**Price Prediction of Cryptocurrency**

A Synopsis Submitted

In Partial Fulfilment of the Requirements

For the Course of

# **Major Project**

In

Fourth year – Eight Semester of

**Bachelor of Technology**

Specialization

In

Business Analytics and Optimization

Under

**Ms. Juhi Aggarwal**

By

|  |  |  |
| --- | --- | --- |
| **Aashu Prashar** | **500071245** | **R10318203** |
| **Hrithvik Sharma** | **500068761** | **R103218059** |
| **Nisheet Agrawal** | **500070404** | **R103218191** |
| **Ritik Karir** | **500070985** | **R103218195** |



DEPARTMENT OF INFORMATICS

SCHOOL OF COMPUTER SCIENCE

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES, BIDHOLI, DEHRADUN, UTTRAKHAND, INDIA

**Abstract**

Bitcoin and other prominent cryptocurrencies have gained much attention over the last several years. Globally known as digital coin and virtual currency, this cryptocurrency is gained and traded within the block chain system. The block chain technology adopted in using cryptocurrency has raised eyebrows within the banking sector, government, stakeholders, and individual investors. The rise of the cryptocurrency within this decade since the inception of Bitcoin in 2009 has taken the market by storm. Cryptocurrency is anticipated as the future currency that might replace the current paper currency worldwide.

In this project, we are predicting the rates of cryptocurrency. We are proposing a system that can predict whether it is increasing or decreasing, and accordingly getting the final cost after the change and analysing the trend of the graph containing a change in rates of cryptocurrency. During the making of this project, we are building up a model of an Artificial Neural Network which will help us in doing the prediction.

**Introduction**

Due to the rapid development of information and communication technologies, many activities in our daily life have been merged online, and they have become more flexible and more effective. The huge growth in the number of online users has activated virtual word concepts and created a new business phenomenon which is a cryptocurrency to facilitate financial activities such as buying, selling, and trading. Cryptocurrency represents valuable and intangible objects which are used electronically in different applications and networks such as online social networks, online social games, virtual worlds, and peer-to-peer networks. The use of virtual currency has become widespread in many different systems in recent years.

Cryptocurrency, an encrypted, peer-to-peer network for facilitating digital barter, is a technology developed twelve years ago. Bitcoin, the first and most popular cryptocurrency followed by Ethereum, paving the way as a disruptive technology to long standing and unchanged financial payment systems that have been in place for many decades.

Bitcoin is a digital currency or cryptocurrency which has emerged as the most successful crypto currency since its appearance back in 2009. It is owned and controlled by its users, peer to peer and has no central control like traditional currencies. It is enabled using a technology called Blockchain, which is a new kind of database or list of encrypted digital blocks linked together in a secure way. Bitcoin was the first real world application of blockchain.

**Objective**

* Effective prediction systems indirectly help traders by providing supportive information such as the future market direction.
* With the help using Artificial Neural Networks through this project, we aim to propose a novel method for the prediction of the stock market.
* Cryptocurrency prediction aims to determine the future movement of the value of a financial exchange. The accurate prediction of share price movement will lead to more profit investors can make. Thus, this project can be directly beneficial to anyone who is new to the realm of Cryptocurrency as well as to someone who knows it well.
* Through this project we are going to present a more feasible method to predict the price of Cryptocurrency with higher level of accuracy.
* A correct prediction of Cryptocurrency can lead to huge profits for interested people.
* To deliver the end-users a hassle-free and easy-going experience.

**Artificial Neural Network**

Artificial Neural Networks are one of the main tools used in machine learning. As the neural part of their name suggests, they are brain-inspired systems which are intended to replicate the way that we humans learn. Neural networks consist of input and output layers, as well as (in most cases) a hidden layer consisting of units that transform the input into something that the output layer can use. They are excellent tools for finding patterns which are far too complex or numerous for a human programmer to extract and teach the machine to recognize.

For a basic idea of how a deep learning neural network learns, imagine a factory line. After the raw materials (the data set) are input, they are then passed down the conveyer belt, with each subsequent stop or layer extracting a different set of high-level features. If the network is intended to recognize an object, the first layer might analyse the brightness of its pixels.

The next layer could then identify any edges in the image, based on lines of similar pixels. After this, another layer may recognize textures and shapes, and so on. By the time the fourth or fifth layer is reached, the deep learning net will have created complex feature detectors. It can figure out that certain image elements (such as a pair of eyes, a nose, and a mouth) are commonly found together.

Once this is done, the researchers who have trained the network can give labels to the output, and then use backpropagation to correct any mistakes which have been made. After a while, the network can carry out its own classification tasks without needing humans to help every time.

