Symbiosis Skills and Professional University (SSPU)

A PROJECT REPORT ON

Bitcoin Price Forecasting using Machine Learning Algorithms

Submitted by:

Name: Sayali Bhosale

CLASS: ML and AI Analyst **Batch: 8**

Under the Guidance of

Prof. Pradeep Khare

Bitcoin Price Forecasting using Machine Learning Algorithms

I. Abstract

Bitcoin is surging nowadays due to an increase in investor enthusiasm and technology that allow more novice investors to begin trading, while also being a hedge against the dollar due to the global affairs. In this project, we investigate further the relationship between these data points and try to fit a model to predict the price of Bitcoin. This work presents all studies, methodology, and results about Bitcoin forecasting with PROPHET and Time Series models. To find the most accurate forecast model, the performance metrics of PROPHET and ARIMA methods are compared on the same dataset. The dataset selected for this study starts from May 2016 and ends in May 2021, which is the interval that Bitcoin values changing significantly against the other currencies. Data is prepared for time series analysis by performing data pre-processing steps along with feature selection. Although the time series analysis has a univariate characteristic, it is aimed to include some additional variables to each model to improve the forecasting accuracy. Those additional variables are selected based on different correlation studies between cryptocurrencies and USD. Finally, three different models are created and compared in terms of performance metrics. Based on the extensive testing we see that Facebook PROPHET outperforms Vector Auto Regressive model (VAR) and XGBoost model by 0.93 to 0.94 in R2 values.

Keywords: Cryptocurrencies, ARIMA, Facebook Prophet, VAR, XGBoost

II. Introduction

Bitcoin is a cryptographic money which is utilized worldwide for advanced instalment or basically for speculation purposes. Bitcoin is decentralized for example it isn't possessed by anybody. Today Bitcoin is a secure transaction system that has a valuable impact on capital. They are awarded under a restriction in which customers offer their computer authority to register and listing trades with the bitcoins.

Exchanges made by Bitcoins are simple as they are not attached to any nation. Speculation should be possible through different commercial centres known as "bitcoin trades". These enable individuals to sell/purchase Bitcoins utilizing various monetary forms. The biggest Bitcoin trade is Mt Gox. Bitcoins are put away in an advanced wallet which is essentially similar to a virtual financial balance. The record of the considerable number of exchanges, the timestamp information is put away in a spot called Block chain. Each record in a block chain is known as a square. Each square contains a pointer to a past square of information. The information on block chain is scrambled. During exchanges the client's name isn't uncovered, however just their wallet ID is made open. The Bitcoin's worth fluctuates simply like a stock though in an unexpected way. There are various calculations utilized on financial exchange information for value forecast. Notwithstanding, the parameters influencing Bitcoin are extraordinary. In this manner it is important to anticipate the estimation of Bitcoin so right venture choices can be made. The cost of Bitcoin doesn't rely upon the business occasions or

mediating government not at all like securities exchange. Hence, to anticipate the worth we feel it is important to use AI innovation and ML techniques to foresee the cost of Bitcoin.

