Forex Price Forecasting

Trần Hải Bình

November 26, 2020

❖ Introduction

- Background

Predicting how the Forex market going is one of the hardest things to do (than the Stock market). Stock trading involves buying and selling shares of individual companies, whereas forex trading involves exchanging – buying and selling simultaneously – cash minted by two different countries. ... Stock trading is best when markets are rising, since low liquidity makes it difficult to short sell in falling markets. There are so many factors involved in the prediction – physical factors vs psychological, rational and irrational behavior, etc. All these aspects combine to make share prices volatile and very difficult to predict with a high degree of accuracy.

What is Forex?

Forex, Foreign Exchange, is a currency price relationship between two economies, e.g. British Pound vs US Dollar or GBPUSD. The first three letters in the symbol represent the first economy called "Base Currency" and the last three letters represent the second economy called "Quote Currency". If the exchange rate of GBPUSD is 1.28818 it means that to buy \$1.28818 you pay £1, plus commission and/or Spread.

Commission, Spread and Pips

If you are exchanging with a friend, then you might use two decimal points and exchange the GBPUSD at 1.29, however, if you are exchanging via a Forex trading platform through a Forex broker, then there are fees.

Commission: It's a fixed fee that the broker charges per transaction. The commission amount is broker-dependant.

Spread: Is the difference between the buying price and selling price. This is how the broker makes a profit.

Let's take an example, if you have pounds and you want to buy dollars then the GBPUSD buy is 1.28820, conversely sell price is 1.28816. That makes the spread:

$$\text{Spread} = \text{Buy} - \text{Sell} = 1.28820 - 1.28816 = 0.00004 = 0.4e\text{-}4$$

The change in price in Forex is usually very small unless there is an event affecting the economy, so traders use PIPs to express the change.

PIP: Price Interest Point is currency specific. For most currencies, including GBPUSD it is: Change x 10000

We can consider the spread as a price change, so we can express it as:

Spread = 0.4e-4 = 0.4 pips

For example, if the selling price of GBPUSD changed from 1.28816 to 1.28827, we say the price moved up by:

1.28827 - 1.28816 = 0.00011 = 1.1 pips

- Problem

Forex Trading

Essentially, if you believe the price is going to increase, you buy the base currency (EUR in our case) using the quote currency (USD in our case) and if you believe the price is going to decrease, you sell the base currency.

Trading is associated with a strategy, take this over-simplistic strategy as an example “Buy if you believe the price will increase by at least 10 pips and sell if you believe the price will decrease by at least 10 pips.”

Your belief in price change could come from many sources, the sky is the limit, examples:

- You think a political decision would affect one of the pair economies
- You expect an out of the ordinary announcement on EUR
- You use some technical indicators and base your decision on them
- You train an ML model on historic data and ask it to predict future prices

In Trading market, we always need something that could help for the better right decision especially on Forex Trading. The installation of machine learning algorithms in the Forex trading online market can automatically make the transactions of buying/selling, by the way having a good Algorithm could help individual, neither does organizations investors in reducing emotions, having tight systems stability, saving time, depveloping a transaction strategy.

Algorithmic Trading

Algo trading is using a bot, a strategy written in code, and executing the trade automatically via an API or other means based on the bot recommendation.

An example is using a bot that will push an input data into an ML model and consult the model about the price change then trade accordingly.

- Interest

Retail/ Individual Investors whose the resource and information access capacity are not so plenty as the big financial institutes and organizations.

❖ Data acquisition and cleaning

- Data sources

The data source of Forex price is not too rare all over the world. We can easily get access through a website with historical data has been save, and download it. There are many API from brokers exchange. In this project I use the fxcmpy package, a Python Wrapper Class for the RESTful API as provided by FXCM Forex Capital Markets Ltd.

