

Time series forecasting

Saurabh Burewar (B18CSE050)

ARIMA

1. Check if data is stationary

First, we use the augmented Dickey-Fuller root test to check for stationary data. If the p-value is more than 0.05 and data is not stationary, we perform differencing.

2. Differencing

We just keep differencing our data and check if it is stationary in every iteration. When the data becomes stationary, we stop and return the dataset. With this, we also get the degree of differencing. We get $d = 2$.

```
Data is not stationary
ADF test statistic: -3.611145916091815
p-value: 0.005551482341980969
Number of lags used: 10
Number of obs used: 59
Degree of differencing: 2
Data is now stationary
```

3. Fitting the model

The ARIMA model consists of two models, the AR and the MA models. The implementation of both of these models is similar.

We split our data into train and test and use a linear regression model for building both AR and MA. We get the coefficients from our linear regression model which is θ and our intercept and return the results.

We run the AR model for 20 iterations and get the best p-value result. We get $p = 7$.

```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing
dftrain2['Predicted'] = Xtrain.dot(
The RMSE is: 0.18421632993709045 , Value of p: 19
arima.py:26: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing
dftrain2['Predicted'] = Xtrain.dot(
The RMSE is: 0.13501816458472723 , Value of p: 20
arima.py:26: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing
dftrain2['Predicted'] = Xtrain.dot(
The RMSE is: 0.027590310611580772 , Value of p: 7
Best p-value from AR model: 7

```

Next we get residuals that are the difference between the original values and predicted values by the AR model and put these into another dataframe which we use for the MA model.

We train the MA model for about 12 iterations and get the best q-value result. We get q = 1.

```

dftrain2['Predicted'] = Xtrain.dot(
The RMSE is: 0.03790304289242748 , Value of q: 11
arima.py:26: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

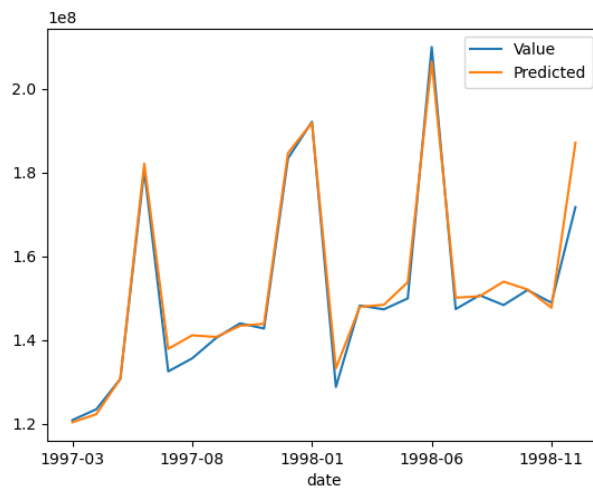
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing
dftrain2['Predicted'] = Xtrain.dot(
The RMSE is: 0.04337445915271737 , Value of q: 12
arima.py:26: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing
dftrain2['Predicted'] = Xtrain.dot(
The RMSE is: 0.031984021565677506 , Value of q: 1
Best q-value from MA model: 1

```

4. Getting the original data

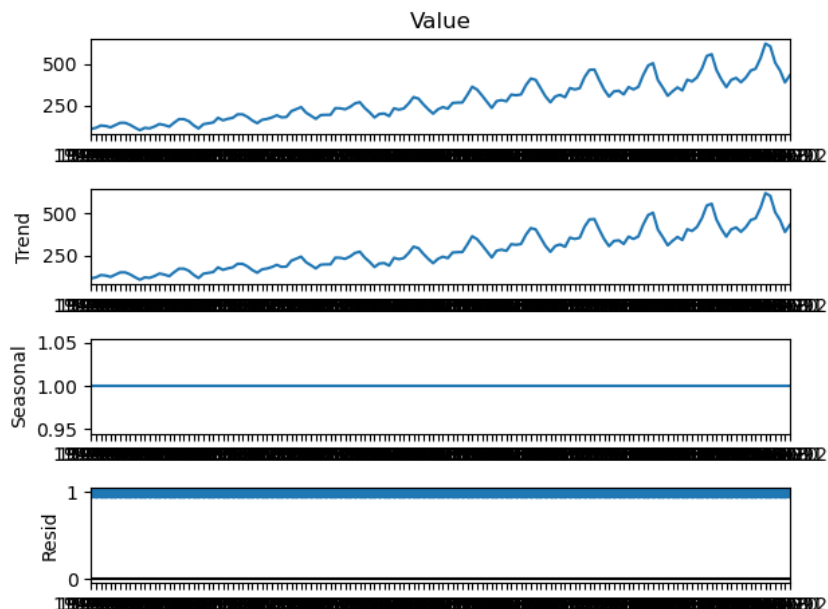
Now, we first need to reverse all the differencing that we did on the data before on the dataframe. We just do the corresponding operations to reverse it and plot our original and predicted values. We get the following result -