III. Proposed Architecture / Methodology:

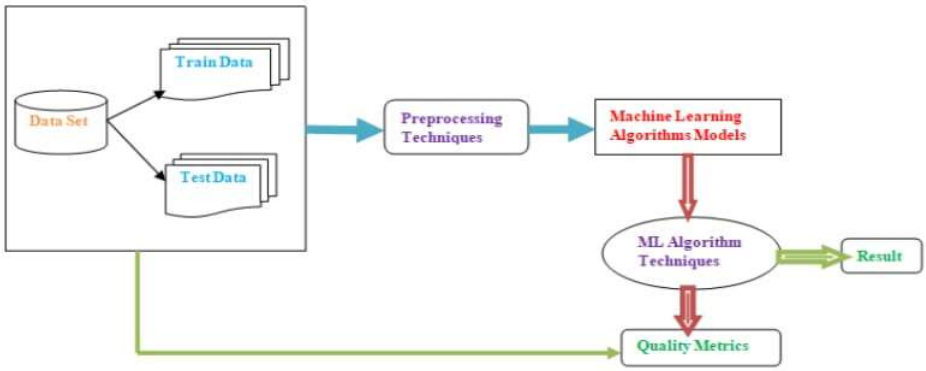
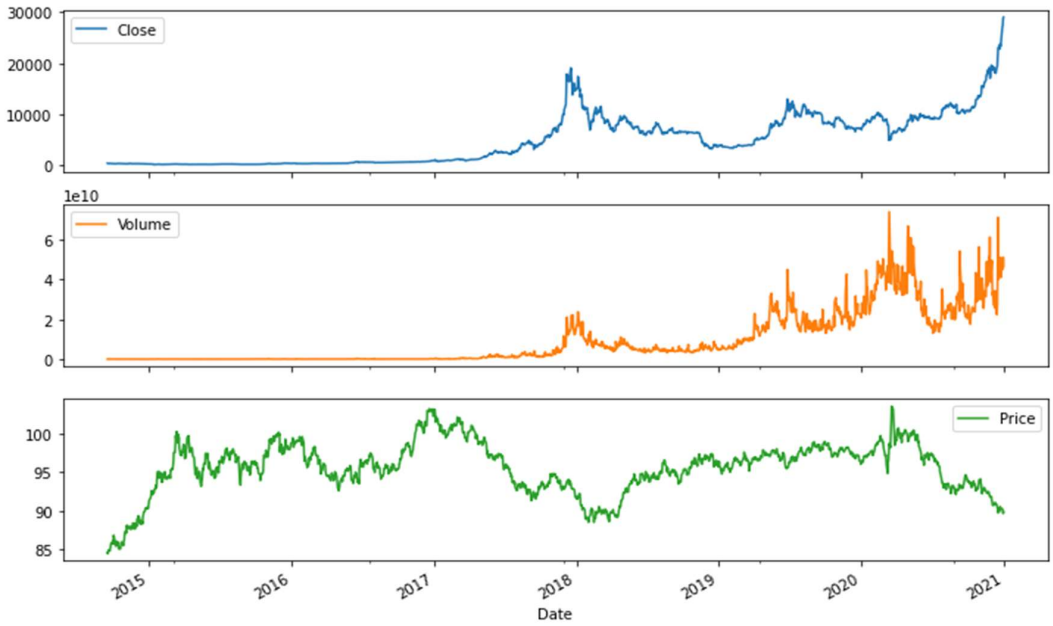


Fig.1.Bitcoin price prediction

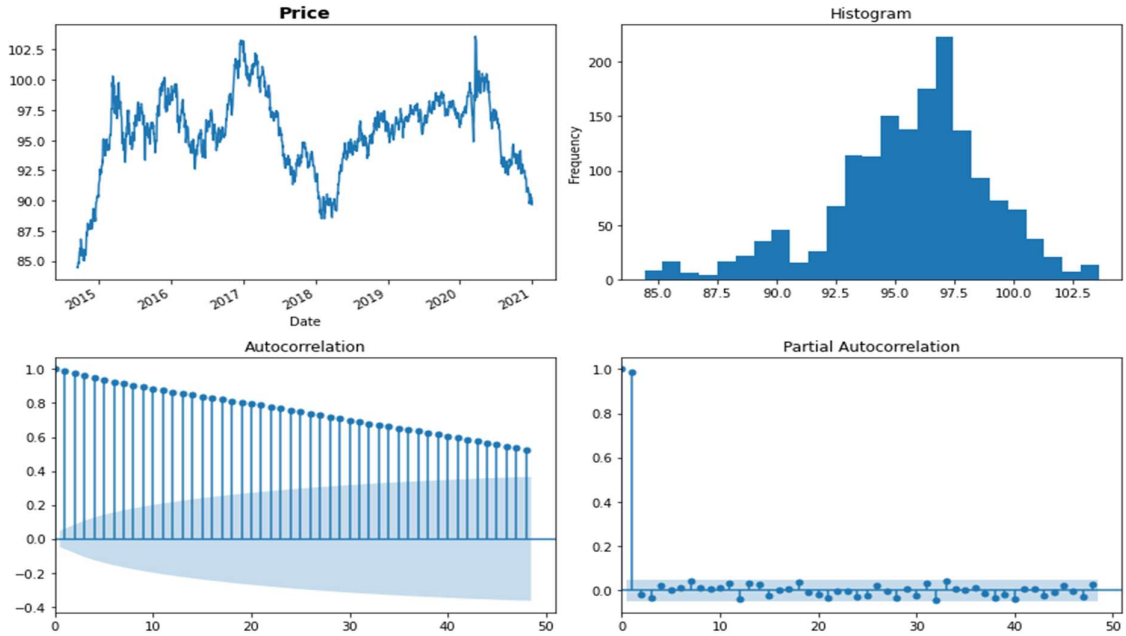
IV. Data Collection and Pre-processing:

