Predicting Bitcoin Market Movements Using Long Short-Term Memory

Author: Thomas Paul, BSc

Knowledge Discovery and Data Mining 2

Introduction

The main goal of this project is to try and predict Bitcoin market movements based on small patterns during the day and the week using a long short-term memory neural network.

Dataset

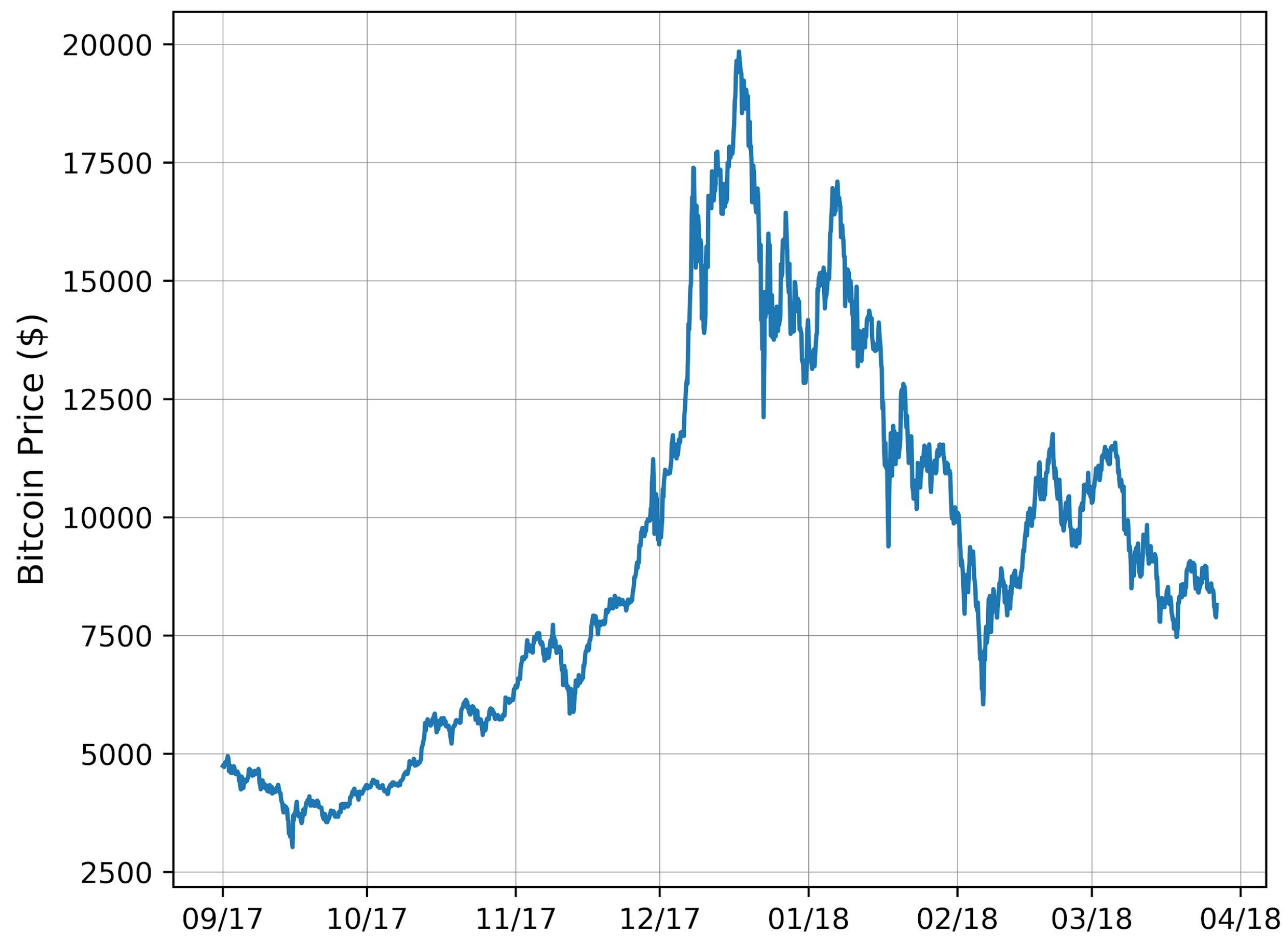
First I wanted to crawl the data myself but there already are considerable records online. I chose a dataset from Kaggle[3] that has

- Timestamp
- Open
- High
- Low
- Close
- Volume_BTC
- Volume_Currency
- Weighted_Price

data for every minute from january the first in 2014 until march the 27. in 2018.

