CryptoCurrency Analysis and Prediction

Investigating the Relationship between Bitcoin and Stock Market and Prediction using AI Algorithms

Wei Min Chen (400156352)
B.Tech - Manufacturing Engineering Technology
McMaster University
Hamilton, Canada
Chenw86@mcmaster.ca

Yihuan Zhang (400350335)
B.Tech - Software Engineering Technology
McMaster University
Hamilton, Canada
zhany870@mcmaster.ca

Abstract

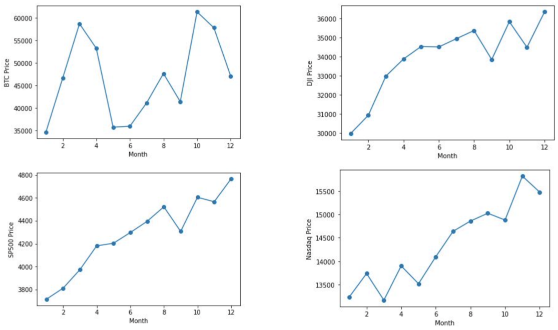
Unlike Stock and Real Estate, CryptoCurrency is a new way of investment which is famous for its anonymy, decentralisation, and high-frequency. CryptoCurrency as a kind of digital asset, shares some similar features with other tradable asset classes. In this report, we take the example of BitCoin, the earliest and largest digital currency in the world, to investigate its price trend with the global economics, especially in the US stock markets. Moreover, this report will explore further, using the AI algorithm, the price trends in the future, not only for BitCoin, but any other assets.

Introduction

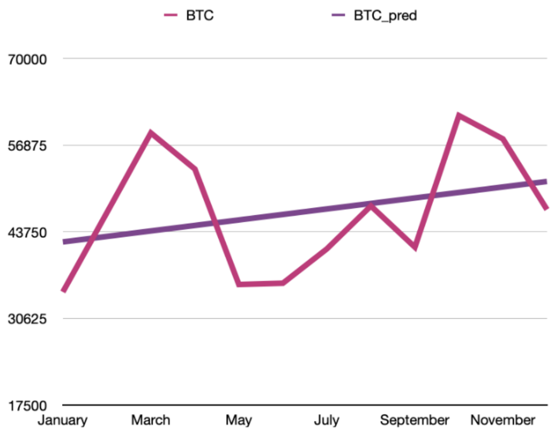
BitCoin was firstly introduced in 2008, by Satoshi Nakamoto. Its market cap, in general, has increased rapidly in the past decade. As more and more individuals and financial organisations are investing in BitCoin, consequently, its market value shows positive correlation with stock markets in the past few years. We therefore, take the market data in the past year 2021. Since the US stock market is the largest one in the world,we use the indexes such S&P500 and Nasdaq, to represent the stock market. Given the condition that CryptoCurrency trading happens regardless of weekend, holidays, and hours, whereas Stock markets are only open certain hours in a day, and certain days in a week, we have to unify the sampling rate for all the assets, either daily, weekly, or monthly. According to the data we have in our research, monthly performance charts are the one we could find for most of them. Therefore, the first part of this report will use monthly performance data in 2021, to compare and analyse the relationship between BitCoin and US stock market indexes in 2021.

Relationship Analysis

To analyse the data, we take the monthly data of BitCoin average price, S&P 500 and Dow Jones monthly values from the spreadsheet, and then plot them into different charts in python.

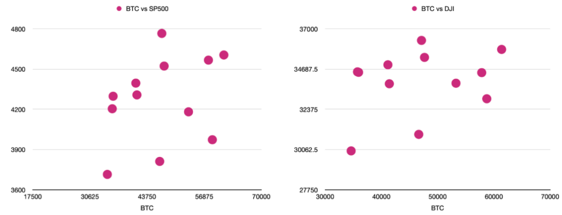


According to the historical data in 2021, we find that all objects have increasing trends through January to December. At certain points, for example in the month of September, both BitCoin and Stock indexes decrease correspondingly with each other at the same time.