FXCM provides a RESTful API (henceforth the "API") to interact with its trading platform. Among others, it allows the retrieval of historical data as well as of streaming data. In addition, it allows to place different types of orders and to read out account information. The overall goal is to allow the implementation automated, algortithmic trading programs.

https://github.com/fxcm/RestAPI

- Data cleaning

```
start = dt.datetime(2010, 1, 1)
stop = dt.datetime(2020, 11, 25)
data = con.get_candles('EUR/USD', period='D1',start=start, stop=stop,columns=['asks', 'tickqty'])
```

```
data
```

date	askopen	askclose	askhigh	asklow	tickqty
2010-01-04 22:00:00	1.43332	1.44144	1.44563	1.42587	55015
2010-01-05 22:00:00	1.44144	1.43676	1.44845	1.43472	60382
2010-01-06 22:00:00	1.43676	1.44091	1.44357	1.42835	58258
2010-01-07 22:00:00	1.44091	1.43104	1.44476	1.43000	56049
2010-01-08 22:00:00	1.43104	1.44156	1.44399	1.42646	58253
...	...	...	...	...	...
2020-11-20 22:00:00	1.18776	1.18583	1.18915	1.18505	193200
2020-11-22 22:00:00	1.18581	1.18571	1.18608	1.18532	131
2020-11-23 22:00:00	1.18589	1.18421	1.19068	1.18006	213636
2020-11-24 22:00:00	1.18417	1.18981	1.18981	1.18389	199070
2020-11-25 22:00:00	1.18957	1.19204	1.19305	1.18825	231671

3235 rows × 5 columns

The data from the 'fxcmpy' package were pretty clean. So we do not really have to do any data cleaning besides renaming the columns for easy understanding and try to figure out why we choose ask or bid price*. We try to retrieve the historical of 'EUR/USC' exchange rate over 10 years.

date	open	close	high	low	volume
2010-01-04 22:00:00	1.43332	1.44144	1.44563	1.42587	55015
2010-01-05 22:00:00	1.44144	1.43676	1.44845	1.43472	60382
2010-01-06 22:00:00	1.43676	1.44091	1.44357	1.42835	58258
2010-01-07 22:00:00	1.44091	1.43104	1.44476	1.43000	56049
2010-01-08 22:00:00	1.43104	1.44156	1.44399	1.42646	58253
...	...	...	...	...	...
2020-11-20 22:00:00	1.18776	1.18583	1.18915	1.18505	193200
2020-11-22 22:00:00	1.18581	1.18571	1.18608	1.18532	131
2020-11-23 22:00:00	1.18589	1.18421	1.19068	1.18006	213636
2020-11-24 22:00:00	1.18417	1.18981	1.18981	1.18389	199070
2020-11-25 22:00:00	1.18957	1.19204	1.19305	1.18825	231671

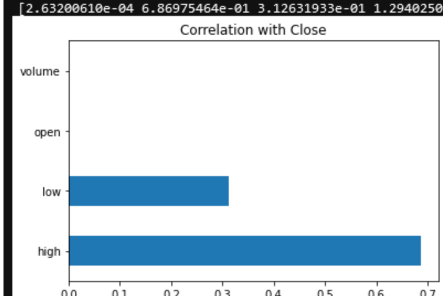
3235 rows × 5 columns

* The bid price is the highest price a buyer is prepared to pay for a financial instrument, while the ask price is the lowest price a seller will accept for the instrument. The difference between the bid price and ask price is often referred to as the bid-ask spread. So we can select ask price for simplifying analysis.