Filtering

Since I wanted to look at recurring price trends on a hourly basis while considering the day of the week, I chose to only use the dates of full hours and drop the other data. I also believe that late 2017 brought many new players to the bitcoin market, therefore I decided to look only at data after the first of september 2017.

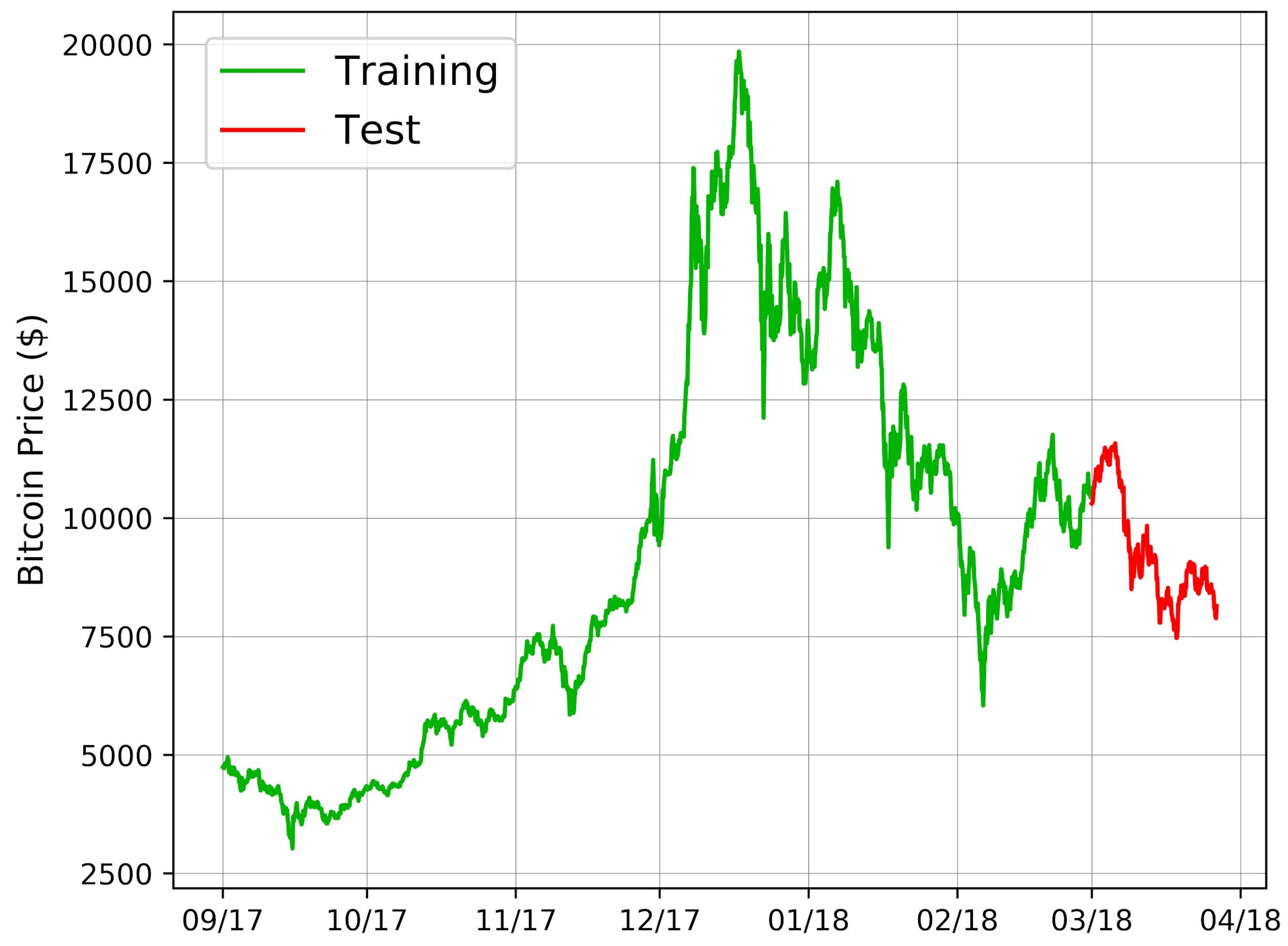


A plot of the dataset after filtering all the unused dates

For my LSTM network I chose to work with the *open* price and the *volume* traded in bitcoin. I also added columns with numbers representing the hour of the day, the day of the week and removed the date since there is no information to be learned from the date.

Training and Test Set

As training set I used the first 6 months in the dataset and the last month I used as test set. I split the data at the 01.03.2018.



