Exploratory analysis on Machine Learning Models

Student number:	**Q15937305**
Name:	**Krishnaben Baldha**

Table of Contents

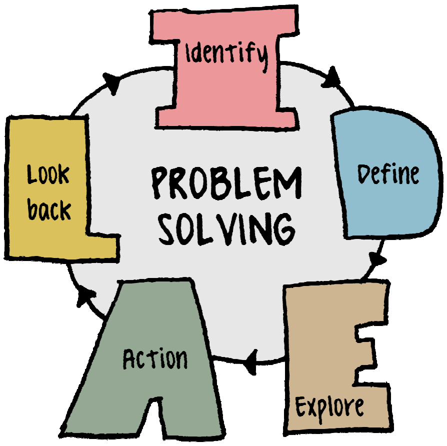
Introduction

Cryptocurrency is a virtual currency, in a form of payment created using encryption algorithms. currency means medium of purchase of goods and services. it also called as private currency there is so many cryptocurrencies are available for example bitcoin, Ethereum, ripple, dogecoin etc.

cryptocurrency is decentralized Unlike other payment systems that banks or governments are administering, cryptocurrencies are decentralised, uncontrolled, and immune to third-party influence. For this reason, a bitcoin transaction never fails there could be many reasons for why denial's transaction failed like the may have exceeded his transfer for his account must have been hacked or there could have been some technical issue with his bank server but on the other hand cryptocurrency changes almost no or very low transaction cost there is no limit for making transactions.

When we send or receive any cryptocurrency from other person then it must be stored somewhere and this is stored in database, the storage of cryptocurrencies is called ledger this ledger is maintained by so many computers. it is peer to peer network so many ledgers are control this network. when we store this ledger in only one place then it may misuse of it so there is one revolutionary solution. The advantages of cryptocurrencies are cheaper and faster money transfers and because of single point failure the decentralized systems do not collapse. The disadvantages of cryptocurrencies are the energy consumption for mining activities is high, and use in criminal activities and their price volatility.

Problem Identification



The ability to anticipate bitcoin prices can help both regulators and financial experts assess the behavior of cryptocurrency markets and help cryptocurrency investors make well-informed investment choices to optimize profits.The network cannot use information from the distant past and cannot learn patterns with long dependencies. To overcome these issues, the long short-term memory introduced here. It is a special type of recurrent neural network and learn long-term patterns; it detects a pattern. Long short-term memory performs better than others and detects a pattern. They have helped us for example in music generation and a bunch of difference in other tasks done with audio.

Dataset

Crypto currency live data has been collected from finance which has historical data of different types of crypto currency. This dataset contains 7 features which are described below:

Date	Date of the day
Open	Price of that particular currency on the beginning of the day.
High	The highest price of coin on that particular day.
Low	The Lowest price of coin on that particular day.
Close	Coin price of that particular currency at the end of the day
Volume	Coin traded quantity on that specific day between buyer and seller.
Adj Close	the closing price after all applicable splitting and dividend payments have been taken into account.

Data Preprocessing

To transform unprocessed data into useful and effective data for pre-processing, Big amount of data analytics were used. It must be transformed into a useful pattern for certain information, especially noisy information, may not have any significance. This problem was solved using a data purification technique. Before data mining, massive data volumes are managed using techniques for reducing data also we have to check null values. If any found then we have to replace of drop that data information. This encourages data gathering and contributes to accurate results. This increases data storage capacity while decreasing the cost of data analysis. I utilised the normalisation technique to increase model accuracy and to make the data meaningful. Using MinMaxScaler, features are transformed by scaling each feature to fit within a specific range.

Value Preposition

As per analyzation, I found that these models useful for time series data forecasting:1)K-Means Clustering 2) Support Vector Machine 3) Random Forest Regressor 4) Long Short Term Memory(LSTM).

I have used LSTM for my model prediction.

LSTM – Long Short Term Memory

LSTM are primarily used for learning a classify sequential data and process because these networks are long period learning method and it is a dependencies in between time steps of data. commonly LSTM includes sentiment analysis, language modeling, speech recognition, and video analysis**.** LSTM network which is right network for the bind a classifying and processing and making predictions on time series data, because in the time series there is lags of unknown duration in between important events. Most advanced and useful model are introduced in the form of LSTM.It allows the neural network to remember the information that it needs to keep hold of the context but also to forget the information that is well no longer applicable.