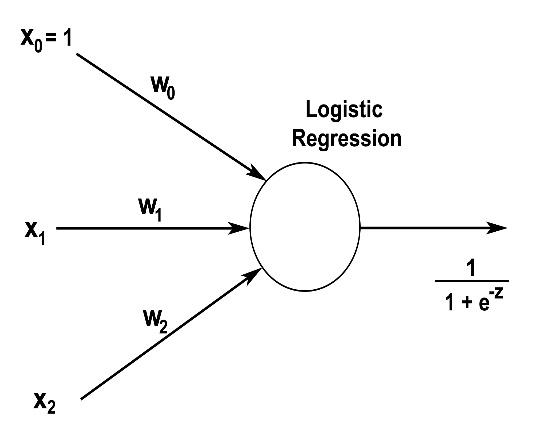
**Why do we need NN?**

Neural Networks have been around even before machine learning gained pace. But they were thought to be computationally too heavy and hence, brushed aside.

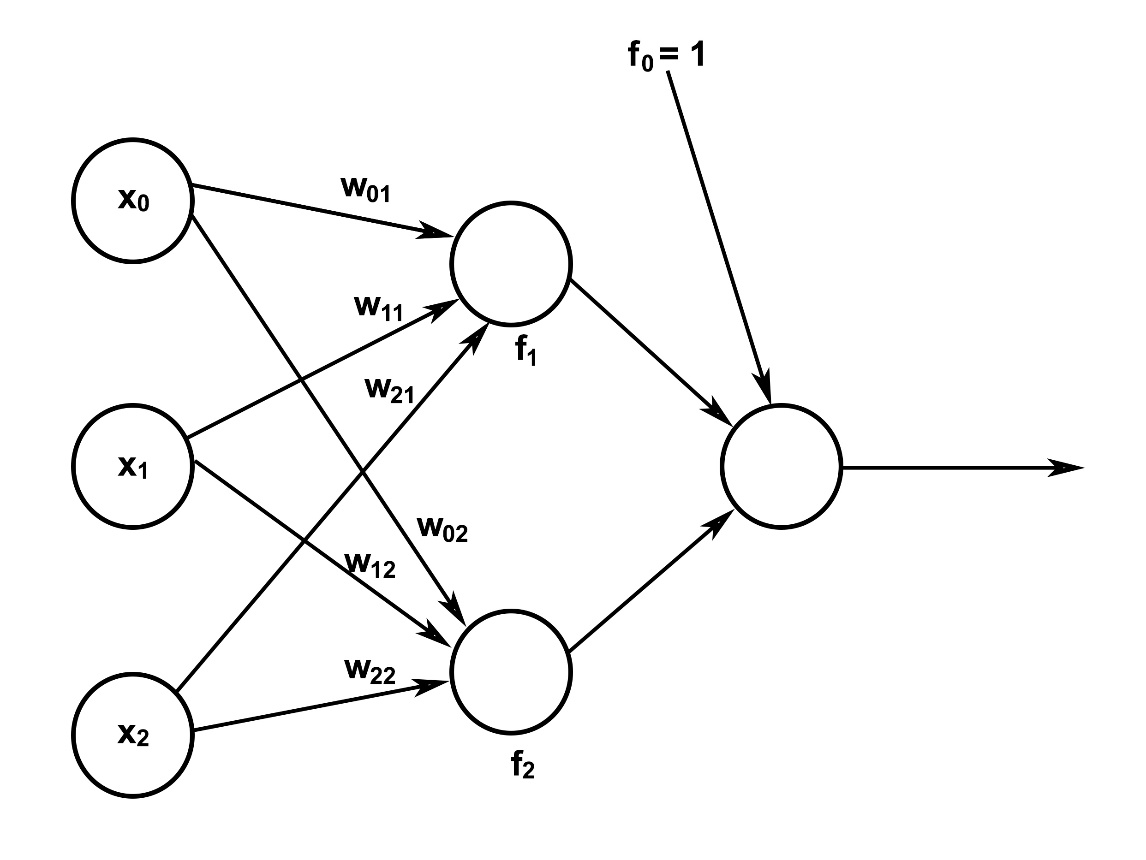
A problem we faced during Logistic Regression was that, even though the decision function (Sigmoid) was non-linear, we got a linear decision boundary. We fixed this problem by adding dummy data with higher powers.

To do that, we had to experiment and decide the degree of features we needed to add. Our decision boundary should be such, that it performs this task on its own.