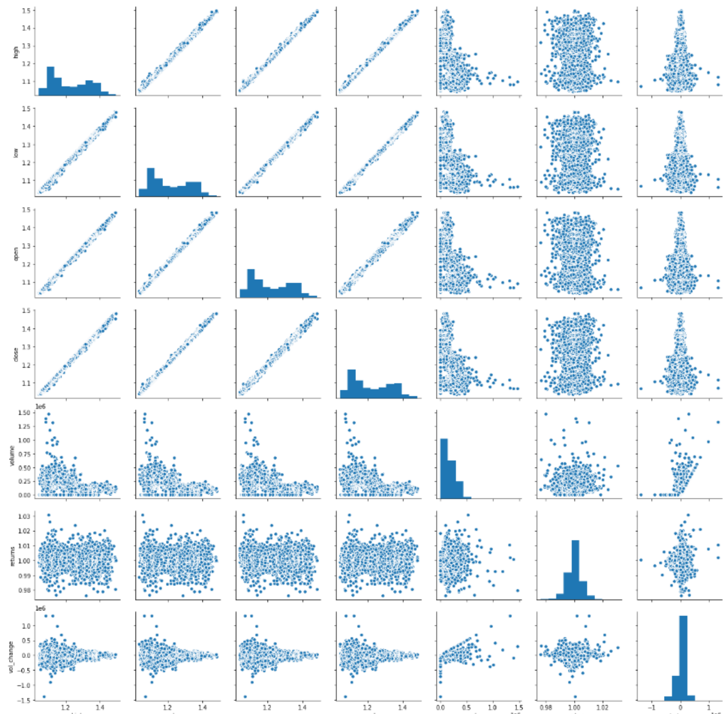
❖ Exploratory Data Analysis

```
X = data[['open','high' ,'low', 'volume']]  #independent columns
y = data.close   #target column i.e price range
from sklearn.ensemble import RandomForestRegressor
import matplotlib.pyplot as plt
model = RandomForestRegressor()
model.fit(X,y)
print(model.feature_importances_) #use inbuilt class feature_importances of tree based regressor
#plot graph of feature importances for better visualization
feat_importances = pd.Series(model.feature_importances_, index=X.columns)
feat_importances.nlargest(10).plot(kind='barh', title='Correlation with Close')
plt.show()
```

```
[2.63200610e-04 6.86975464e-01 3.12631933e-01 1.29402502e-04]
```



We create 2 new features by simply calculation of the daily change in price and trading volumes of the stock on a certain day and the next. We see that the daily return and volume trade of the 'EUR/USD' we chose to predict tend to increase over time, but not like the Stock market, the purpose of prediction not only to expect the price going up by its value but also determine downtrend for every single day, that's the keypoint to make Forex market different from Stock market, becoming more hard to play with and the fluctuations in more chaos than the trend in overall that we see. Please be aware that this distribution only applies to this 'EUR/USD' exchange rate which I chose alone. If you chose any others in the Forex market. The results can be quite different.



For our chosen 'EUR/USD' the distribution of daily return and volume change suggest that the Money exchange rate which issued the 'EUR/USD' is still making profit over its life cycle. This implied by the close price reduced overtime. For the volume trading increase, can come from 2 reasons: As the economic gets bigger the value of the money getting higher will issue more shares to increase the liquidity of the Forex market and the improvement in power of its country market and economy in overall.

In general, a higher exchange rate is better. This is because, when you exchange currencies, you'll get more of the foreign currency you're buying.

For example, let's say that you intend to exchange £100,000 into euros, to buy a villa on Spain's Costa del Sol. You look at the pound to euro exchange rate one day, and it's 1.10, meaning you'd get €110,000.

Then, you look at the exchange rate a week later, and it's risen to 1.120, meaning you'd now get €120,000 when you exchange currencies. In this case, a higher exchange rate is better, because it means you'll get more euros for your villa.

❖ **Predictive Modeling**

The task to predict Forex money exchange rate (close price) falls into the time series forecasting category. Time series data can be phrased as **supervised learning**. Given a sequence of numbers for a time series dataset, we can restructure the data to look like a supervised learning problem. We can do this by using previous time steps as input variables and use the next time step as the output variable. Despite the similarities, time series forecasting is quite different from other supervised regression tasks. Time Series Forecasts Different from other Supervised Learning Problems:

- It is unprincipled to use k fold randomized cross-validation (because maintaining temporal order is important)
- We are extending a trend outside the range of observed data
- Understanding uncertainty around point estimations can be more important than the point estimates themselves for decision making, especially in a business setting.

In this study I will try to test the results of different models like **ARIMA**, **LSTM-RNN** and **Prophet**

Performances of different models:

ARIMA	0.000129
LSTM	0.001934
Prophet	0.004429

The **ARIMA** model currently gives the best result among 3 models for the task to make predictions on the data of the stock that we chose. The next best performed models are **LSTM** and then **Prophet**.

❖ AutoRegressive Integrated Moving Average(ARIMA)

Arima model: Arima model is one method for forecasting time series, it is assumed that past value of the series plus previous error terms contain information for the purpose of forecasting. The main advantage of Arima forecasting is that it requires data on time series in question only. However, Arima model are essentially backward looking, they are generally poor at predicting turning points, unless the turning point represents a return to a long-run equilibrium (Meyler et al., 1998).

