Course Report: Time Series and Survival Analysis

Nan He

1 Main objective of the analysis

Acknowledgement: this is a course project for IBM Machine Learning professional certificate. The project notebook can be accessed here.

In this report, the data I will be using is the stock price time series data for Apple (AAPL), Boeing (BA), and Walmart (WMT) during 2017-1 and 2018-1, the data are acquired here. **The goal of this report is to use time series methods to forecast the stock price in one month.**

The target includes:

- Prepare an clean the data for modeling.
- Train different time series models on those three stock price series.
- Forecast one month into the future, comparing the predictions with the actual stock price.

I will be using both **smoothing methods**, **ARIMA**, and **SARIMA** for modeling. There are **some possible snags** that can be problematic for the analysis including:

- Stock price has strong trend and vague seasonality, the assumption is difficult to make.
- It is doubt that whether stock price solely depends on past data? I would consider not.

Those problems can be addressed by:

- I will assume an close-to-linear up-going trend, and a monthly seasonality (since weekly seasonality is to short for predictions).
- Due to the nature of the stock market, the prediction cannot be very accurate. We should focus mainly on some general patterns and trends.

2 Description of the data set

The original data set is the stock price for 30 different companies from 2017 to 2018. I took Apple (AAPL), Boeing (BA), and Walmart (WMT), representing different sectors. The following is a **data dictionary**. The feature columns are:

- Date: the date label
- Open: the open price
- High: the highest price of the day
- Low: the lowest price of the day
- Close: the close price
- Volume: the trade volume
- Name: the name/label of the stock

The date is a date label that can be converted to date-time, the Open, High, Low, Close are in float64, and Volume is int. The Name is a string object.

3 Data exploration, cleaning, and feature engineering

The data need to be cleaned. Here is the list of actions I took:

- Separate each Name into a single dataframe. making three dataframe characterizing AAPL, BA, and WMT.
- Drop "Name", "Volume". Since they are not related with our goal.
- Convert data to pandas datetime.
- Check NaN, fill with average data from the previous and next day.

The example data after cleaning looks like:

Date	Open	High	Low	Close	Volume	Name
2017-01-03	115.80	116.33	114.76	116.15	28781865	AAPL
2017-01-04	115.85	116.51	115.75	116.02	21118116	AAPL
2017-01-05	115.92	116.86	115.81	116.61	22193587	AAPL
2017-01-06	116.78	118.16	116.47	117.91	31751900	AAPL
2017-01-09	117.95	119.43	117.94	118.99	33561948	AAPL

Figure 1: AAPL stock price data example.