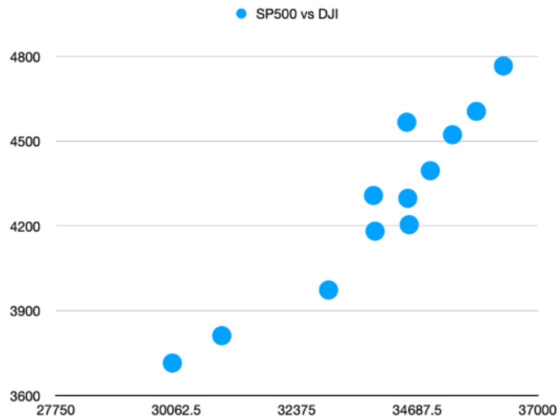
To model the linear regression of BitCoin in 2021, we use the equation in the linear regression model. The intercept and slope for the BitCoin price in 2021 can be described as $\hat{y}$ = 833.86168X + 41373.65742. The the θ_0 is a positive value, which indicates that as month increases the BitCoin price also increases.

The linear regression lines for stock indexes (Dow Jones and S&P 500) show a stronger relationship in comparing with BitCoin, and both of their RMSE are lower than BitCoin.



(BitCoin price and S&P 500, weak relationship) (BitCoin price and Dow Jones, weak relationship)

However, when Comparing with BitCoin with stock indexes, they all still show a relatively weak relationship. Although a higher BitCoin price (X-axis) is often associated with a high DJI or SP500 index value, there are many exceptions, for instance, the highest index value is not associated with the peak BitCoin value.



As a reference, the two stock market indexes are tightly connected and well-aligned in 2021, there is less RMSE in between SP500 and DJI according to the graph above.

Market Prediction

This section will focus on the AI Algorithm we use (LSTM) and the result of applying the model.

We use the LSTM (Long Short-term Memory) Algorithm and see if we can predict the future price of bitcoin. LSTM is a special RNN algorithm which is designed to overcome limitations of RNNs such as gradient vanishing and exploding, complex training, and difficulty to process very long sequences. Remembering information for long periods of time is intrinsic to LSTM. In this case, simply to say, we will use the bitcoin historical data to predict the future data.

There are seven steps to create this algorithm.

1. Data Generator

   To generate the data, we import the data by using pd.read_csv. Because we only need the price, therefore, we only use the data in the second column. We decided to use 80% of the total data to train our model and 20% of the total data used for testing to verify our result.

```
# Load the dataset
df = pd.read_csv('Bitcoin Historical Data.csv',usecols=[1])
plt.plot(df)
dataset = df.values
dataset = dataset.astype('float32')
scaler = MinMaxScaler(feature_range=(0,1))
dataset = scaler.fit_transform(dataset)
```

```
[3] #80% of the data is used for training
    train_size = int(len(dataset)*0.8)
    test_size = len(dataset)-train_size
    train = dataset[0:train_size, :]
    test = dataset[train_size:len(dataset),:]
    print('train_data: ', train.shape)
    print('test_data: ', test.shape)

    train_data:  (292, 1)
    test_data:  (73, 1)
```

2. Data Normalisation

   Normalisation is changing the values of numeric columns in the dataset so we use Scikit-Learn's MinMaxScaler with numbers between zero and one. Further, we have to convert the dataset into a form of 3D array to the LSTM model. First, we create data in 5 sequential steps and then convert the data into a 3D array with trainX, 5 sequential steps, and one feature at each step. The data shape will become: X_train: (286, 5, 1).

```
# Set up the time step as 5 days
def to_sequences(dataset, seq_size=1):
  x = []
  y = []

  for i in range(len(dataset) - seq_size-1):
    window = dataset[i: (i+seq_size),0]
    x.append(window)
    y.append(dataset[i+seq_size, 0])

  return np.array(x), np.array(y)

seq_size = 5
trainX, trainY = to_sequences(train, seq_size)
testX, testY = to_sequences(test, seq_size)


trainX = trainX.reshape(trainX.shape[0],trainX.shape[1], 1)
testX = testX.reshape(testX.shape[0], testX.shape[1], 1)
print("X_train: ", trainX.shape)
print("X_test: ", testX.shape)
print(trainX)
```