Exchange rate forecasting: Exchange rate forecasting means estimating the rate which will be any of future time.

Methodology:

Arima model is used as the main methodology of this research. Arima can be fully written as Autoregressive Integrated Moving Average. This model was showed in publish by Box and Jenkins in 1970. Arima is constructed by three parts: AR (autoregressive part), I (integrated part), MA (moving average part) first show in publish. In order to estimate Arima model, we have to do 4 steps as follow: Recognizing model, estimating variables and choosing model, testing model and forecasting.

○ **AR**: Autoregression. A model that uses the dependent relationship between an observation and some number of lagged observations.

○ **I**: Integrated. The use of differencing of raw observations (e.g. subtracting an observation from an observation at the previous time step) in order to make the time series stationary.

○ **MA**: Moving Average. A model that uses the dependency between an observation and a residual error from a moving average model applied to lagged observations.

Each of these components are explicitly specified in the model as a parameter. A standard notation is used of ARIMA(p, d, q) where the parameters are substituted with integer values to quickly indicate the specific ARIMA model being used.

The parameters of the ARIMA model are defined as follows:

● p: The number of lag observations included in the model, also called the lag order.

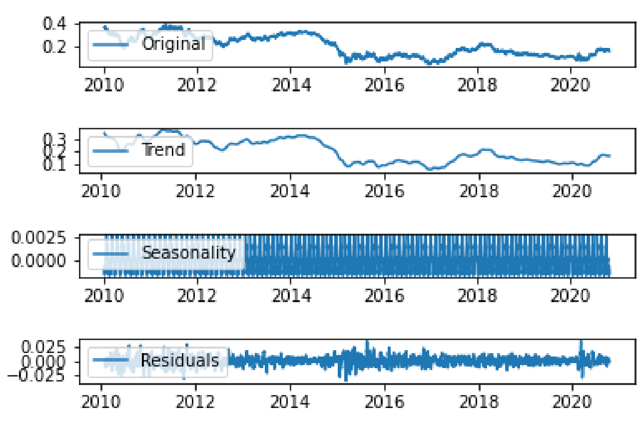
● d: The number of times that the raw observations are differenced, also called the degree of differencing.

● q: The size of the moving average window, also called the order of moving average. In order to find the best parameter for the model I have tried out 3 different approaches:

Approach 1: through PACF & ACF Function

In the AR model, Partial Auto Correlation Function(PACF) graph is used to find (p) value and in MA model, Auto Correlation Function(ACF) graph to find (q) value. Integration Function is used to find the (d) value. ie, the differentiation

- **Seasonal trend, Trend Line and Residuals in the data:**

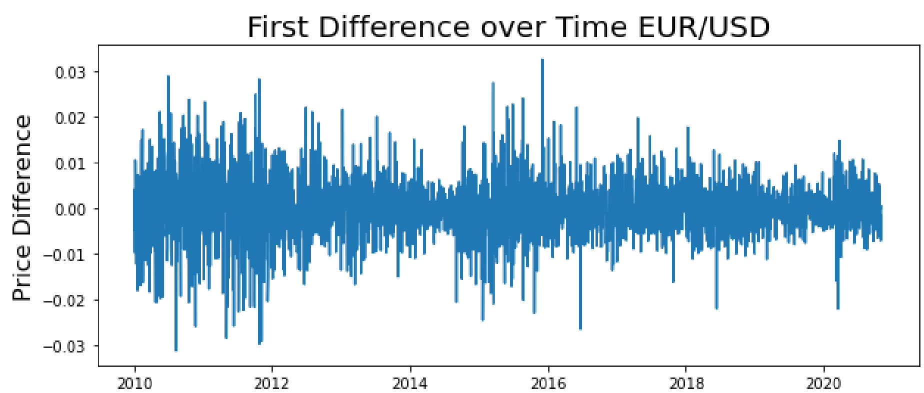


Trend line shows an upward trend gradually after a big downtrend in the mid of 2015.

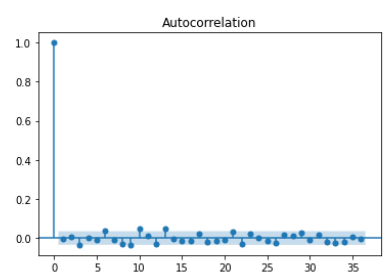
From the seasonality graph shows the yearly seasonality of the data.