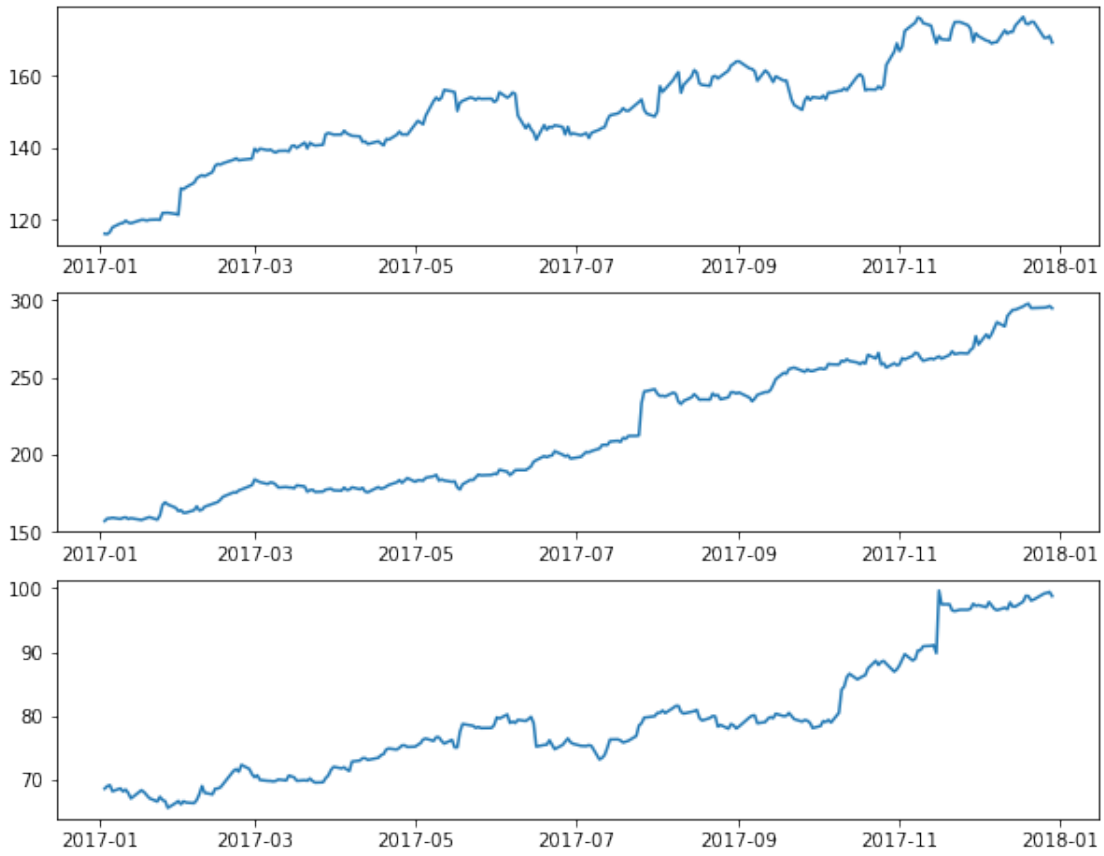

Figure 2: Close price for AAPL, BA, and WMT from 2017-1 to 2018-1.

The data after cleaning contains 251 data labels for each stock. Here are plots for all three stock, taking the Close price of each day:

There are some interesting observations here, for example:

- There is a clear growing trend in this series. It is necessary to assume a linear trend or conduct differencing.
- The data have some sudden changes, for example, WMT has a steep growth in 2017-11. It needs to be very carefully considered when choosing seasonality.

4 Stationary, and the assumptions about trend and seasonality.

In this report, I take the "Close" price of everyday for time-series modelling. Before going into modeling, I firstly confirmed that the stock series is non-stationary. To take account for the trend and seasonality, I have to make the following assumptions:

- Their trends are additive, each components are linear.

- The seasonal components are additive, the period is either 5 days or 21 days. This is a rough estimation, and we will test whether the seasonal components is necessary.

The data is then split into train and test set, including 230 and 21 labels, respectively. Every model below will be trained on the train set, and forecast one month (21 days) into the further. The prediction results will be compared with the test set; their mean square error (MSE) will be used as the evaluation metric. In the next four sections, four different type of models will be trained, and forecast on the test set.

5 Double Exponential Smoothing

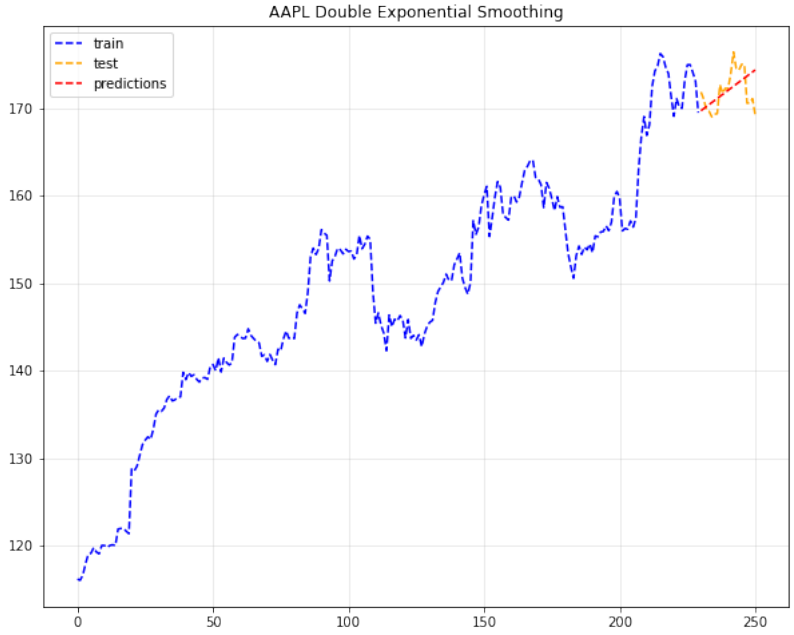

Figure 3: Double exponential smoothing for AAPL training on 2017-1 to 2017-12, predicting 2018-1.