The data for our model is collected by very popular Kaggle data science community. It is a historical data for over past 5 years from 23rd May 2016 to 23rd May 2021. We have also used the USD Price data for the same timeline.

In data preparation phase, we assess the condition of data and figure out the trends, outliers, inconsistencies present in the data.



By comparing the prices of Bitcoin and USD visually, we can see a peak in bitcoin around 2017-2018 and a decline in the USD around the same time. There is a similar pattern toward the end of 2020 going into 2021.



Here we plot four essential graphs that provide us insight into the closing price of bitcoin over this series of time. The bottom two graphs are the graphs that allow you to look at the dependency structure. They're called the autocorrelation and partial autocorrection function graphs. Each bar on in the autocorrelation function graph captures the series itself and how it is correlated with its own past. These sorts of graphs are very useful for identifying the order of our model.

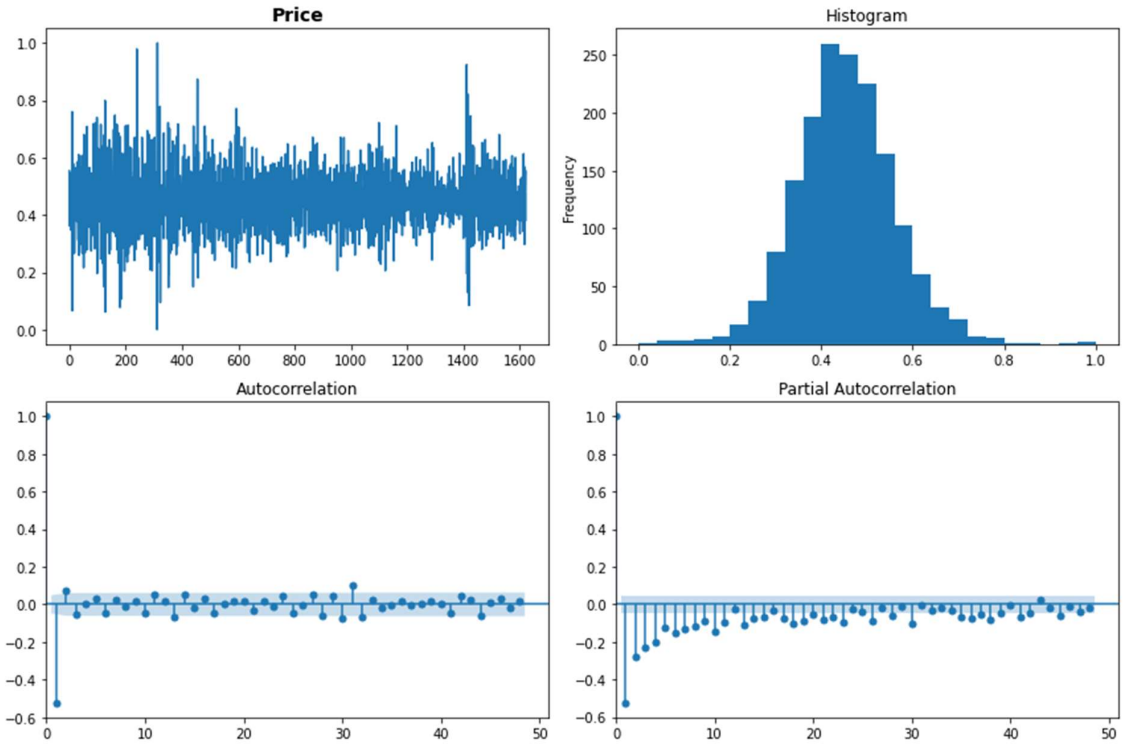
a. Normalization

Normalized data within statistics, often involves eliminating units of measurement from a dataset. As a result, this enables us to easily compare data with different scales and are measured from different sources.