Dataset split into training and test set

Preprocessing

For the neural network to work best some of the columns have to be normalized. The columns for *open* and *volume* had to be normalized so their values would not differ too heavy. I left the columns for hours and weekdays with their normal values since those were basically constants.

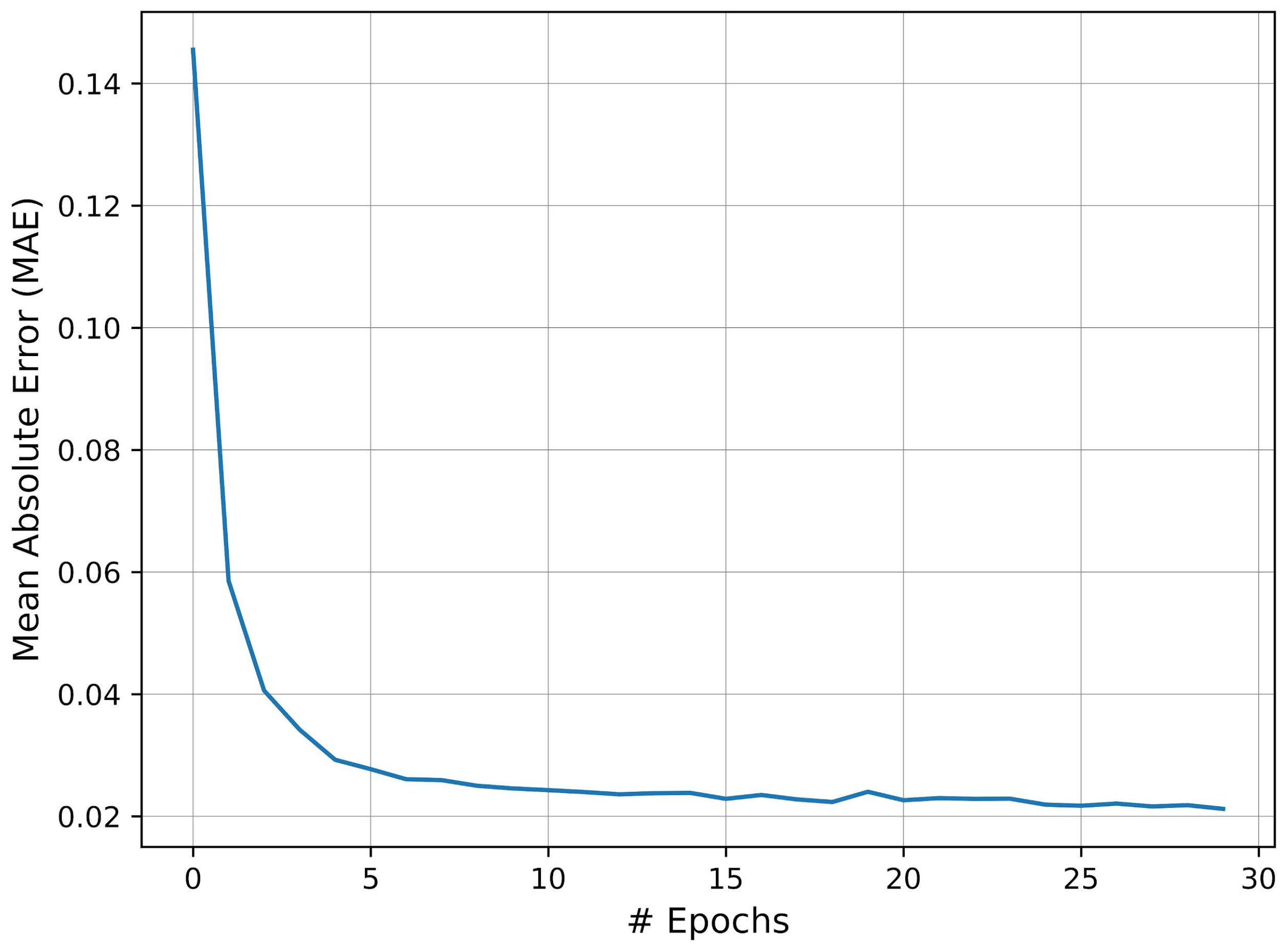
Next I split the data into chunks for training the model for a specific time frame. At this point I used every third hour and a chunk size of 12 dates wich equals a time frame of 36 hours.