Logistic regression had the following structure:



**The intuition behind Neural Networks is a follow:**



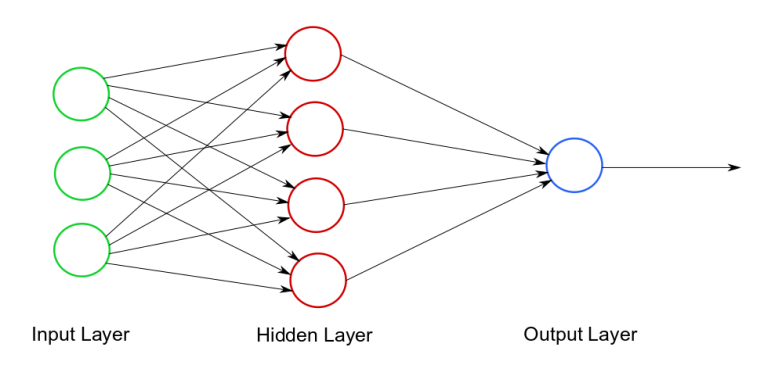
So, here the final output will not be linear with respect to 𝑥1, 𝑥2, 𝑥0x1, x2, and x0. The functions 𝑓1, 𝑓2f1, f2 need not necessarily be Sigmoid. We can choose any function. Using this method, we can create quite interesting decision boundaries without applying the dummy feature method.

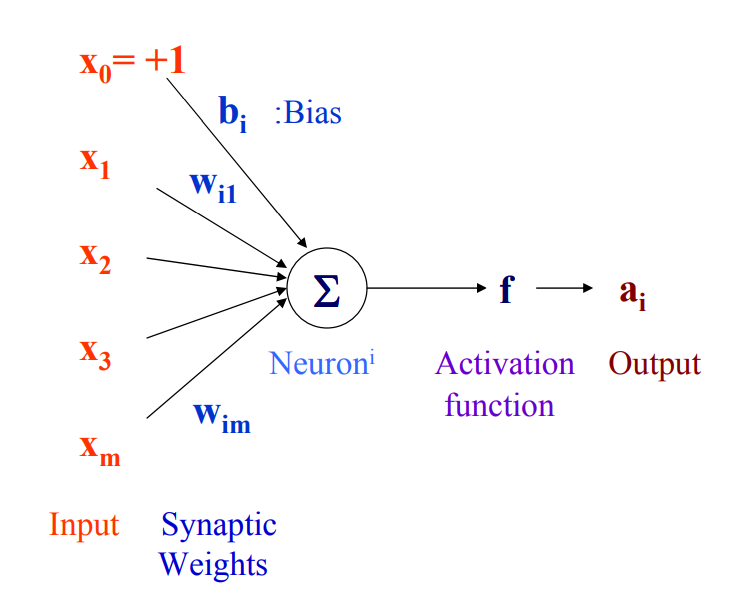
**Terminologies**

1. **Neuron**: A single unit in any layer is called neuron.
2. **Input Layer**: The Input layer communicates with the external environment that presents a pattern to the neural network. Its job is to deal with all the inputs only. The input layer should represent the condition for which we are training the neural network. Every input neuron should represent some independent variable that has an influence over the output of the neural network.
3. **Hidden Layer**: The hidden layer is the collection of neurons which has activation function applied on it and it is an intermediate layer found between the input layer and the output layer. Its job is to process the inputs obtained by its previous layer. So it is the layer which is responsible extracting the required features from the input data.
4. **Output Layer**: The output layer of the neural network collects and transmits the information accordingly in way it has been designed to give. The pattern presented by the output layer can be directly traced back to the input layer. The number of neurons in output layer should be directly related to the type of work that the neural network was performing.

**Weights for each neuron will be found using some algorithm. What we need to decide is:**

1. How many hidden layers we want?
2. How many neurons in each layer?
3. Function to be applied over hidden and output layer.

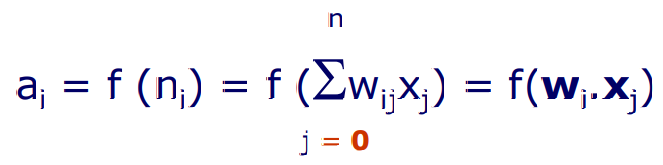




The ability of the neural network to provide useful data manipulation lies in the proper selection of the weights. This is different from conventional information processing.

**Bias**

Bias can be incorporated as another weight clamped to a fixed input of +1.0. This extra free variable (bias) makes the neuron more powerful.



**ROC (Receiver Operating Characteristic curve) curve**

An **ROC curve** (**receiver operating characteristic curve**) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:

1. True Positive Rate
2. False Positive Rate

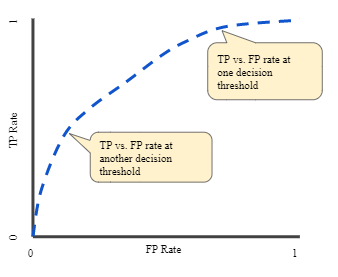
**True Positive Rate** (**TPR**) is a synonym for recall and is therefore defined as follows:

TPR=TP/TP+FN

**False Positive Rate** (**FPR**) is defined as follows:

FPR=FP/FP+TN

An ROC curve plots TPR vs. FPR at different classification thresholds. Lowering the classification threshold classifies more items as positive, thus increasing both False Positives and True Positives. The following figure shows a typical ROC curve.