When training a machine learning model, we are aiming to bring the data to a common scale and so the various features become less sensitive to each other. In this case, we utilize data normalization as a method of transforming our data, which may have different units or scales (Bitcoin and USD). This allows our model to train with features that could lead to more accurate predictions.

b. Transformation

Applying differencing or seasonal differencing log of the series would make the series stationary. Simply putting, stationarity removes the trends from the dataset which can be extremely intrusive to models. So, stationarity makes our models perform and predict better.



Simply put, stationarity removes trends from the dataset which can be extremely intrusive to our models. Basically, stationarity makes our models perform and predict better.

Here we are running a statistical test to determine how well the times series was transformed to be stationary. This is a test that outputs certain statistical patterns that we can use to judge whether each parameter is stationary.

By doing all such necessary processes, data is now ready to feed into our proposed models.

V. Implementation

Based on our study we demonstrated that LSTM and SVR models can also be used but while we were performing the forecasting model with our data those models fail to give us satisfactory outputs. So, we decided move forward with following three models which gives high accuracy along with spectacular visualizations.

Three methods for forecasting time series data.

1. Vector Autoregressive (VAR) Model
2. XGBoost Model
3. Facebook Prophet

Train-Test Split Data

Whole data get split into the sample for training and validation sets. In time series, selection of training and testing set is not done randomly, because of the time dependence. Our dataset is provided with daily intervals and so we can forecast up to a particular number of days.

a. Vector Autoregressive (VAR) Model

A VAR model is a generalisation of the univariate autoregressive model for forecasting a vector of time series. It comprises one equation per variable in the system. The right-hand side of each equation includes a constant and lags of all of the variables in the system. To keep it simple, we will consider a two variable VAR with one lag. If the series are stationary, we forecast them by fitting a VAR to the data directly (known as a "VAR in levels"). If the series are non-stationary, we take differences of the data in order to make them stationary, then fit a VAR model (known as a "VAR in differences"). In both cases, the models are estimated equation by equation using the principle of least squares. VAR models are a specific case of more general VARMA models. VARMA models for multivariate time series include the VAR structure above along with moving average terms for each variable. More generally yet, these are special cases of ARMAX models that allow for the addition of other predictors that are outside the multivariate set of principal interest.

b. XGBoost Model

Usually, ARIMA regressions are used in classical statistical approaches, when the goals not just prediction, but also understanding on how different explanatory variables relate with the dependent variable and with each other. ARIMA are thought specifically for time series data.

On the contrary, XGBoost models are used in pure Machine Learning approaches, where we exclusively care about quality of prediction. XGBoost regressors can be used for time series forecast, even though they are not specifically meant for long term forecasts. But they can work.

Since XGBoost is not specific for time series data in order to build more robust models, it is common to do a k-fold cross validation where all the entries in the original training dataset are used for both training as well as validation.