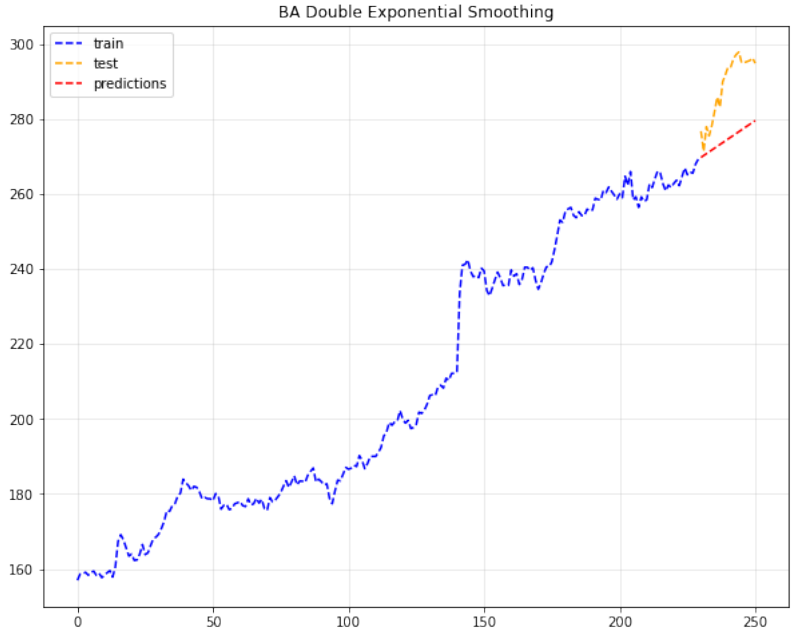

Figure 4: Double exponential smoothing for BA training on 2017-1 to 2017-12, predicting 2018-1.

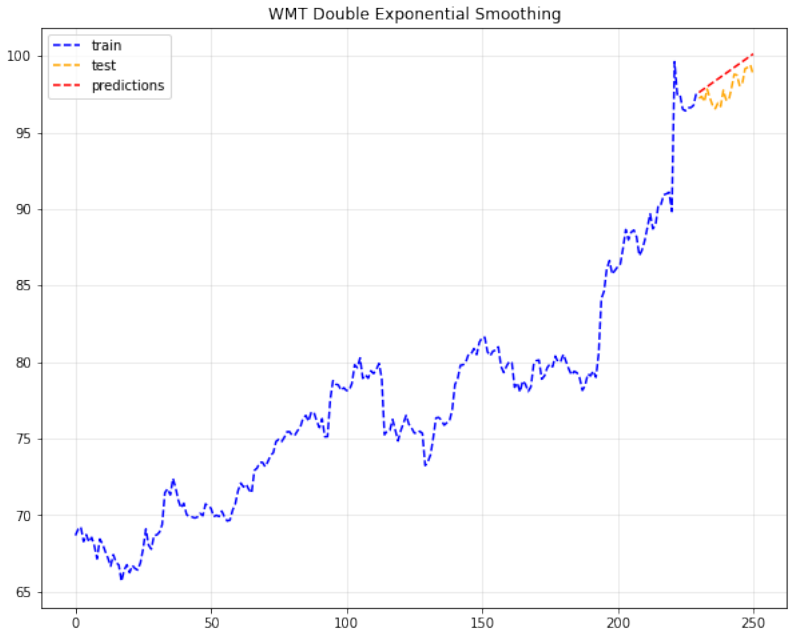

Figure 5: Double exponential smoothing for WMT training on 2017-1 to 2017-12, predicting 2018-1.

6 Triple Exponential Smoothing

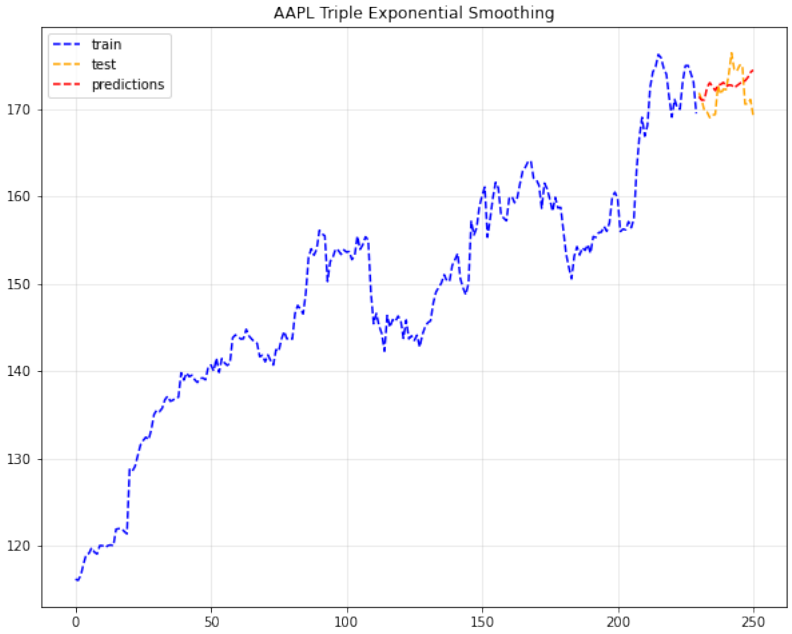

Figure 6: Triple exponential smoothing for AAPL training on 2017-1 to 2017-12, predicting 2018-1.