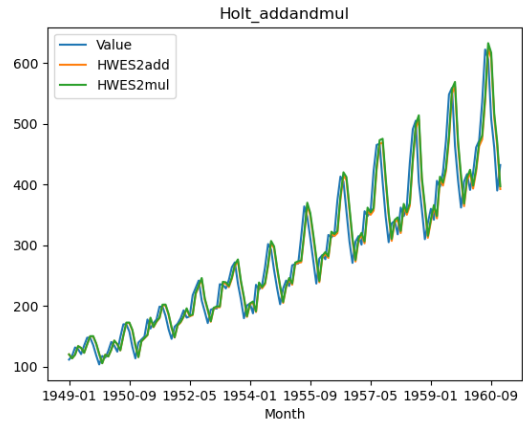
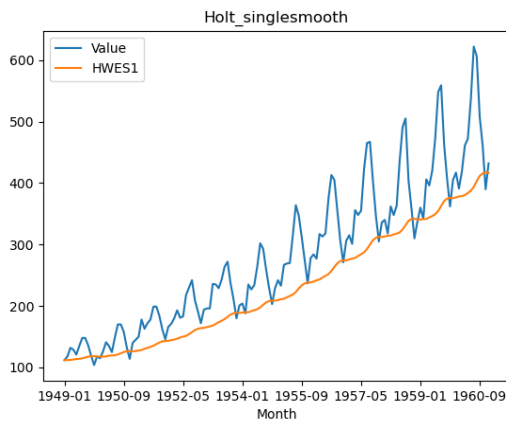
Holt-Winters method

This method is implemented using the functions available in the statsmodels module.

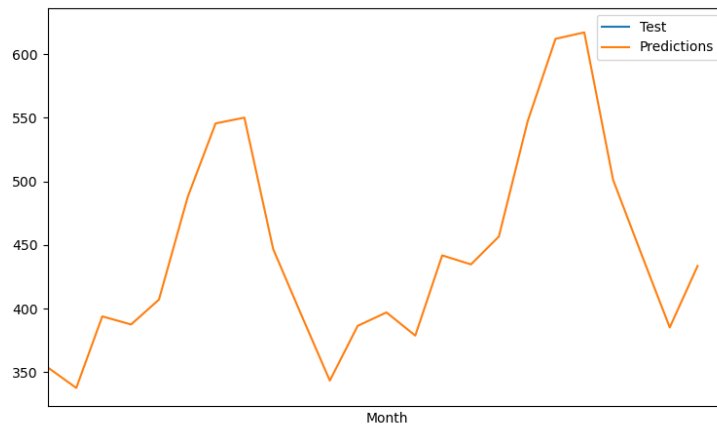
We check the data for seasonality, trends and residual using the seasonal decomposition. This gives us the following result -



Now, we try different levels of exponential smoothing to see which gives the best results. First, we do single smoothing, which gives results in the left figure. Then for double, we get the right figure which is a promising result.



Finally, we train the model using the second configuration and then predict the values. We get the following predictions-



References

1. <https://www.machinelearningplus.com/time-series/time-series-analysis-python/>
2. <https://www.machinelearningplus.com/time-series/arma-model-time-series-forecasting-python/>
3. <https://www.analyticsvidhya.com/blog/2021/08/holt-winters-method-for-time-series-analysis/>