Residuals are the irregularity in the data, that it didn’t have any shape in it.

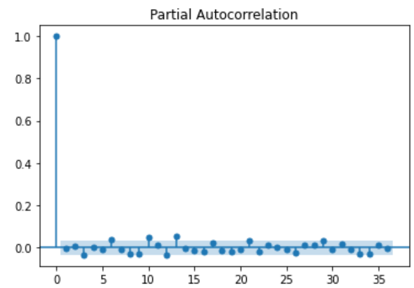
- **Find First different to de-trend and make the data more stationary:**



- **Autocorrelation and Partial Auto-Correlation Functions:**

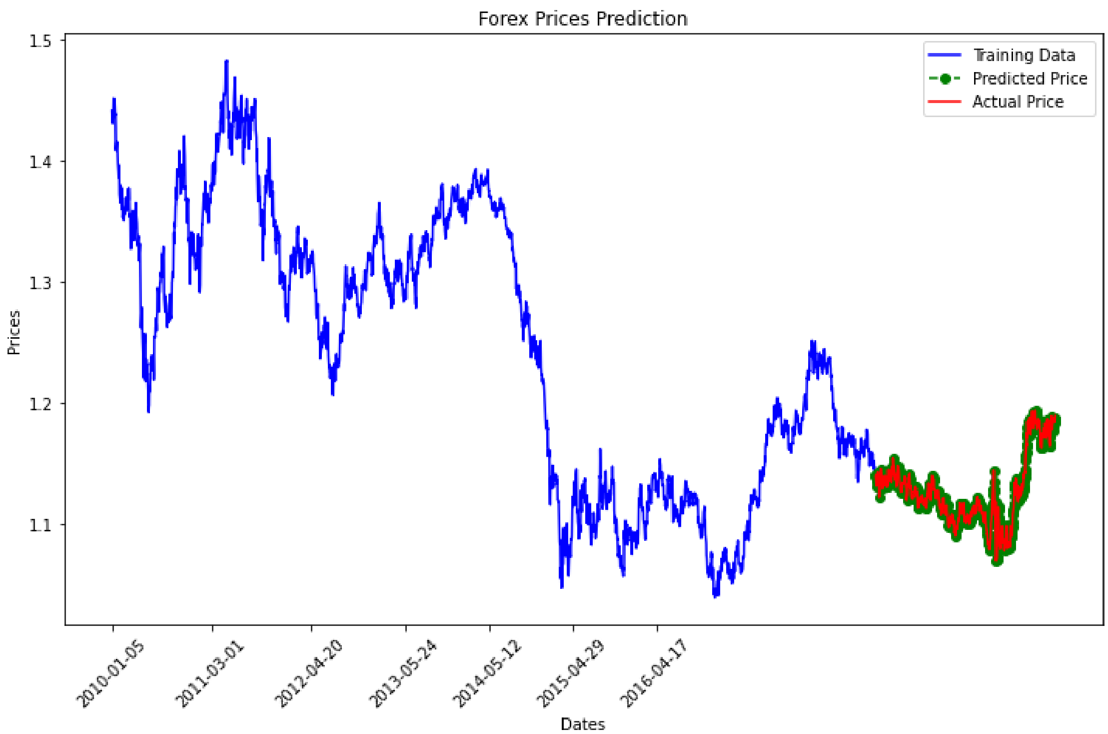


ACF isn't that informative



PACF also doesn't tell us much

So after some trial and error plus experience from previous projects to forecast time-series data I go with this order for the parameter(p,d,q) to fit the model. Here the p value as 2, d value as 1 and q value as 0.