c. Facebook Prophet

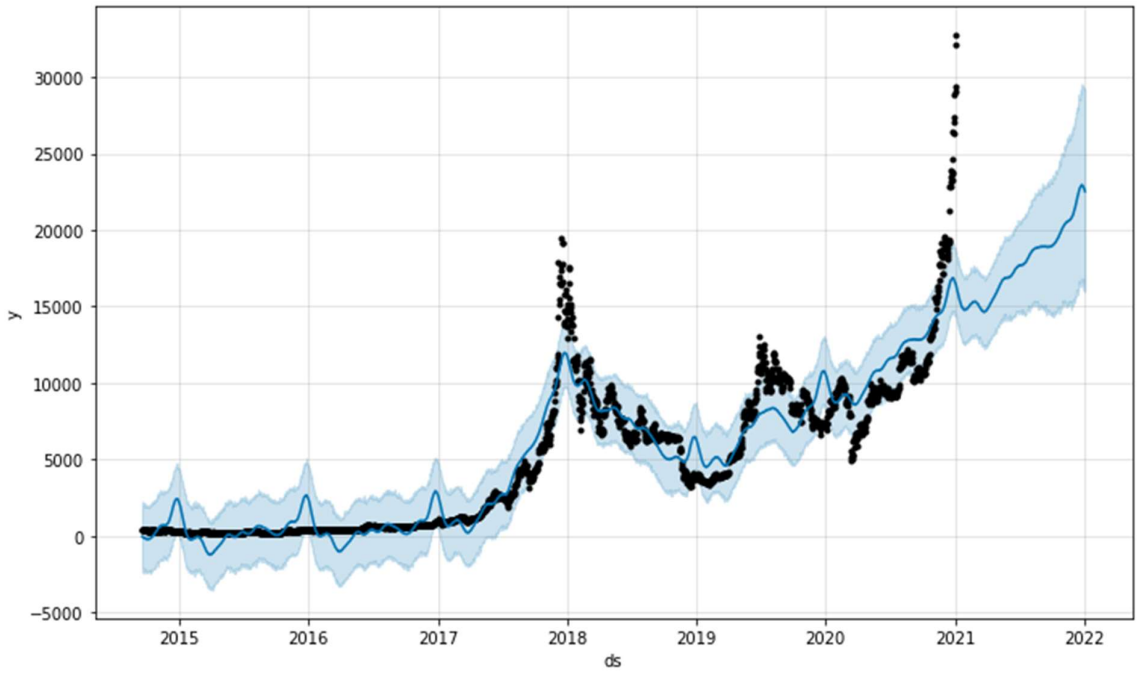
PROPHET is published by Facebook's Core Data Science team. Facebook prophet was created to work as a tool for most general time series predictions. Facebook prophet is able to visualize significant features in the time series such as trends, outliers, seasonality, etc. Also, the forecasting method is robust enough to handle any missing values.

It depends on a contribution model where non-linear trends are fit with weekly and yearly seasonality and plus holidays. In addition, it gets a reasonable estimate of the mixed data without spending manual effort. Purely automatic prediction techniques are not flexible to combine useful assumptions because they are fragile. Furthermore, high quality estimates are not easy to make, requiring special data science skills.

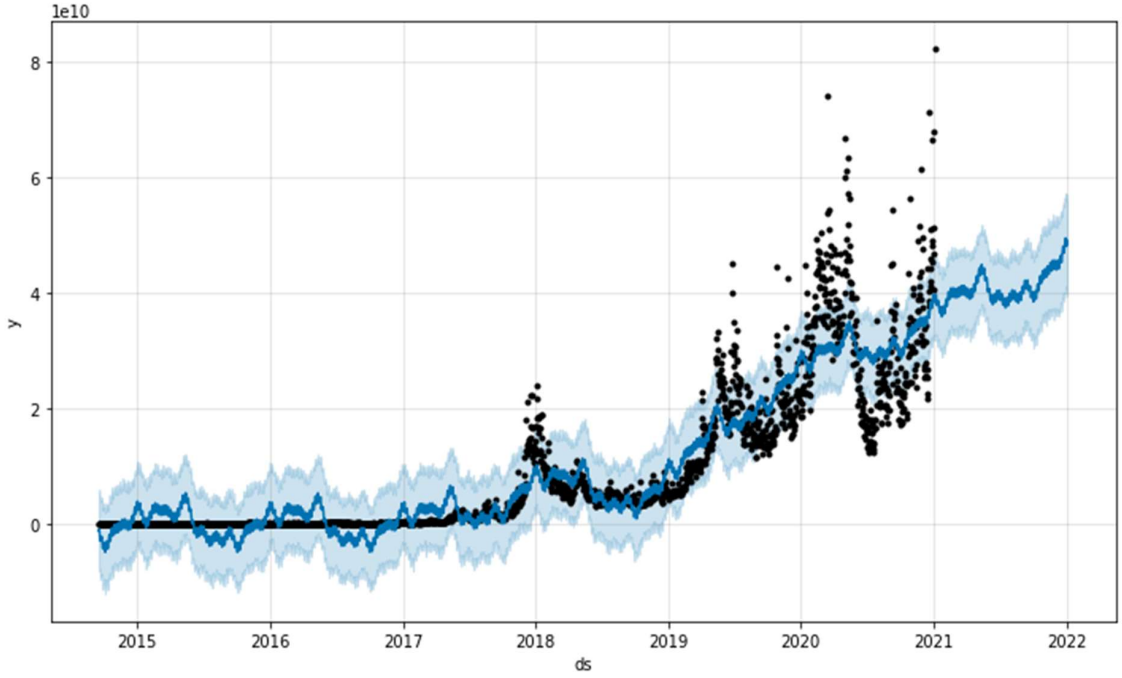
PROPHET framework has its own special data frame to handle time series and seasonality easily. The data frame needs two basic columns. One of these columns is" ds" and this column stores date time series. The other column is" y" and it stores the corresponding values of the time series in the data frame. Thus, the framework can work on seasonal time series quiet well and it provides some options to handle seasonality of the dataset. These options are yearly, weekly, and daily seasonality. Due to providing these options, a data analyst can choose the available time granularity for the forecast model on the dataset.