3. Defining the model

   Sequential for initiating the neural network, LSTM to add the LSTM layer. Also, we add one hidden layer in the neural network and add an output layer at the end. To compile our model, we use the Adam optimizer and set the loss as the mean_squarded_error.

```
model = Sequential()
# Adding a LSTM layer with 10 internal units
model.add(LSTM(64,input_shape=(None,1),activation='relu'))
# Adding a Dense layer with 1 units.
model.add(Dense(32))
model.add(Dense(1))
# Loss function + optimizer
model.compile(loss='mean_squared_error',optimizer='adam')
```

4. Training the model

   We use the model.fit to train our model for 100 epochs, the epochs are the number of times the learning algorithm will work through the training set.

```
model.fit(trainX, trainY, validation_data = (testX, testY), verbose = 2, epochs = 100)
```

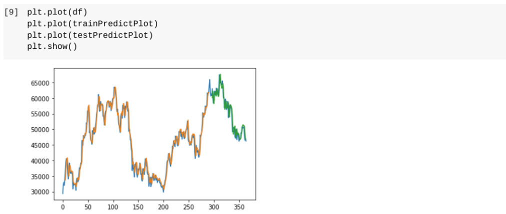
5. Making the Prediction
   After the model training , we start to make the prediction. We just need to use model.predict to get the training and test predicted data. The only thing we have to modify is to convert the 0 to 1 range of dataset to normal data by using scaler.inverse_transform.

```
trainPredict = model.predict(trainX)
testPredict = model.predict(testX)
print(testPredict.shape)

trainPredict = scaler.inverse_transform(trainPredict)
trainY = scaler.inverse_transform([trainY])
testPredict = scaler.inverse_transform(testPredict)
testY = scaler.inverse_transform([testY])
```

6. Evaluating the Prediction

   We plot the training prediction, test prediction data with the actual price data on the graph, and we can see that the prediction data is very close to the actual price data.The orange curve represents the training prediction and the green curve represents the test prediction. So we can conclude that the LSTM can be somewhat effectively to predict the Bitcoin future price.



Conclusion

Through this AI project, we are attempting to apply the AI algorithms we have learned so far to make predictions in the industry. At some point, we find that BitCoin price is somehow correlated with stock indexes (especially with Nasdaq) but still not as close as indexes themselves. In a monthly scope, we would expect that a decrease of stock indexes may cause the drop of BitCoin price or vice versa.

In the project we were also trying to use Genetic Algorithm to find the highest and lowest price with its corresponding month. However, as the polynomial function we calculated is a descriptive function and can only be used to describe the price in 2021, we were not able to get a fitness result for future prediction. In the future research, we can use real data along with Keras LSTM model to train it and may get a better result.

Another defects for the project is that, both Regression analysis and LSTM modelling use too little data, but they (both CryptoCurrency and Stock Indexes) are also changing macroscopically, whether by the regional government, the global news, or the pandemic. The data we used for LSTM prediction is an array with size of 365 floats. Among them, 255 were used for training and 110 were used for testing, and only 10 days were actually used to predict the 11th day’s price, which is also not enough for prediction (the examples we have seen online uses 60 days to train to get the 61st day’s price).

References

Luke Sun (April. 7 2020), LSTM for Stock Price: Prediction: https://towardsdatascience.com/lstm-for-google-stock-price-prediction-e35f5cc84165

Avijeet Biswal (2022, Feb. 16), An Easy Guide to Stock Price Prediction Using Machine Learning: https://www.simplilearn.com/tutorials/machine-learning-tutorial/stock-price-prediction-using-machine-learning#stock_price_prediction

Wikipedia, BitCoin: https://en.wikipedia.org/wiki/Bitcoin