Approach 2: Auto-Arima

```
from pmdarima.arima import auto_arima
arima_model =  auto_arima(train,start_p=0, d=1, start_q=0,
                          max_p=5, max_d=5, max_q=5, start_P=0,
                          D=1, start_Q=0, max_P=5, max_D=5,
                          max_Q=5, m=12, seasonal=True,
                          error_action='warn',trace = True,
                          supress_warnings=True,stepwise = True,
                          random_state=20,n_fits = 50 )

Performing stepwise search to minimize aic
```

The advantages of Auto-Arima is obviously you have less work to do in exchange for the understanding an control over the elements of the model

```
Best model:  ARIMA(0,1,1)(4,1,1)[12]
Total fit time: 1749.394 seconds
```

Approach 3: Grid Search

```python
# evaluate combinations of p, d and q values for an ARIMA model
def evaluate_models(dataset, p_values, d_values, q_values):
    dataset = dataset.astype('float32')
    best_score, best_cfg = float("inf"), None
    for p in p_values:
        for d in d_values:
            for q in q_values:
                order = (p,d,q)
                try:
                    mse = evaluate_arima_model(dataset, order)
                    if mse < best_score:
                        best_score, best_cfg = mse, order
                    print('ARIMA%s MSE=%.8f' % (order,mse))
                except:
                    continue
    print('Best ARIMA%s MSE=%.8f' % (best_cfg, best_score))

# evaluate parameters
p_values = [0, 1, 2, 4, 6, 8, 10]
d_values = range(0, 3)
q_values = range(0, 3)
warnings.filterwarnings("ignore")
evaluate_models(series.values, p_values, d_values, q_values)
```

```
ARIMA(0, 0, 0) MSE=0.00881252
ARIMA(0, 0, 1) MSE=0.00235609
ARIMA(0, 0, 2) MSE=0.00081418
ARIMA(0, 1, 0) MSE=0.00001953
ARIMA(0, 1, 1) MSE=0.00001954
ARIMA(0, 1, 2) MSE=0.00001954
ARIMA(0, 2, 0) MSE=0.00003882
ARIMA(0, 2, 1) MSE=0.00001955
ARIMA(0, 2, 2) MSE=0.00001957
ARIMA(1, 0, 0) MSE=0.00001947
ARIMA(1, 0, 1) MSE=0.00001947
ARIMA(1, 0, 2) MSE=0.00001947
ARIMA(1, 1, 0) MSE=0.00001954
ARIMA(1, 2, 0) MSE=0.00002789
ARIMA(1, 2, 1) MSE=0.00001962
ARIMA(1, 2, 2) MSE=0.00001959
ARIMA(2, 0, 0) MSE=0.00001947
ARIMA(2, 1, 0) MSE=0.00001954
ARIMA(2, 2, 0) MSE=0.00002553
```

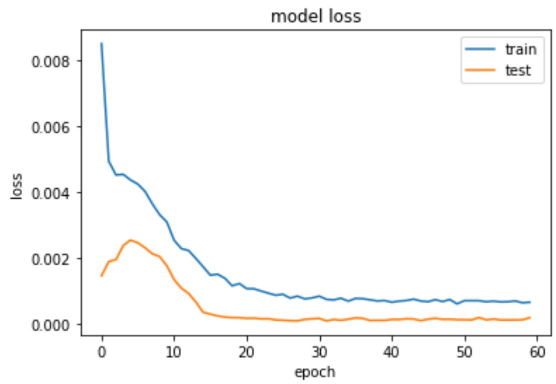
The last approach I tried to get the best order for the ARIMA(order=(p,d,q) suggest 2 more promising model

with ARIMA(0, 1, 0) MSE=0.00001953 and ARIMA(2, 1, 0) MSE=0.00001954

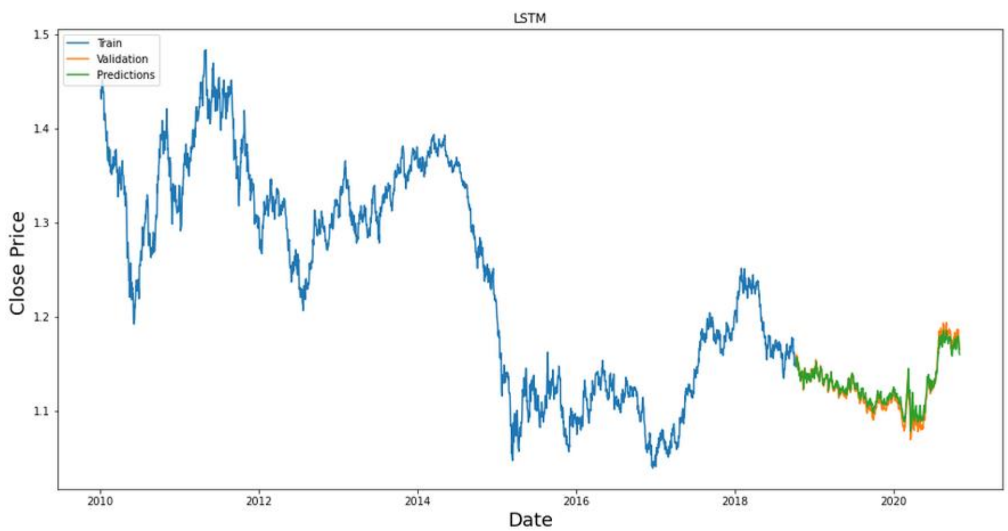
❖ Recurrent Neural Network – Long Short Term Memory (LTSM)