VII. Insights

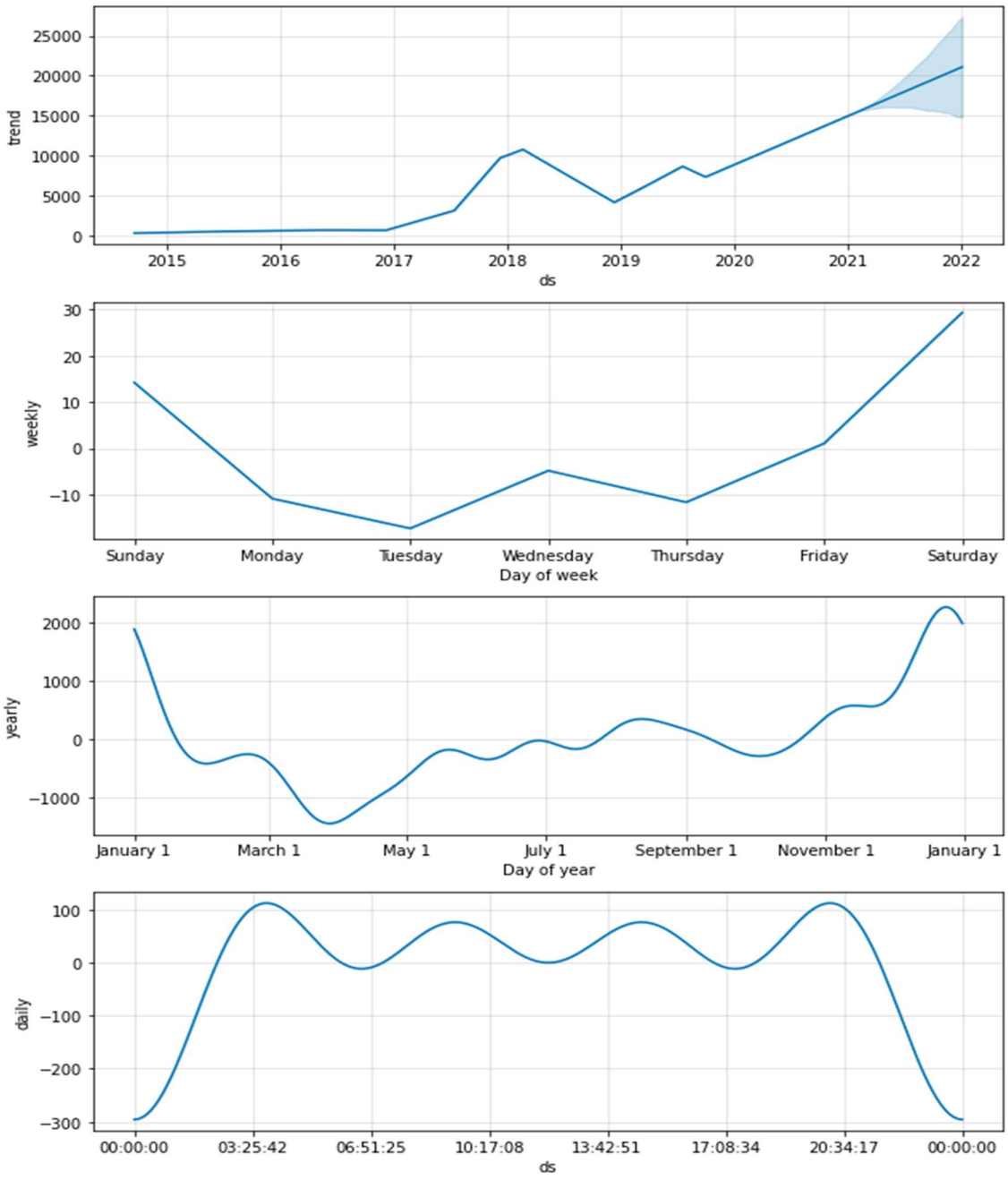
- The forecast is expecting bitcoin to continue rising in value. There has a been a new spike at the end of 2020 that added to this prediction.



