money with blockchain technology trial*, 2016. [Online]. Available: http://www.afr.com/technology/central-banks-look-to-the-future-of-money-with-blockchain-technology-trial-20161117-gss4nd.

[8] T. Economist, "The great chain of being sure about things", *The Economist Newspaper Limited*, 2015.

[9] J. BRITO and A. CASTILLO, *Bitcoin: A Primer for Policymakers*. 2013.

[10] M. Sparkes, *The coming digital anarchy*, 2014. [Online]. Available: http://www.telegraph.co.uk/technology/news/10881213/The-coming-digital-anarchy.html.

[11] J. Kelleher, *Bitcoin*. [Online]. Available: https://www.investopedia.com/terms/b/bitcoin.asp.

[12] P. Bajpai, *Ethereum*. [Online]. Available: https://www.investopedia.com/terms/e/ethereum.asp.

[13] J. A. PULIKKOTTIL, *GREY MATTER DIALOGUES*. PARTRIDGE INDIA, 2017.

[14] *Ether*. [Online]. Available: https://www.ethereum.org/ether.

[15] J. Martindale, *Get to know ripple, a different kind of cryptocurrency*, 2018. [Online]. Available: https://www.digitaltrends.com/computing/what-is-ripple/.

[16] A. Rosic, *Best cryptocurrency exchanges: The ultimate guide*, 2017. [Online]. Available: https://blockgeeks.com/guides/best-cryptocurrency-exchanges/.

[17] *Where can you trade cryptocurrency? list of crypto exchanges 2017*, 2017. [Online]. Available: https://steemit.com/cryptocurrency/dragonsteem/where-can-you-trade-cryptocurrency-list-of-crypto-exchanges.

[18] A. Dr. Steel, "Predictions in financial time series data", Master's thesis, Institute of Technology Blanchardstown, 2014.

[19] M. Briere, K. Oosterlinck, and A. Szafarz, "Virtual currency, tangible return: Portfolio diversification with bitcoin", *Journal of Asset Management*, vol. 16, no. 6, pp. 365–373, 2015. DOI: 10.1057/jam.2015.5.

[20] J. G. De Gooijer and R. J. Hyndman, "25 years of time series forecasting", *International Journal of Forecasting*, vol. 22, no. 3, pp. 443–473, 2006. DOI: 10.1016/j.ijforecast.2006.01.001.

[21] J. Han, M. Kamber, and J. Pei, *Data mining*. Elsevier/Morgan Kaufmann, 2012.

[22] T. C. Mills, *The foundations of modern time series analysis*. Palgrave Macmillan, 2011.

[23] M. K. So and R. S. Chung, "Dynamic seasonality in time series", *Computational Statistics & Data Analysis*, vol. 70, pp. 212–226, 2014, ISSN: 0167-9473. DOI: https://doi.org/10.1016/j.csda.2013.09.010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167947313003253.

[24] L. Yu, S. Wang, and K. K. Lai, *A neural-network-based nonlinear metamodeling approach to financial time series forecasting*, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.asoc.2008.08.001.

[25] S. Walczak, *An empirical analysis of data requirements for financial forecasting with neural networks*, 2001. [Online]. Available: http://dl.acm.org/citation.cfm?id=1289668.1289677.

[26] M. Cubiles-de-la-Vega, R. Pino-Mejias, A. Pascual-Acosta, and J. Munoz Garcia, *Building neural network forecasting models from time series arima models: A procedure and a comparative analysis*, 2002. [Online]. Available: http://dl.acm.org/citation.cfm?id=1293993.1293996.

[27] W. Huang, Y. Nakamori, and S.-Y. Wang, "Forecasting stock market movement direction with support vector machine", *Computers & Operations Research*, vol. 32, no. 10, pp. 2513–2522, 2005. DOI: 10.1016/j.cor.2004.03.016.

[28] G. CHEN, B. ABRAHAM, and G. W. BENNETT, "Parametric and non-parametric modelling of time series - an empirical study", *Environmetrics*, vol. 8, no. 1, pp. 63–74, 1997. DOI: 10.1002/(sici)1099-095x(199701)8:1<63::aid-env238>3.0.co;2-b.

[29] R. Meir, "Nonparametric time series prediction through adaptive model selection", *Machine Learning*, vol. 39, no. 1, pp. 5–34, 2000. DOI: 10.1023/a:1007602715810.

[30] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis*. Wiley, 2008.

[31] R. H. Shumway and D. S. Stoffer, *Time series analysis and its applications*. Springer, 2011.

[32] V. Vapnik, *The nature of statistical learning theory*. Springer, 1995.

[33] V. Vapnik, "An overview of statistical learning theory", *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 988–999, 1999. DOI: 10.1109/72.788640.

[34] A. F. Sheta, S. Elsir M.Ahmed, and H. Faris, "A comparison between regression, artificial neural networks and support vector machines for predicting stock market index", *International Journal of Advanced Research in Artificial Intelligence*, vol. 4, no. 7, 2015. DOI: 10.14569/ijarai.2015.040710.

[35] B. Krollner, B. Vanstone, and G. Finnie, "Financial time series forecasting with machine learning techniques: A survey", *European Symposium on Artificial Neural Networks: Computational and Machine Learning*, 2010. [Online]. Available: http://epublications.bond.edu.au/infotech_pubs/110/.

[36] A. K. Palit and D. Popovic, *Computational Intelligence in Time Series Forecasting*. Springer, 2006.

[37] J. Schmidhuber, "Netzwerkarchitekturen, zielfunktionen und kettenregel (network architectures, objective functions, and chain rule)", PhD thesis, Institut fur Informatik, Technische Universitat Munchen, 1993.

[38] C. Olah, *Understanding lstm networks – colah's blog*, 2015. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[39] *Written memories: Understanding, deriving and extending the lstm - r2rt 2016*, 2016. [Online]. Available: https://r2rt.com/written-memories-understanding-deriving-and-extending-the-lstm.html.

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory", vol. 9, pp. 1735–80, Dec. 1997.

[41] K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation", in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. [Online]. Available: http://www.aclweb.org/anthology/D14-1179.

[42] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling", Dec. 2014.

[43] *Opening price definition & example | investinganswers*. [Online]. Available: http://www.investinganswers.com/financial-dictionary/stock-market/opening-price-6797.

[44] *Closing price definition & example | investinganswers*. [Online]. Available: http://www.investinganswers.com/financial-dictionary/stock-market/closing-price-5488.

[45] I. Staff, *Today's low*. [Online]. Available: https://www.investopedia.com/terms/t/todayslow.asp.

[46] ——, *Today's high*. [Online]. Available: https://www.investopedia.com/terms/t/todayshigh.asp.

[47] ——, *Volume*. [Online]. Available: https://www.investopedia.com/terms/v/volume.asp.

[48] W. M. K. Trochim, J. P. Donnelly, and K. Arora, *Research methods*. Cengage Learning, 2016.

[49] A. Singh Walia, *Activation functions and it's types-which is better?*, 2017. [Online]. Available: https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f.

[50] [Online]. Available: http://math.feld.cvut.cz/mt/txtb/4/txe3ba4f.htm.

[51] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors", vol. arXiv, Jul. 2012.

[52] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy", *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006. DOI: 10.1016/j.ijforecast.2006.03.001.

[53] C. Kamalakannan, L. P. Suresh, S. S. Dash, and B. K. Panigrahi, *Power Electronics and Renewable Energy Systems*. Springer India, 2015.