The LSTM networks are well-suited for time series problems. Explaining the details of this layer is outside the scope of this story (as I followed the Data Science course), for details: click here

This model gave the best error of 0.001934 and best validation error of 0.0025. Here is what the Training loss vs Validation loss looked like:



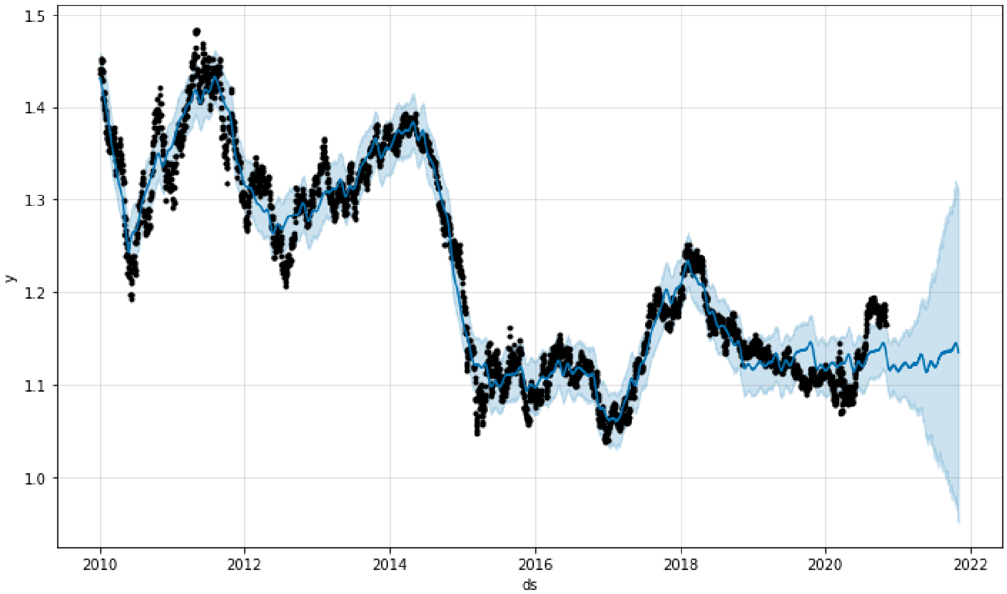
This is how the prediction looked with above model:

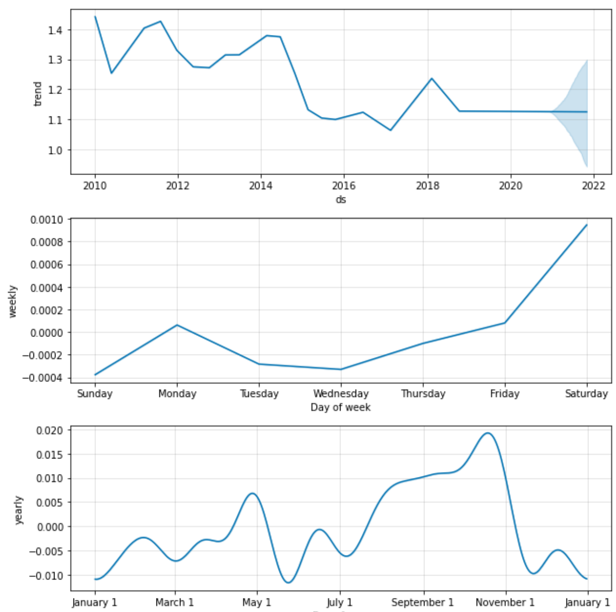


❖ Prophet (by Facebook)