**Figure 1. TP vs. FP rate at different classification thresholds.**

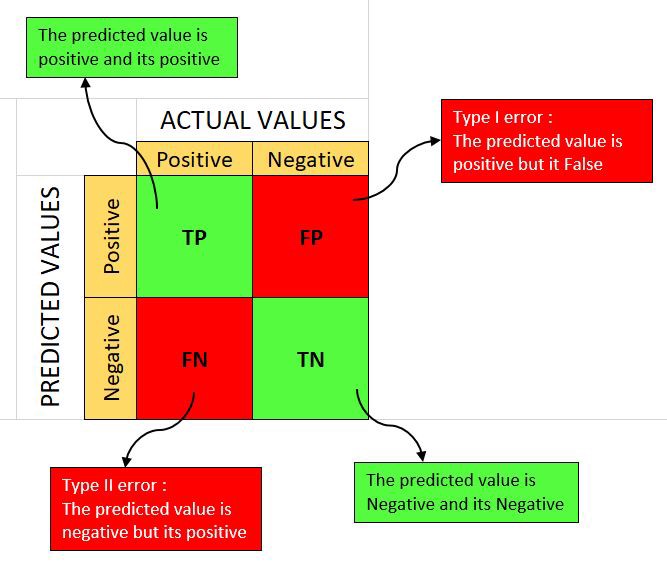
**Confusion Matrix**

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. **The confusion matrix shows the ways in which your classification model is confused when it makes predictions.**

It gives us insight not only into the errors being made by our classifier but more importantly the types of errors that are being made. It is this breakdown that overcomes the limitation of using classification accuracy alone.

**There are some terms that one must know regarding confusion matrices:**

1. **True Positives:** This is the number of samples predicted positive which were actually positive.
2. **True Negatives:** This is the number of samples predicted negative which were actually negative.
3. **False Positives:** This is the number of samples predicted positive which were **not** actually positive.
4. **False Negatives:** This is the number of samples predicted negative which were **not** actually negative.



**How to Calculate a Confusion Matrix?**

Below is the process for calculating a confusion Matrix:

1. We need a test dataset or a validation dataset with expected outcome values.
2. We make a prediction for each row in our test dataset.
3. From the expected outcomes and predictions count:
4. The number of correct predictions for each class.
5. The number of incorrect predictions for each class, organized by the class that was predicted.

These numbers are then organized into a table, or a matrix as follows:

1. **Expected down the side**: Each row of the matrix corresponds to a predicted class.
2. **Predicted across the top**: Each column of the matrix corresponds to an actual class.
3. The counts of correct and incorrect classification are then filled into the table.
4. The total number of correct predictions for a class go into the expected row for that class value and the predicted column for that class value.
5. In the same way, the total number of incorrect predictions for a class go into the expected row for that class value and the predicted column for that class value.

**Methodology**

1. First, we need to import the data of cryptocurrency rates from any open-source website. In our case, we’ve taken the data from

**THE COINDESK 20 (**[**https://www.coindesk.com/coindesk20**](https://www.coindesk.com/coindesk20)**)**

1. For better understanding, we took the data for a period of 1 year (which can be modified according to the preference).
2. Next, we need to clean the data first, as the model can’t work upon the **NULL values**. In our case, the data received is already cleaned and standardized.
3. Importing the data in the code for building the model. For that, we need to split the data into **test and train** and proceed further according to our requirements.
4. If the testing comes out to be accurate, then we are ready to train our model so as to get predictions, i.e., prediction of rates (increasing or decreasing) of cryptocurrency in last 24 hours.
5. Lastly, we will have **ROC curve, Confusion Matrix, Graph of the final trend** of the same.

**References:**

1. <https://www.datasciencecentral.com/profiles/blogs/artificial-neural-network-ann-in-machine-learning>
2. <https://machinelearningmastery.com/confusion-matrix-machine-learning/>
3. <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>
4. Jani, Shailak. (2018). The Growth of Cryptocurrency in India: Its Challenges & Potential Impacts on Legislation. 10.13140/RG.2.2.14220.36486.
5. Fern´andez-Villaverde, J. and D. Sanches, (2016). “Can currency competition work?”National Bureau of Economic Research No. w22157.
6. Swati Chaturvedi
7. 3:30 PM
8. Dourado, Eli & Brito, Jerry. (2014). Cryptocurrency. The New Palgrave Dictionary of Economics. 10.1057/978-1-349-95121-5\_2895-1.
9. Giudici, G., Milne, A. & Vinogradov, D. Cryptocurrencies: market analysis and perspectives. J. Ind. Bus. Econ. 47, 1–18 (2020). <https://doi.org/10.1007/s40812-019-00138-6>
10. <https://www.coindesk.com/10-must-read-cryptocurrency-research-papers-from-2015>