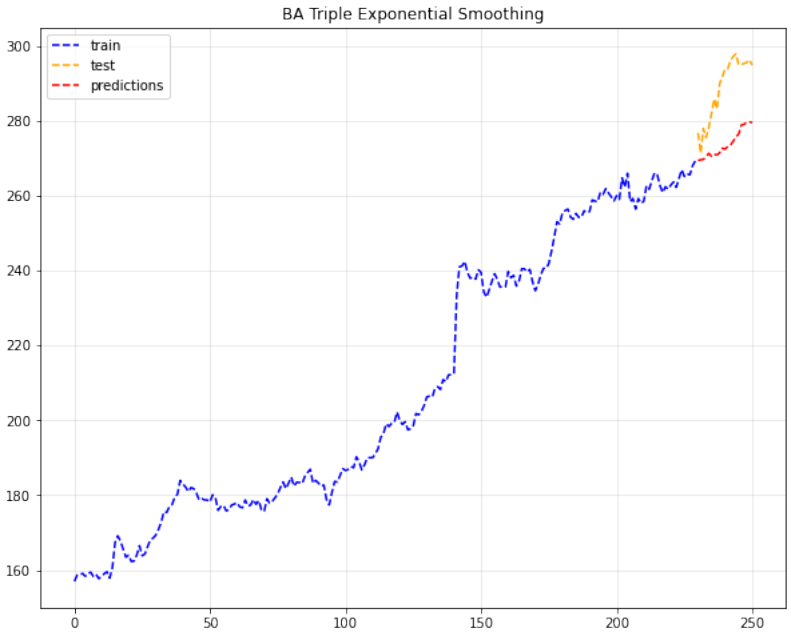

Figure 7: Triple exponential smoothing for BA training on 2017-1 to 2017-12, predicting 2018-1.

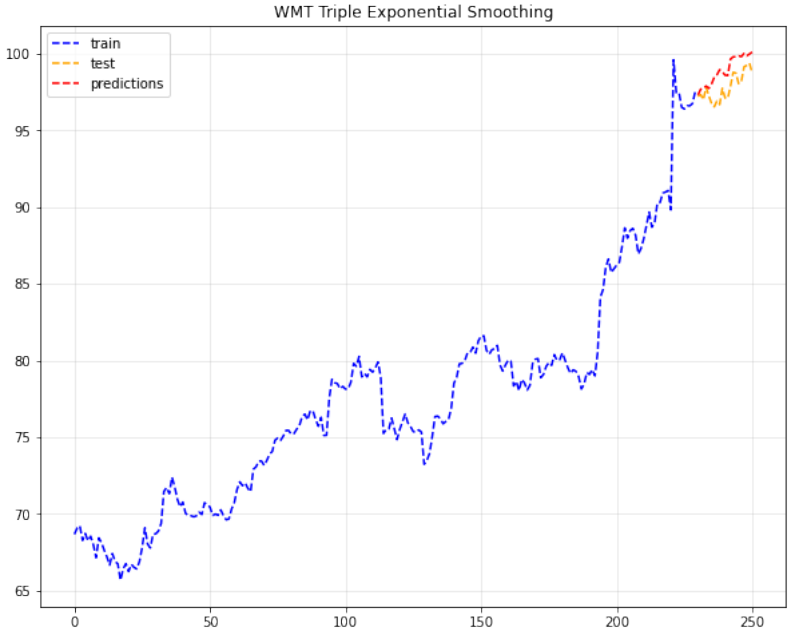

Figure 8: Triple exponential smoothing for WMT training on 2017-1 to 2017-12, predicting 2018-1.