We can utilise recurrent nets to convert vectors to scalars, and by adding sequences to the mix, we may create a variety of architectures that can be applied to different problems.One design is a vector to sequence model, where we take a vector and create a sequence of the necessary length. Current research using this is picture captioning, where the input can be a vector representation of an image and the output is a series of words that describe that image. Sequence to vector models, which take word sequences as input and produce fixed-length words as output, is the second architecture we discussed. In a sentiment analysis common use case, the words of a movie or product review could be the input, as well as the output could be a two-dimensional vector indicating whether the review was positive or negative.

The encoder/decoder architecture is a distinct kind of architecture that accepts an input sequence and outputs a sequence of various lengths from the input. The encoder, which turns a sequence into a vector, is the first component of the design. The decoder, which turns a vector into a sequence, is the second.

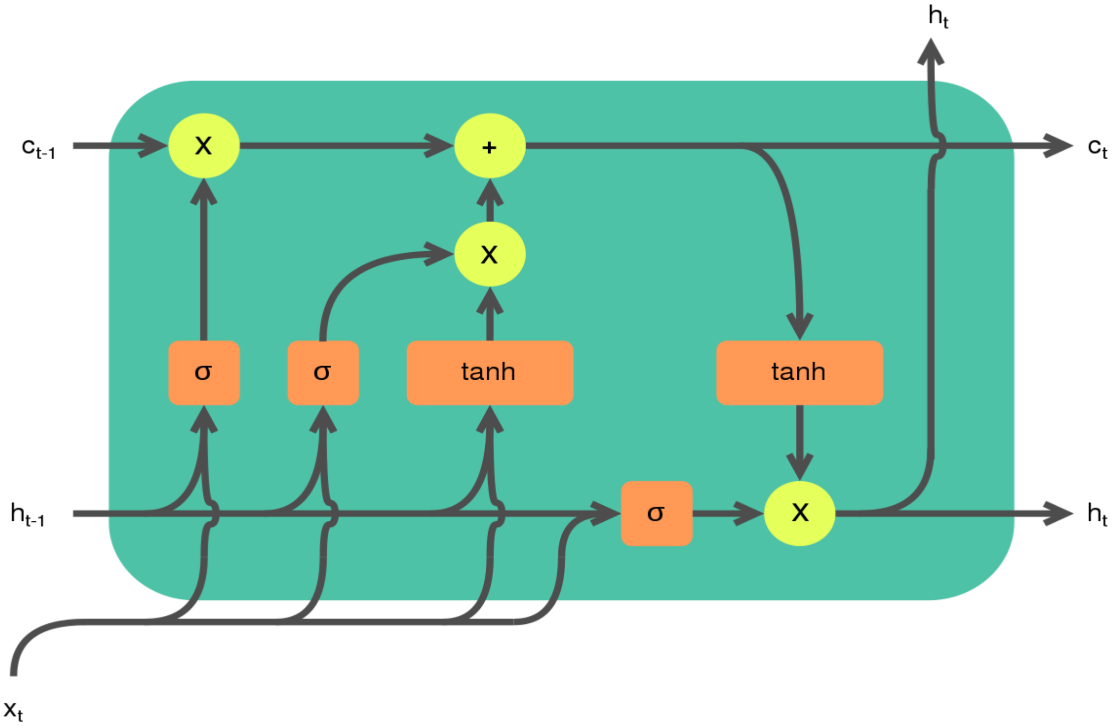
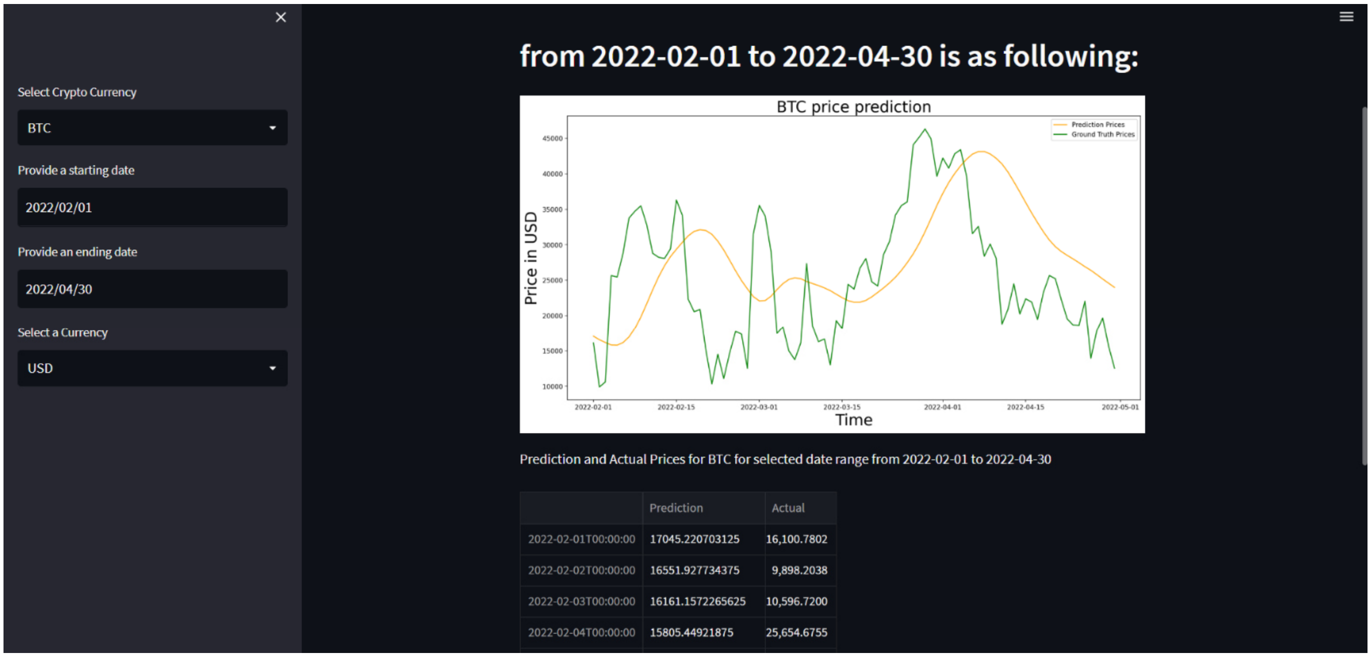