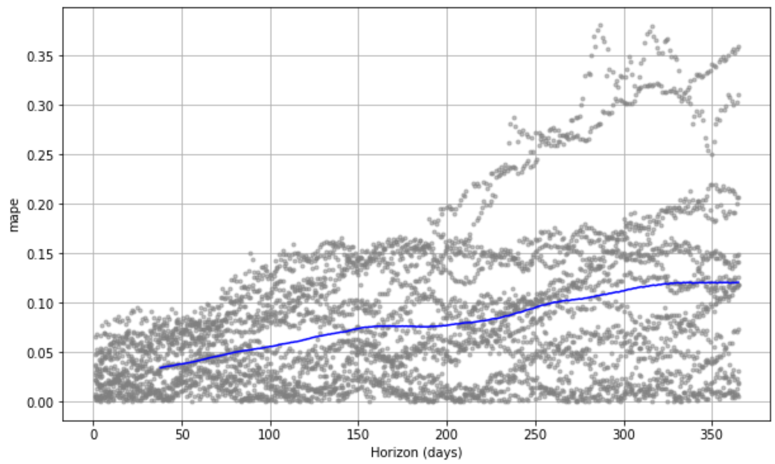
Prophet uses a decomposable time series model with three main model components: growth, seasonality and holidays. They are combined using the equation:

$$y(t) = g(t) + s(t) + h(t) + e(t),$$





Even though the Prophet model performed worst among the 3 models. The component forecast part of the model suggests a very promising capability of predicting trend and yearly seasonality. As you can see in the Figure. The model successfully captured the phenomenal **“Sell in May and go away"** –a-well-known financial-world adage. It is based on the historical underperformance of some stocks in the "summery" six-month period commencing in May and ending in October, compared to the "wintery" six-month period from November to April. If an investor follows this strategy, they would divest their equity holdings in May (or at least, the late spring) and invest again in November (or the mid-autumn).



The blue line shows the MAPE, where the mean is taken over a rolling window of the dots. We see for this forecast that errors around 10% are typical for predictions one month into the future, and that errors increase up to around 30% for predictions that are a year out.

❖ Conclusions

These model can be helpful for individual or organizational users in some cases, it help to see the vision and improving the ability to make a decision before transactions. We should not look at the small value of MSE or RMSE to enhence the belief in models because the change in price too small could let us lose a lot of pips in each transactions.

This research examines applied ability of Arima model in forecasting foreign exchange rate in Forex Trading, in case of foreign exchange rate between Euro and United State Dollar. The results show that Arima model is absolutely suitable for forecasting. The policy makers should apply Arima model in forecasting foreign exchange rate in any broker platforms. Specially, in foreign exchange business of the commercial joint stock banks in Vietnam, the financial planners should apply Arima model in forecasting as well as care the results of forecasting in measuring foreign exchange rate risk in order to make more benefit for their bank.

The forecasting results of our model show that foreign exchange rate EUR/USD in 2020 tends to

increase. In foreign exchange business, gain/loss from foreign exchange business depends on foreign exchange rate fluctuation and foreign currency position (Long, 2010). In condition of foreign exchange rate increase, one commercial joint stock bank will get gain in foreign exchange business if this commercial joint stock bank sustains long foreign currency and vice versa. According to forecasting results of this research, foreign exchange rate EUR/USD increase continuously, therefore the managers of the commercial joint stock banks in Vietnam should care about this result and maintain long foreign currency position in foreign exchange business.

Although, in our multi-step predictions graph, not all predictions were right, remember two things:

1 — This is only the start and this model has a huge room for improvement as suggested in the “Continuing and Expanding The Research” section.

2 — It is enough to have a certain percentage of predictions right, to make a profit.

I feared in the beginning that the results will follow a “mean reversion” trend, where the prediction will try to go back to the previous price average. But this wasn’t the case.

❖ Future directions

We can run the model for all the money exchange rates in the world Forex market list to see which model performed best on which sector of the economy for example: ARIMA may be good at predicting Forex price of the specific money (EU/USD, GBP/USD..). But Prophet or LSTM may be better in others like Gold, Oil,...