7 ARIMA

The ARIMA model needs to determine the AR, I, and MA orders. I do it by ploting the Autocorrelation and Partial-Autocorrelation Functions (ACF and PACF). Here I take one example to show

the parameter tuning process: Start with AAPL, the ACF and PACF plots are:

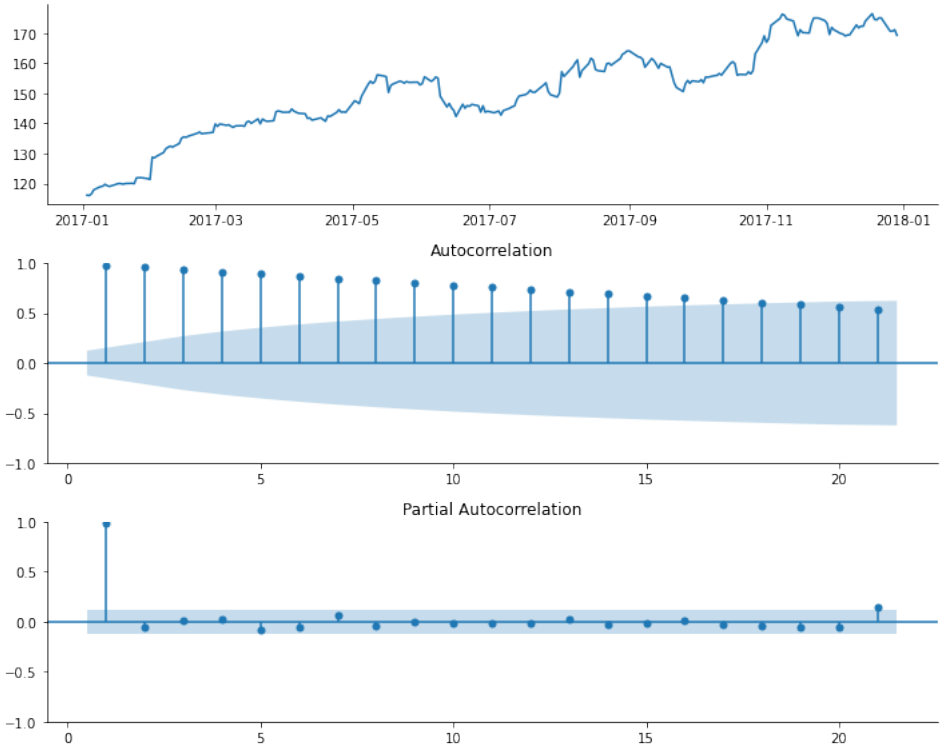

Figure 9: AAPL ACF and PACF plots.

- The autocorrelation is large, try using difference I=1.
- Following the Box-Jenkins procedure, I choose a AR + MA model, with order=1, determined from the PACF plots.

Here is the statistical summary of a (1,1,1) model:

Dep. Variable:	Close	**No. Observations:**	251
Model:	SARIMAX(1, 1, 1)	**Log Likelihood**	-486.482
Date:	Tue, 15 Feb 2022	**AIC**	980.965
Time:	10:22:23	**BIC**	995.051
Sample:	0	**HQIC**	986.634
	- 251		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
drift	0.0010	0.001	0.657	0.511	-0.002	0.004
ar.L1	-0.0982	1.611	-0.061	0.951	-3.256	3.059
ma.L1	0.1499	1.612	0.093	0.926	-3.010	3.310
sigma2	2.8817	0.165	17.430	0.000	2.558	3.206

Ljung-Box (L1) (Q):	0.02	**Jarque-Bera (JB):**	118.32
Prob(Q):	0.89	**Prob(JB):**	0.00
Heteroskedasticity (H):	2.17	**Skew:**	0.16
Prob(H) (two-sided):	0.00	**Kurtosis:**	6.36

Figure 10: AAPL AR=1, I=1, MA=1 model statistics.

The residue also looks ok:

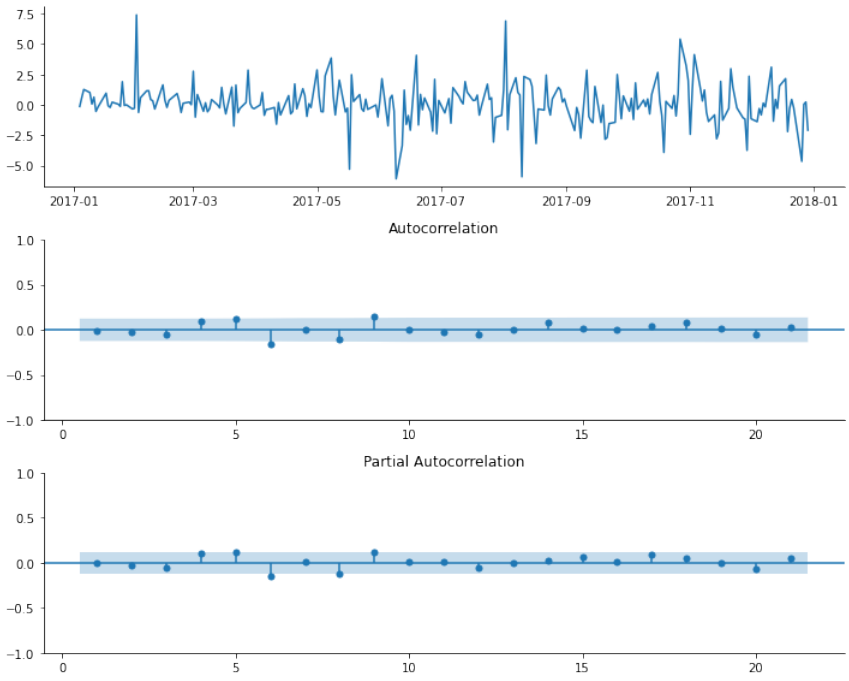

Figure 11: AAPL AR=1, I=1, MA=1 model residue, ACF, and PACF.

So the model I will be using for AAPL is (1,1,1), similar procedures are conducted BA and WMT,

too. I tested their forecast the same way as previous smoothing models:

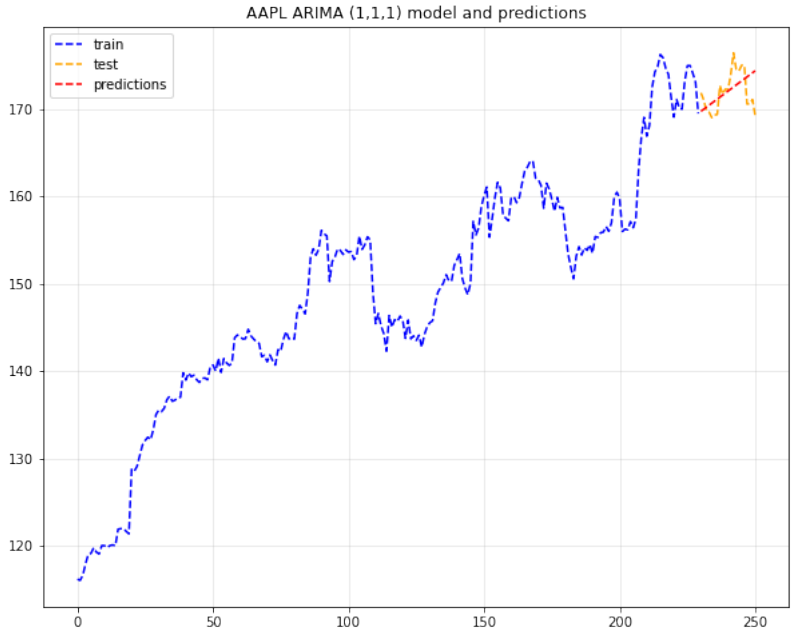

Figure 12: AAPL ARIMA (1,1,1) model, training on 2017-1 to 2017-12, predicting 2018-1.

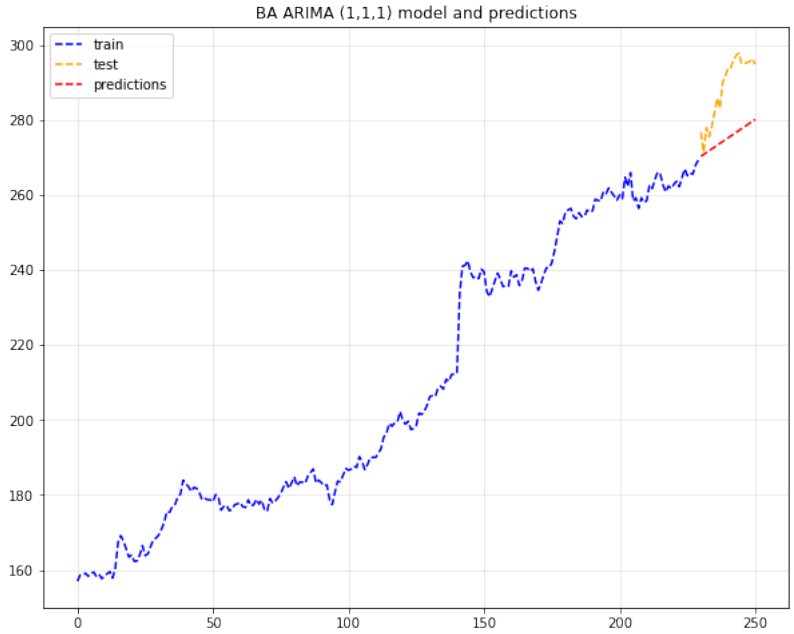

Figure 13: BA ARIMA (1,1,1) model, training on 2017-1 to 2017-12, predicting 2018-1.