- The forecast is predicting the trend to continue to rise. There is growing demand and as more individuals tend to buy more bitcoins and is in circulation throughout the economy.



- These plots shows the trends by weekly, yearly and daily manner so that it's easy to predict the behaviour of Bitcoin price along with certain time intervals.



VIII. Result and Evaluation

- VAR Model:
  Using VAR model, we get the value 26.8791 as Mean Squared Error (MSE). It is simply the average of the square of the difference between the original values and the predicted values. There are no acceptable limits for MSE except that the lower the MSE the higher the accuracy of prediction as there would be excellent match between the actual and predicted data set.

```
In [20]: from sklearn.metrics import mean_squared_error
         from numpy import asarray as arr
         mse = mean_squared_error(test, forecast_values)
         print("\nMean Squared Error: ", mse)

         Mean Squared Error:  26.87915225460752
```

- XGBoost Model
  These models are used in pure Machine Learning approaches, where we exclusively care about quality of prediction. XGBoost regressors can be used for time series forecast, even though they are not specifically meant for long term forecasts. But they can work. Using XGBoost model on our dataset we get the accuracy of 78.79%

```
[ ] from sklearn.model_selection import GridSearchCV

    clf = GridSearchCV(xg_reg, {'max_depth': [2,4,6],'n_estimators': [50, 100, 200]}, verbose=1,n_jobs=2)
    clf.fit(X, y)

    print(clf.best_score_)
    print(clf.best_params_)

    Fitting 5 folds for each of 9 candidates, totalling 45 fits
    [Parallel(n_jobs=2)]: Using backend LokyBackend with 2 concurrent workers.
    [17:01:40] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor of reg:squarederror.
    0.7879727674240937
    {'max_depth': 2, 'n_estimators': 200}
    [Parallel(n_jobs=2)]: Done  45 out of  45 | elapsed:    3.1s finished
```

- Facebook Prophet
  We have used the FBProphet model on two different datasets to predict the Bitcoin Price. For the first dataset we got the 93.17% accuracy while for second dataset we got the accuracy about 94.24%.

```
metric_usd.dropna(inplace=True)

[ ] r2_score(metric_usd.y, metric_usd.yhat)

    0.9424398708981777

[ ] mean_squared_error(metric_bc.y, metric_bc.yhat)

    10464949.380877443
```

IX. Conclusion

Prophet easily allows us to quick view forecasts for an individual series. Here we were able to do so for the Bitcoin closing price, the USD price, and the column of Bitcoin. These variables are important because, the surge in bitcoin due to the ease of individuals being able to trade via robin hood and other apps, as well as a hedge against the US Dollar because of the current state of global affairs.

According to the forecasts for each series, at least for the near future we are going to continue to see bitcoin to rise in value. This could be attributed to the forecast that the US Dollar to decrease and the forecast that the amount of bitcoin is circulation is also going to increase.
The initial two models (VAR & XGBoost) were able to determine at least one of these insights, but not as easily or quickly as prophet.