Figure: Long Short-Term Memory

GUI:

I have used Streamlit for creating GUI. It will shows as follows:



Select Crypto Currency: BTC

Provide a starting date: 2022/02/01

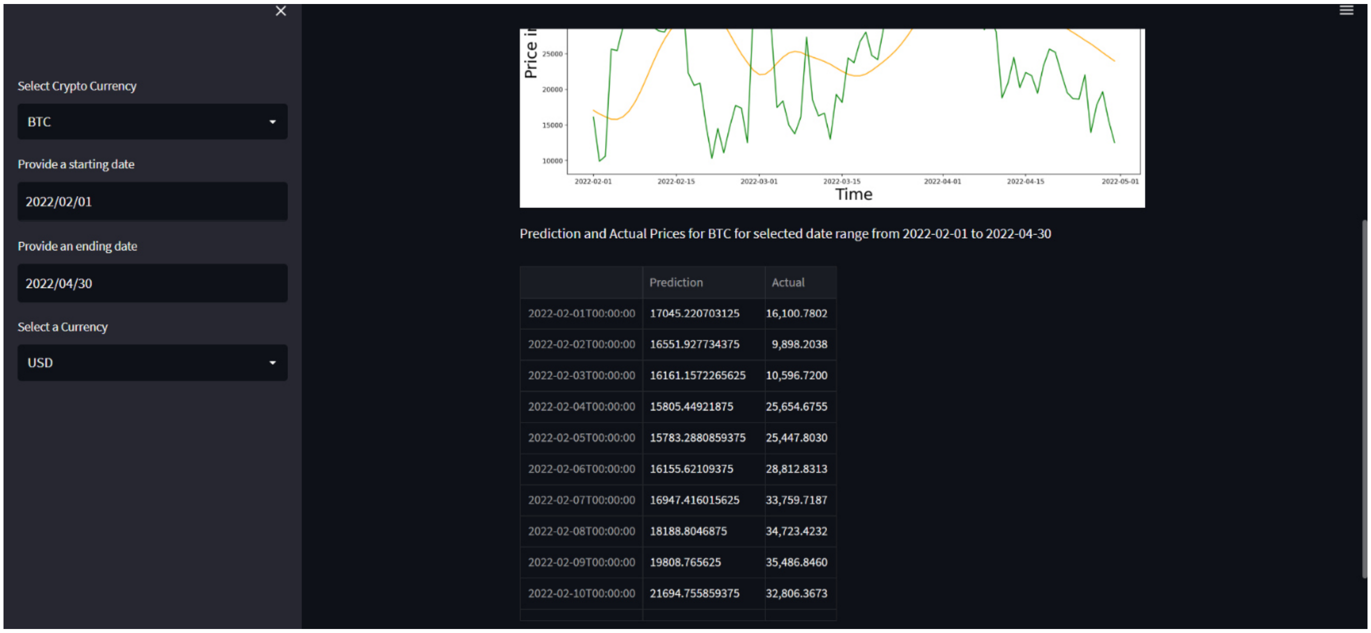
Provide an ending date: 2022/04/30

Select a Currency: USD

from 2022-02-01 to 2022-04-30 is as following:

Prediction and Actual Prices for BTC for selected date range from 2022-02-01 to 2022-04-30

	Prediction	Actual
2022-02-01T00:00:00	17045.220703125	16,100.7802
2022-02-02T00:00:00	16551.927734375	9,898.2038
2022-02-03T00:00:00	16161.1572265625	10,596.7200
2022-02-04T00:00:00	15805.44921875	25,654.6755



Select Crypto Currency: BTC

Provide a starting date: 2022/02/01

Provide an ending date: 2022/04/30

Select a Currency: USD

Prediction and Actual Prices for BTC for selected date range from 2022-02-01 to 2022-04-30

	Prediction	Actual
