Final Project - Bitcoin Price Prediction

Katerina Tsilingiri
2806
tsaikaterini@uth.gr

Chrysa Noli
2780
nochrysoula@uth.gr

Panos Petropoulos
2610
papetropoulos@uth.gr

I. Introduction

In the final project of our course Machine Learning, we had to select an application area of Time Series Forecasting from those that our professor has provided. We have chosen the cryptocurrency field and more specifically we have decided to try to predict the price of the well known cryptocurrency, Bitcoin for the year of 2022. Some of the models/methods we have used are LSTM Neural Networks, XGBoost, SVM, Random Forest and AutoML (AutoTS).

II. Exploratory Data Analysis (EDA)

The dataset that we have selected has information about the daily price (in EUR) of Bitcoin (BTC) from 21 November 2017 until 15 June 2022 and has the following features:

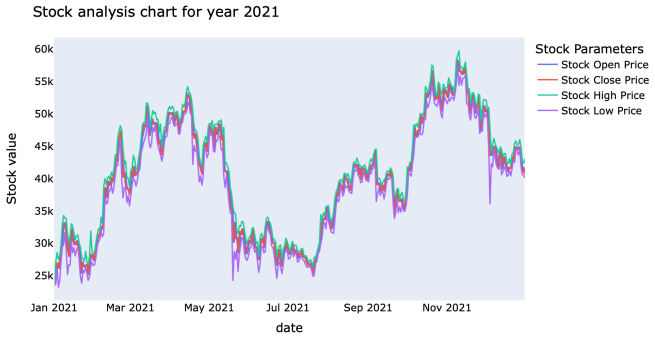
- date: with format YYYY/MM/DD 00:00:00
- symbol: BTC/EUR
- open: bitcoin price at the start of the day
- high: the highest price reached throughout this day
- low: the lowest price reached throughout this day
- close: bitcoin price at the end of the day
- Volume EUR: indicates how many Bitcoins are being bought and sold on specific exchanges in euros
- Volume BTC: indicates how many Bitcoins are being bought and sold on specific exchanges

Our dataset consists of 1668 entries with start date 2017-11-21 and end date 2022-06-15. Also, it does not include missing or null values.

Before modelling our dataset, we have decided to make a more detailed analysis on the given data. In this way, studying the behavior of the price during the years of 2018, 2019, 2020, 2021 and the first half of 2022, we will be able to understand some interesting relationships and patterns between them.

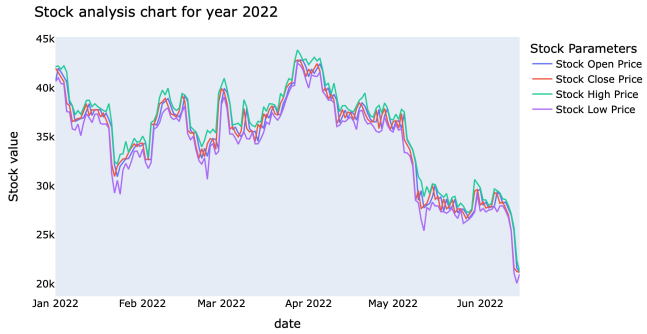
To do this, we drop the time from the column 'date' as the time in all days is the same (00:00:00) and it does not give us furthermore information about what we want to succeed. In addition, we "split" each year in months for better understanding.

We exclude year 2018 from the whole dataset with the .loc function of pandas library and we keep only the features: date, open, high, low, close. Firstly, we make a monthwise comparison between Stock open and close BTC price in year 2018 and the we continue to a monthwise High and Low stock BTC price comparison. Then, we compute and plot the stock analysis chart for year 2018, which includes all the information about open, high, low, close BTC prices per month, as you can see below. We follow the same process for the years 2019, 2020, 2021 and 2022.