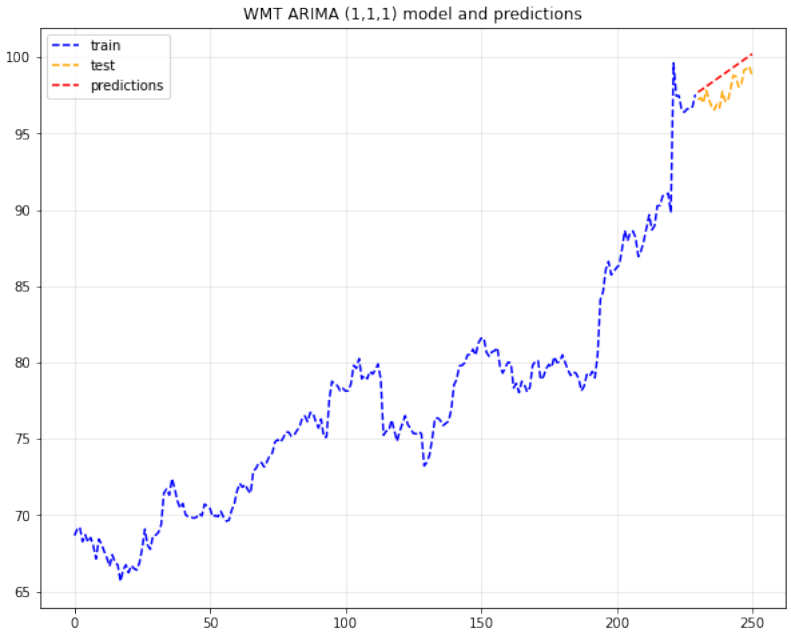

Figure 14: WMT ARIMA (1,1,1) model, training on 2017-1 to 2017-12, predicting 2018-1.

8 SARIMA

The SARIMA model adds seasonal components to ARIMA. As we assumed before, a 21-day seasonality will be considered. The parameter is also (1,1,1).

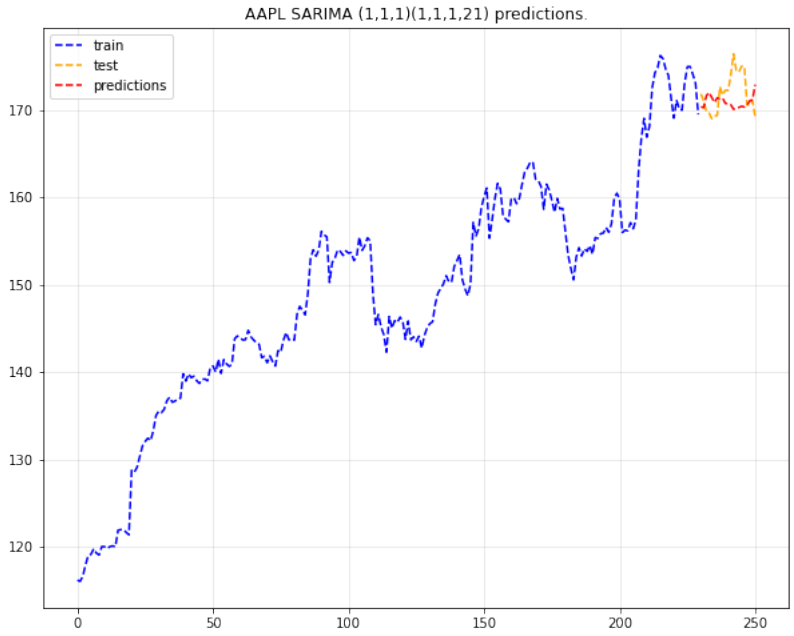

Figure 15: AAPL SARIMA (1,1,1)(1,1,1,21) model, training on 2017-1 to 2017-12, predicting 2018-1.