2022-02-01T00:00:00	17045.220703125	16,100.7802
2022-02-02T00:00:00	16551.927734375	9,898.2038
2022-02-03T00:00:00	16161.1572265625	10,596.7200
2022-02-04T00:00:00	15805.44921875	25,654.6755
2022-02-05T00:00:00	15783.2880859375	25,447.8030
2022-02-06T00:00:00	16155.62109375	28,812.8313
2022-02-07T00:00:00	16947.416015625	33,759.7187
2022-02-08T00:00:00	18188.8046875	34,723.4232
2022-02-09T00:00:00	19808.765625	35,486.8460
2022-02-10T00:00:00	21694.755859375	32,806.3673

Accuracy Estimation:

The R square Rating, Mean Absolute Error, Root Mean Squared Error, and Mean Square Error factors are used to assess the numbers.

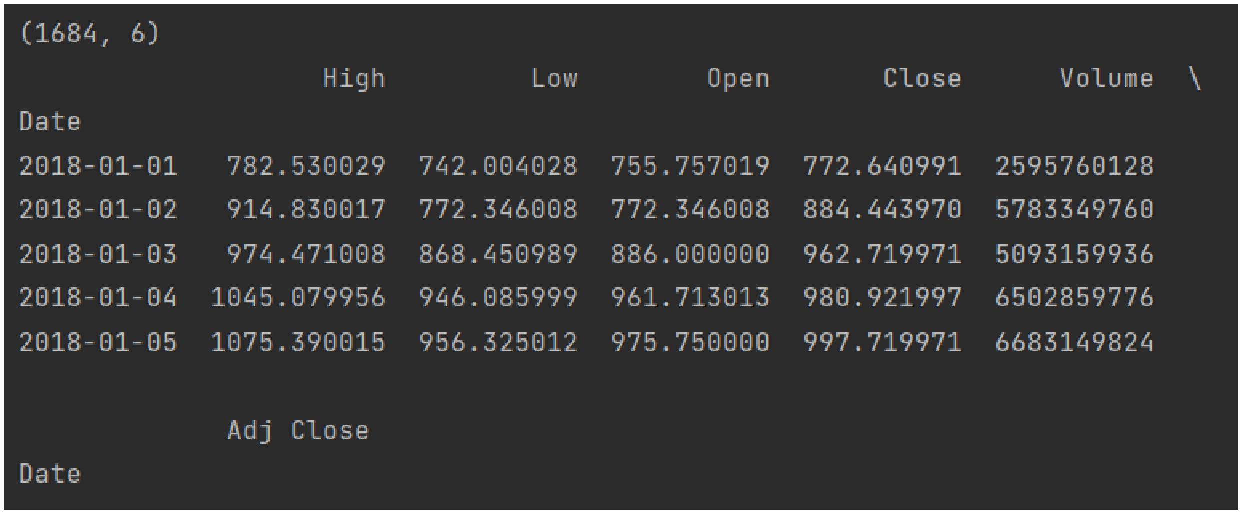
R2: The r2 score is expressed as a percentage between 0% and 100%. The MSE is comparable to, but distinct from, the MSE. A statistical measure of a regression model's efficiancy is R-squared. R-square is best when its value is 1.