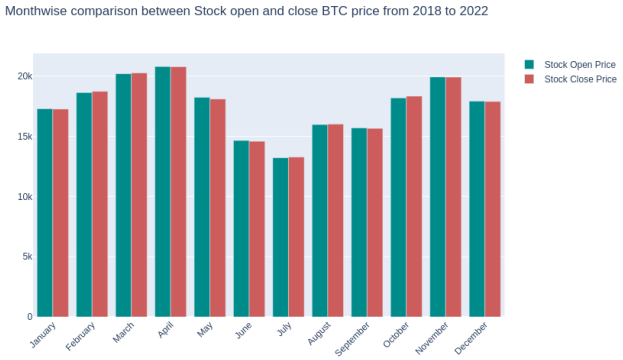


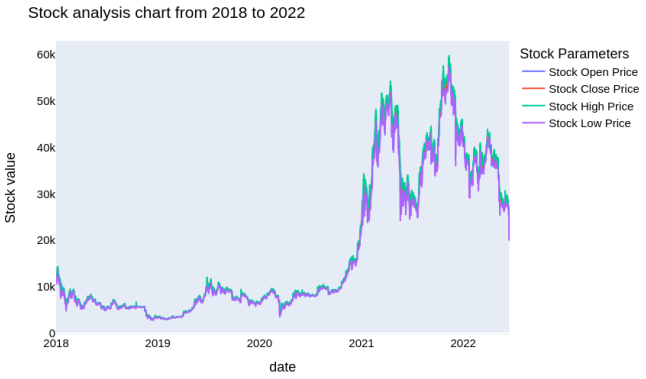
A. 2022

The year of 2022 is a special case, as we have information only until 15/06. For this reason, the rest of the months are NaN (missing values). So, we continue the process described in year 2018 with these and we plot the Stock analysis chart for year 2022.



B. 2018-2022 (overall analysis)





III. Pre-Process

Before applying the various models on our data, we prepare our data for the training part. Using the function add_datepart from the library fastai, we separate the 'date' column from our dataset into 'Day', 'Dayofweek', 'Dayofyear', 'Is_month_end', "Is_month_start', 'Is_quarter_end', 'Is_quarter_start', 'Is_year_end' and 'Is_year_start'. We have decided to do that, because in some models, we do not need the 'date' column (and more specifically its format), but at the same time, we did need the information of the date. After changing the format of the date, we have separated the "y" of our dataset (which is the 'close' column) from the rest of the data and we have scaled them (using the function MinMaxScaler()). Since some models we will be working with are Neural Networks, if we have skipped this step, we would have a very large loss in the training and the testing. Then, we have splitted our data set into train and test part (setting the test size equal to 0.1 and the shuffle option disabled to keep the correct order of the information). Also, since the main goal of this project is to predict the price of Bitcoin during the year of 2022, the train dataset is defined from 21 November 2017 (21/11/2017) to 31 December 2021 (31/12/2021) and the test dataset is defined from 01 January 2022 (01/01/2022) to 15 June 2022 (15/06/2022).

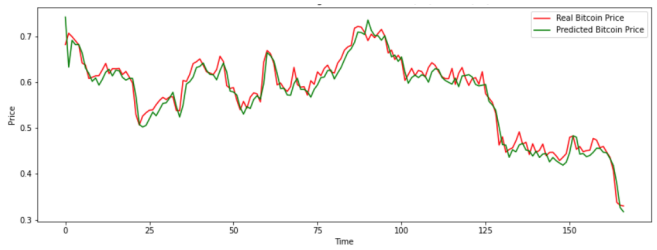
IV. LSTM Neural Network

For the first model of our analysis, we have chosen the LSTM Neural Network. Long short-term memory NN helps us predict the price of BTC for the next 6 months. We have

defined our model using 10 LSTM layers with activation function the reLU with only one Dense layer. Since it has a single hidden layer in the layer it is a Vanilla LSTM. After fitting the model for 200 epochs with batch size set to 32, we can notice in the output of the jupyter notebook the following results that measure the accuracy of our model.

Evaluation of LSTM	
Loss	0.00041342
R^2 score of train data	0.99562
R^2 score of test data	0.9497

Fig. 1. Bitcoin price prediction using LSTM from 01/01/2022 to 15/06/2022



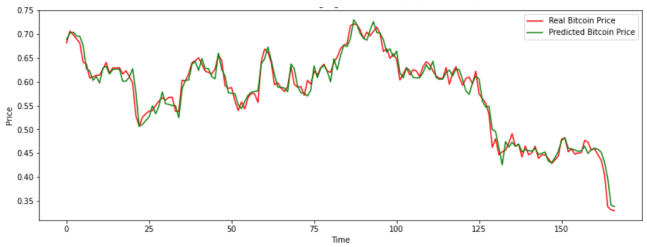
As we can see from the plot, the lines are almost identical. From our dataset, we have the real price of Bitcoin until 15/06/2022 (which is shown with the red line). Our prediction is shown with the green line. Also, the numbers in the the x-axis (Time), represent the days (which are equal to 5,5 months, approximately 165).