Building the Model

For my Model i used following parameters:

- model = Sequential()
- neurons = 25
- dropout=0.25
- activ_func="linear"
- loss="mae"
- optimizer="adam"

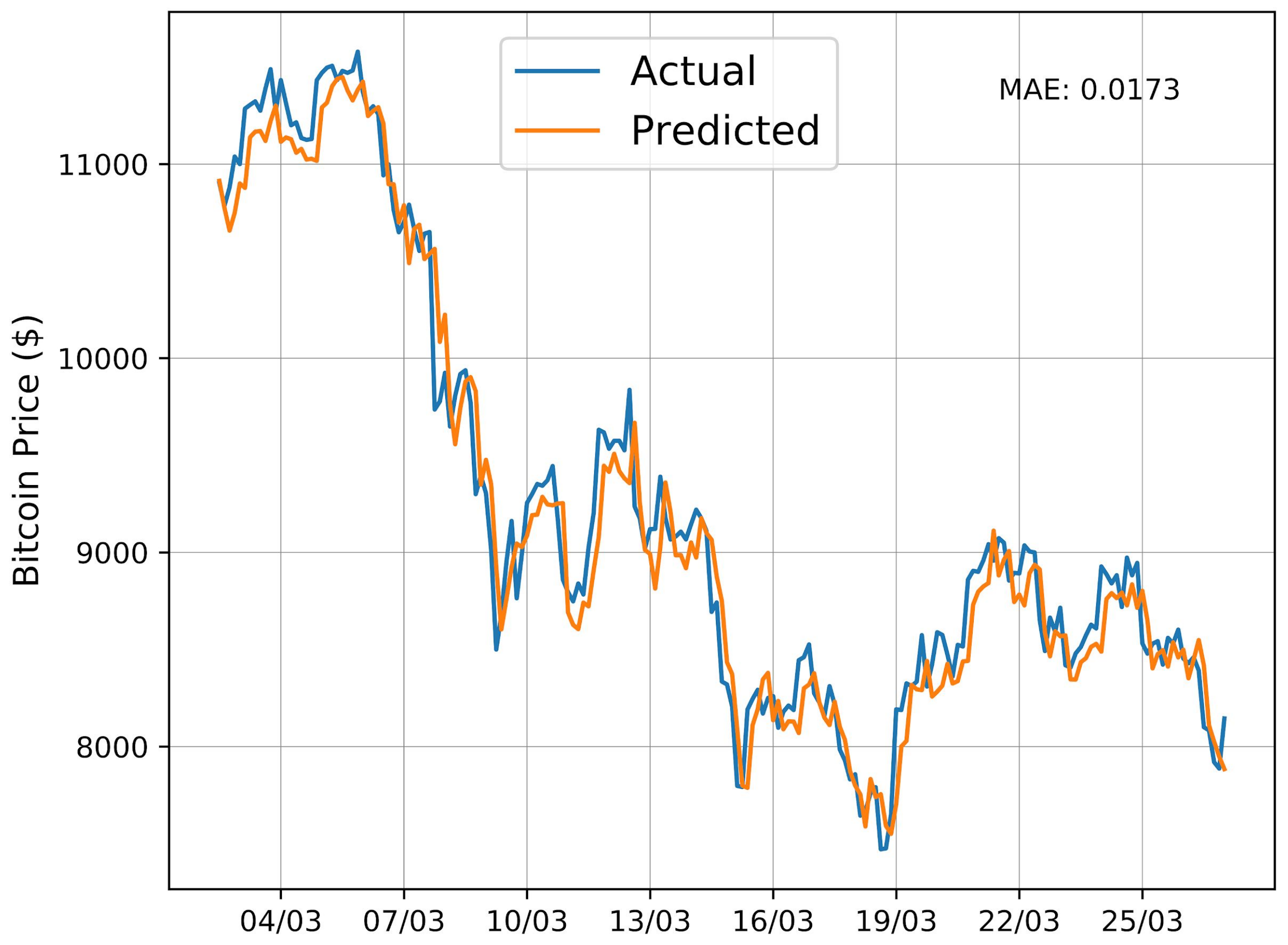
After building the model I trained it for 30 epochs until the mean absolute error was not getting any lower.



Mean Absolute Error while training the model

Testing the Model

The last step was testing the model with the test data set the model has never seen before, to see if it could be of any use for predicting the future price of bitcoins.



Test Set: Single Timepoint Prediction

Conclusion

On the first look the prediction looks very promising, but it has some downfalls. Even though there are some parts it manages to predict in a moderate manner, there are parts where it overshoots the actual price and many of the good predictions have a simple lag characteristic.

Apart from these errors the model does really well considering the basic data it was fed, it definitely shows the awesome potential machine learning has.

Literature / Cites

[1] https://www.kaggle.com/mczielinski/bitcoin-historical-data/data