RMSE: The standard deviation of our error appropriation can be accurately predicted by the RMSE. The standard deviation from deposits is known as the root mean square error (RMSE). The distance between the information locations on the relapse line is measured by deposits; the communication between these buildups is estimated by RMSE. It reveals how closely the data is packed all along line of greatest fit as a result. To accept exploratory observations in climatology, estimating, and relapse inquiry, the root implies square fallacy is frequently used.

MAE: The absolute error is the sum of all calculation errors. That which separates the calculated value from the "real" value is the difference.The Mean Absolute Error is the total average of all errors (MAE). In the test data set, this prediction error is determined for each entry. Afterward, change the negative number to a positive one if necessary. Calculating the absolute value of any error allows for this. Finally, find the average of all absolute mistakes found. the mean of all errors in absolute terms.

Performance Analysis and Discussion

This research utilised transaction data and measurements of the number of closures at a particular moment to forecast the price of a cryptocurrency. The information gathered is based on yahoofinance's transaction history for the calendar year 2018. The following properties make up the collection components' 1684 data records: symbol, date, open, high, low, close, volume, and adj. close. A sample of the dataset is shown in Table 1. The remaining columns in the dataset have been removed since, aside from "close," they are not as important for forecasting the "close" price.



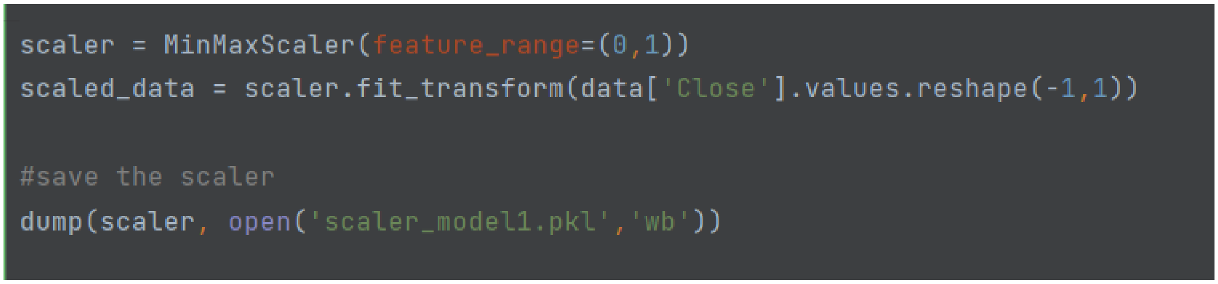
```
(1684, 6)
                    High         Low        Open       Close      Volume  \
Date
2018-01-01    782.530029  742.004028  755.757019  772.640991  2595760128
2018-01-02    914.830017  772.346008  772.346008  884.443970  5783349760
2018-01-03    974.471008  868.450989  886.000000  962.719971  5093159936
2018-01-04   1045.079956  946.085999  961.713013  980.921997  6502859776
2018-01-05   1075.390015  956.325012  975.750000  997.719971  6683149824

              Adj Close
Date
```

Then data will be loaded from start to end date which is (2018-01-01) to till now and data will also be saved in 'cryptoCurrency.csv'. For checking if there is any missing values using : data.isnull().sum()

After that, Created minmax scaler to pre-process the data,

Scaler only works NumPy array so, converting pandas series into NumPy array using data['Close'].values and then reshaping it using .reshape(-1,1) to make it 2D from 1D NumPy array:

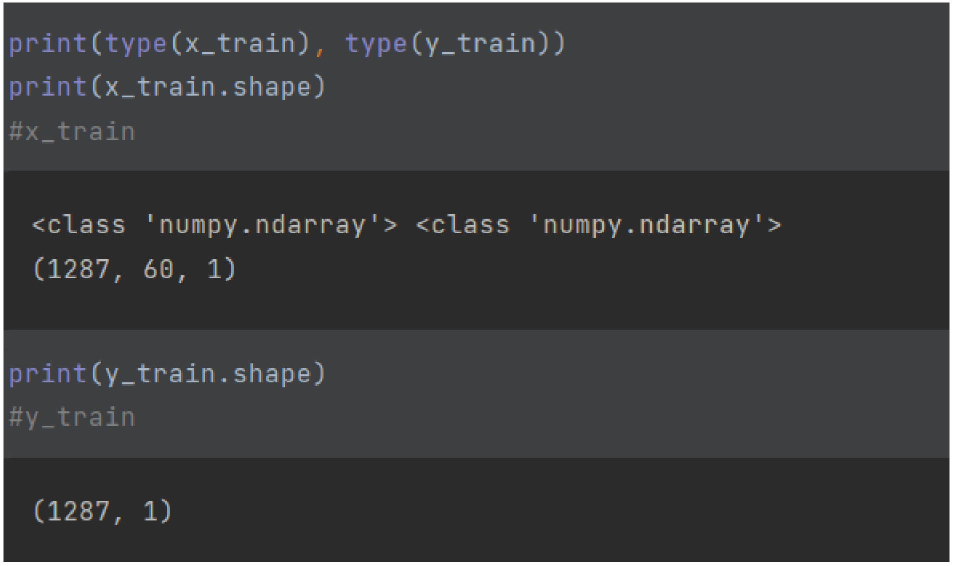


```
scaler = MinMaxScaler(feature_range=(0,1))
scaled_data = scaler.fit_transform(data['Close'].values.reshape(-1,1))

#save the scaler
dump(scaler, open('scaler_model1.pkl','wb'))
```

Splitting the data:

Split the data by Training = 0.8 or 80% and Testing = 0.2 or 20%. We take past 60 days and predict 1 future day.Means, we are looking 60 days in back to predict 61th day in future



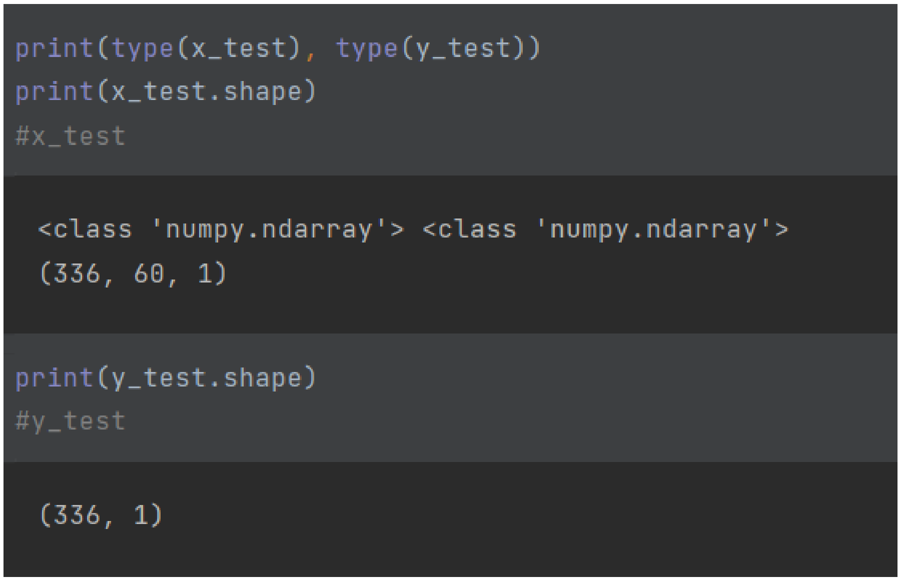
```
print(type(x_train), type(y_train))
print(x_train.shape)
#x_train
```

```
<class 'numpy.ndarray'> <class 'numpy.ndarray'>
(1287, 60, 1)
```

```
print(y_train.shape)
#y_train
```

```
(1287, 1)
```

Preparing for test data:



```
print(type(x_test), type(y_test))
print(x_test.shape)
#x_test
```

```
<class 'numpy.ndarray'> <class 'numpy.ndarray'>
(336, 60, 1)
```

```
print(y_test.shape)
#y_test
```

```
(336, 1)
```

Now, we are creating model for prediction.First, Creating a neural network :
Which has these layers : LSTM, dropout, LSTM, dropout, LSTM, Dense.