V. XGBOOST

For our second model, we have used the XGBoost. It is a gradient boosting tree algorithm. More specifically, the trees are built in parallel (hence it is very fast)and following a level-wise stategy predicts the desired value with a very high accuracy. For the purpose of our prediction, we have used 1000 estimators and we have followed the same procedure as in LSTM, meaning that we fitted our model and then we predicted the train and test sets. Also, from the output of the notebook, we have notice that the learning rate that the model uses is equal to 0.30.

Evaluation of XGBoost	
R^2 score of train data	0.99998
R^2 score of test data	0.9686

Fig. 2. Bitcoin price prediction using XGBoost from 01/01/2022 to 15/06/2022



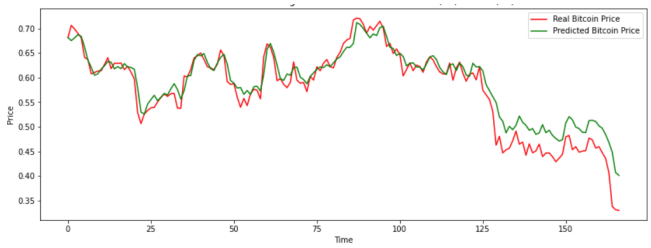
Observing the plot, in this case, also the lines are very similar. Our predictions using the XGBoost model match the real price of Bitcoin. Again, the x-axis represents the number of days that we have predicted the price.

VI. SVM

The third chosen model is SVM. For this model, we have tried two different kernel, since we have noticed some differences in the accuracy. In both kernels, we have set the regularization parameter to 1000. Firstly, we have chosen the linear kernel, which is the simplest. After predicting the test and training sets :

Evaluation of SVM (linear)	
R^2 score of train data	0.9361
R^2 score of test data	0.8903

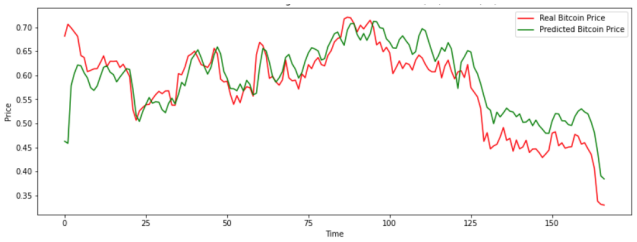
Fig. 3. Bitcoin price prediction using SVM with linear kernel from 01/01/2022 to 15/06/2022



Using the linear kernel of the SVM model, the lines in the beginning of the plot are very similar. However, in the end we can notice that the real price of the Bitcoin is lower than the one we have predicted. This does not cause a big problem, because in both cases there is a decrease in the price. Therefore, our model still predicted correctly, even though there is a small difference in the final price.
Then, we have used the Radial Basis Function (RBF) kernel, in which the results as anyone can observe below, are not very good regarding the test data.

Evaluation of SVM (RBF)	
R^2 score of train data	0.9767
R^2 score of test data	0.6704

Fig. 4. Bitcoin price prediction using SVM with RBF kernel from 01/01/2022 to 15/06/2022



Using the RBF kernel of the SVM model, we observe that the lines in the middle are almost the same regarding the

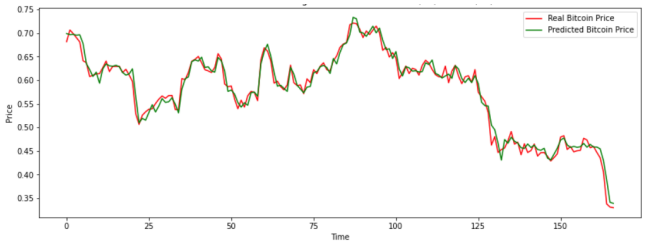
slope and the values. In the beginning and in the end, there is a small deviation in the value but, the positive thing is that the slope is the same, apart from a few days in the beginning. More specifically the price of Bitcoin (according to the real data) has a small increase and then a big fall in the price, while our predicted values represent that there is a greater increase and then the decrease. Because this "false" prediction is in the beginning, we can justify it by assuming that the model wanted some more iterations to finally predict correctly the data. Also, this can be verified by the R^2 value from the table above, which is low (compared to other models we have used) in the test data.