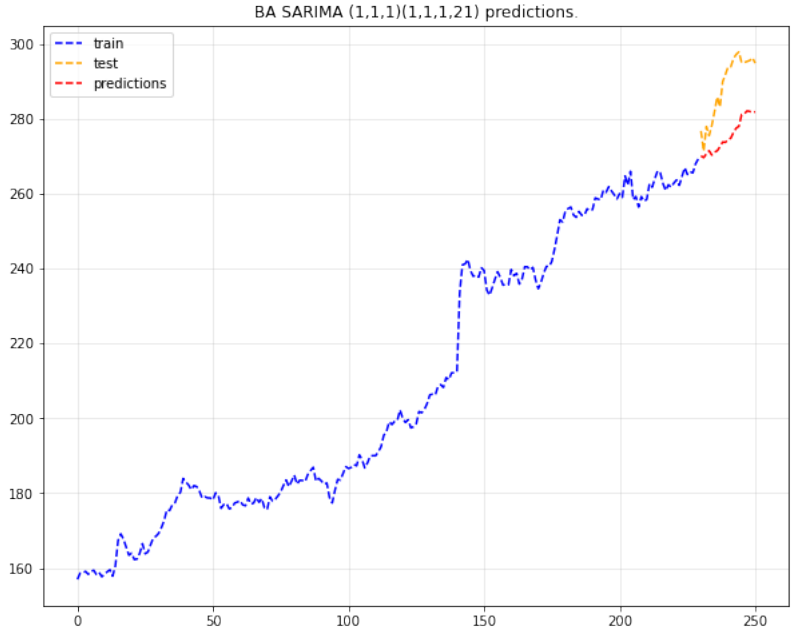

Figure 16: BA SARIMA (1,1,1)(1,1,1,21) model, training on 2017-1 to 2017-12, predicting 2018-1.

9 Final model choices and analysis

Here we compute and compare the mean square error of all model predictions:

From the table, we can see that **ARIMA model works best in general**. Double Exponen-

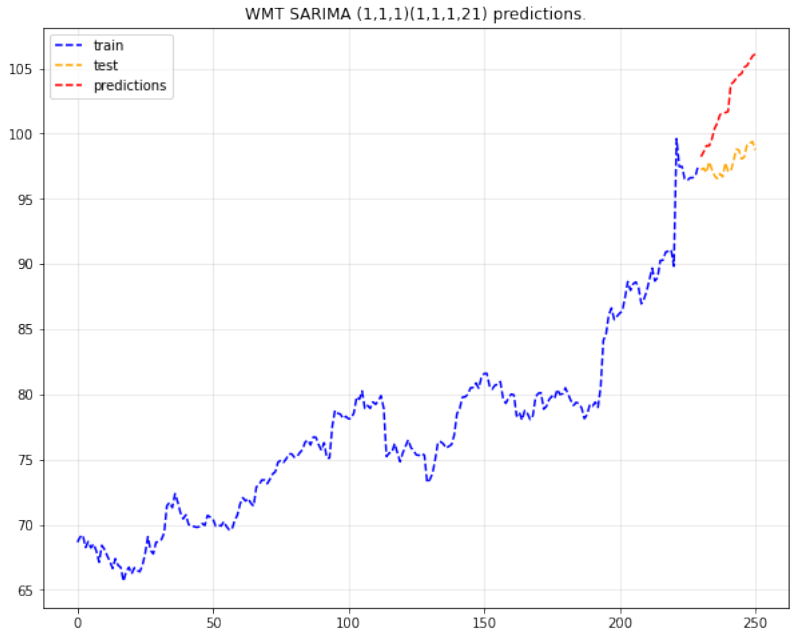

Figure 17: WMT SARIMA (1,1,1)(1,1,1,21) model, training on 2017-1 to 2017-12, predicting 2018-1.

Mean Square Error	AAPL	BA	WMT
Double Exponential Smoothing	102.0	4880.2	30.1
Triple Exponential Smoothing	129.3	5452.9	34.5
ARIMA	101.9	4552.2	34.4
SARIMA	174.0	4242.4	532.4

tial Smoothing is also prefered. Two models containing seasonal components (Triple Exponential Smoothing and SARIMA) are less prefered.

10 Key Findings and Insights

The results of the modeling indicates:

- The stock price series are dominated by upward trend components.
- The seasonal components is not correctly captured, either it is not important, or it has other external factors contributing.
- Stock price are not really predictable, the predictions generally have high MSE.

11 Summary and suggestions for next steps

In summary, this report explores time series modeling on stock prices. The model do capture the general trend, but not the detailed structures of the stock price series. The problem with the modeling is two fold:

- They cannot capture seasonality in stock prices correctly, since it is not a simple periodic time stamp.
- They fail to capture the complex structures.

Therefore, the **possible next steps** include:

- Use deep learning methods to try to capture the complex seasonality and substructures.
- Consider what external factors affect stock prices, and involve them into the model.