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)            Output Shape           Param #
=================================================================
lstm_12 (LSTM)          (None, 60, 50)         10400
_________________________________________________________________
dropout_12 (Dropout)       (None, 60, 50)         0
_________________________________________________________________
lstm_13 (LSTM)          (None, 60, 50)         20200
_________________________________________________________________
dropout_13 (Dropout)       (None, 60, 50)         0
_________________________________________________________________
lstm_14 (LSTM)          (None, 50)            20200
_________________________________________________________________
dropout_14 (Dropout)       (None, 50)            0
_________________________________________________________________
dense_4 (Dense)          (None, 1)             51
=================================================================
Total params: 50,851
Trainable params: 50,851
Non-trainable params: 0
```

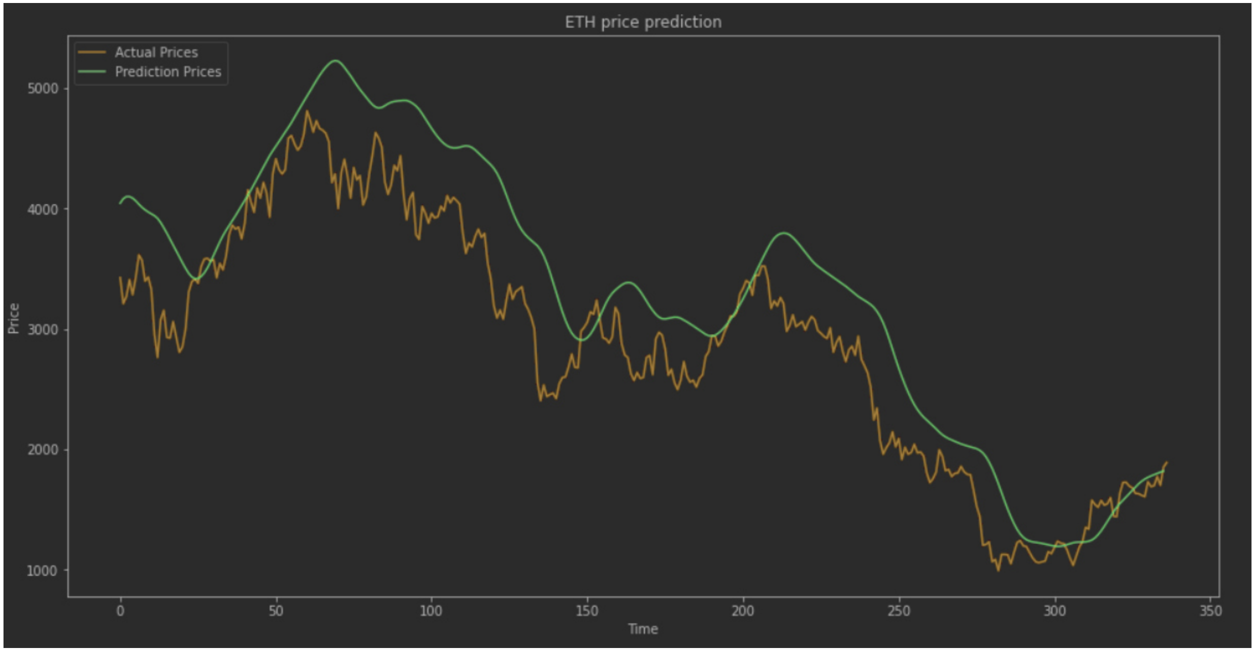
For train the model we keep validation data.Means Validation_split=0.1 or 10, training =0.9 or 90%.
Batch_size=32 means total sequence.for example, In this case 1287/32 = 40.21 so batches will be of around 37.

Model training's output :