VII. Random Forest

The fourth model we have tried to predict the price of Bitcoin is Random Forest. We have used the RandomForestRegressor from the sklearn library. After fitting and predicting the train and test sets we have accomplished a very high accuracy as seen below:

Evaluation of Random Forest	
R^2 score of train data	0.9998
R^2 score of test data	0.9767

Fig. 5. Bitcoin price prediction using Random Forest from 01/01/2022 to 15/06/2022

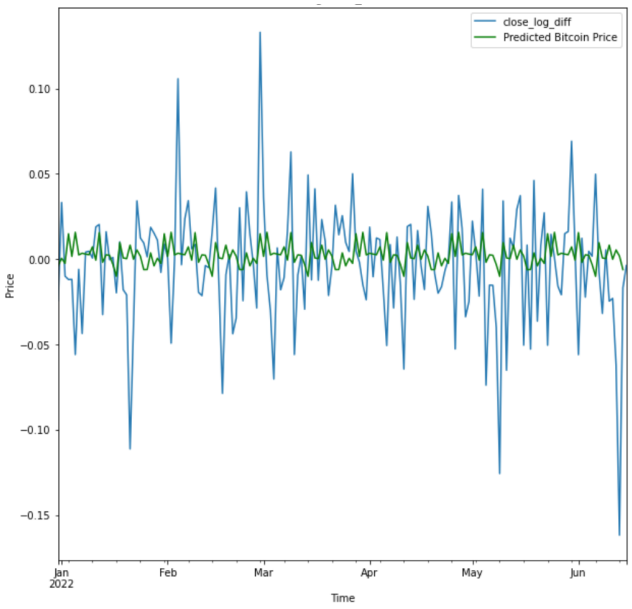


In the above graph, we can easily conclude that, this is the best so far. After all, the R^2 score of the test data is very high.

VIII. AutoML - AutoTimeSeries (Auto-TS)

Lastly, we have used AutoML and more specifically AutoTS (AutoTimeSeries) to predict the future price of Bitcoin. In this method, the AutoTS given the train data, tries to find by itself the best model to make the most accurate predictions. In order to define our model, we have set the forecast length equal to 165 days (until 15/06/2022) since we do not predict the test set, like we did in the previous models.
After fitting the model, we get that the best model for our data is the Seasonal Naïve. This is very similar to Naïve. The only difference is that in order to predict a new value, uses the the value of the same period a season ago (instead of the most recent one that does the Naive). In our case, when the model tried to predict the price of Bitcoin in February 2022, used the price in February 2021 and so on. From the following plot, unfortunately the predicted price of Bitcoin is not exactly the same as real practice. But we can observe that when the real price is decreased, our predicted price is also is decreased. The same thing happens when the price increases.

Fig. 6. Bitcoin price prediction using AutoTS (Seasonal Naïve) from 01/01/2022 to 15/06/2022



IX. Conclusion

In conclusion, we believe that our analysis and prediction of the price of Bitcoin for the first half of 2022 was very successful. There are some exceptions, like in the method of SVM (both kernels) and Seasonal Naïve, where the values were not the same. However, since the cryptocurrency field is not very stable and it is very hard (if not impossible!) to predict with 100% accuracy the price, we are very satisfied with the overall results.

References

[1] Source of Dataset (Cryptodatadownload)
https://www.cryptodatadownload.com/data/
[2] WHAT IS XGBOOST?
https://www.nvidia.com/en-us/glossary/data-science/xgboost
[3] XGBoost versus Random Forest
https://medium.com/geekculture/xgboost-versus-random-forest-898e42870f30
[4] Simple forecasting methods
https://openforecast.org/adam/simpleForecastingMethods.html
[5] Intro AutoTS
https://winedarksea.github.io/AutoTS/build/html/source/intro.html
[6] autots package
https://winedarksea.github.io/AutoTS/build/html/source/autots.html
[7] Lecture Notes from our course Machine Learning