```
Epoch 1/12
37/37 [==============================] - 8s 90ms/step - loss: 5.7859e-04 - val_loss: 0.0231
Epoch 2/12
37/37 [==============================] - 2s 66ms/step - loss: 3.6387e-04 - val_loss: 0.0077
Epoch 3/12
37/37 [==============================] - 3s 75ms/step - loss: 4.5223e-04 - val_loss: 0.0083
Epoch 4/12
37/37 [==============================] - 3s 77ms/step - loss: 3.5327e-04 - val_loss: 0.0080
Epoch 5/12
37/37 [==============================] - 3s 77ms/step - loss: 4.6378e-04 - val_loss: 0.0072
```

Save the model to use it later in prediction method or can be used in other modules too, so for that we are using , from keras.models import load_models.After evaluation, It gives loss value and compare it with training loss values from different epoch.

Then,Plot the Prediction(prediction_prices) vs Y_test (actual_prices)



We are also creating methods to save more time and improve efficiency.: input_set and predict.

LSTM model improves demand forecasters' accuracy, which helps the business make better decisions.

Evaluation:

As per the study of the real-time prediction of prices, the previous process does not contain enough information and solution for forecasting price changes. Business people or investors who find risk while investing. To overcome these problems of people-related cryptocurrency price prediction, we come up with a machine learning-based approach to price prediction for a financial institution. The expected system contains the Y Finance API to fetch stock and financial-related data. We can use this system on various cryptocurrencies like Bitcoin, Litecoin, Ripple, Tether, IOTA, Monaro, etc. The results show the presented system is accurate the performance of price prediction is higher than other algorithms. The price prediction effectiveness of the different cryptocurrencies gives the ability of complete power computing to worldwide people.

Limitations:

- long short-term memory models could not be found at fault openly
- long short-term memory struggles to learn long-term trends.
- It requires a large amount of data to train than other models

- long short-term memory does not actually forecast. Instead, they use lagged values in order to make forecasts of the immediately following observation.
- Fixed-size input is required for training.

Conclusion:

We have successfully implemented a good time series forecasting system for various cryptocurrencies. In this system, users can analyze the past 5 years of the database. Accordingly, business users or stock brokers in the market can easily predict the rates or prices of cryptocurrencies. The main aim of the system is the analyze price of the past cryptocurrency and develop predictive values for the future. In this system, we cover almost 20 various cryptocurrencies like Bitcoin, Litecoin, Ripple, Tether, IOTA, Monaro, Ethereum, Binance coin, etc. The results show the launched system is accurate the performance of price prediction is higher than other algorithms.

References:

1. A New Forecasting Framework for Bitcoin Price with LSTM - IEEE International Conference on Data Mining Workshops (ICDMW) Year: 2018 | Conference Paper | Publisher: IEEE - https://ieeexplore.ieee.org/document/8637486

2. A Cryptocurrency Prediction Model Using LSTM and GRU Algorithms - IEEE/ACIS 6th International Conference on Big Data, Cloud Computing, and Data Science (BCD) Year: 2021 | Conference Paper | Publisher: IEEE - https://ieeexplore.ieee.org/document/9581397

3. A LSTM-Method for Bitcoin Price Prediction: A Case Study Yahoo Finance Stock Market - International Conference on Electrical Engineering and Computer Science (ICECOS) Year: 2019 | Conference Paper | Publisher: IEEE - https://www.researchgate.net/publication/339092042_A_LSTM-Method_for_Bitcoin_Price_Prediction_A_Case_Study_Yahoo_Finance_Stock_Market

4. Article An Advanced CNN-LSTM Model for Cryptocurrency Forecasting Ioannis E. Liveris 1, *, Niki Kiriakidou 2, Stavros Stavroyiannis 3 and Panagiotis Pinellas 1 - https://www.mdpi.com/2079-9292/10/3/287

5. Understanding LSTM Networks - http://colah.github.io/posts/2015-08-Understanding-LSTMs/

6. https://www.analyticsvidhya.com/blog/2021/01/understanding-architecture-of-lstm/

7. https://www.analyticsvidhya.com/blog/2021/07/lets-understand-the-problems-with-

8. Yiu, T., 2019. *Understanding Random Forest.* [Online]
Available at: https://towardsdatascience.com/understanding-random-forest-58381e0602d2

9. SOL Page : https://learn.solent.ac.uk/course/view.php?id=44116&section=5#tabs-tree-start