CSC 591/ECE 592 IoT Analytics - Project 4
- Forecasting

Objectives

To use various forecasting algorithms to determine the best model for a time series. For this analysis you can use any statistical package, such as MatLab, R, SAS, or use Python with all the available statistical functions. You do not need to use a single package. You may use different packages/functions to execute different tasks

Data Set

You will use the same data set you obtained in Project 2. If your data set **does not appear to be good** (cannot yield good fitting model), additional data samples may be collected.

Task 1. Check for Stationarity and Non-Stationary Properties

1.1 Plot the entire time series, that is, both training and test set and check for stationarity.

1.2 If the data set does not appear to be stationary, then check three features for non-stationary time-series, that is, trend, variance, and seasonality.

1.3 If the data set does not appear to be stationary, use differencing, seasonal differencing, and logarithm transformation to check stationary properties.

Task 2. Fit a simple moving average model (using the training set)

2.1 Apply the simple moving average model $s_t = \frac{1}{k} \sum_{i=t-k}^{i=t-1} x_i$ to the training data set, for a given k.

2.2 Calculate the error, i.e., the difference between the predicted and original value in the training data set, and compute the root mean squared error (RMSE) and Mean Absolute Percentage Error (MAPE) on slides 29/72.

2.3 Repeat the above two steps by varying k and calculate the RMSE and MAPE.

2.4 Plot RMSE and MAPE vs k, respectively. Select k based on the lowest RMSE or MAPE value. For the best value of k, plot the predicted values against the original values.

2.5 Comment on your results.

Task 3: Fit an exponential smoothing model

3.1 Apply the exponential smoothing model $\hat{x}_t = ax_{t-1} + (1-a)\hat{x}_{t-1}$ to the training data set for $a = 0.1$.

3.2 Calculate the error, i.e., the difference between the predicted and original value in the training data set, and compute the root mean squared error (RMSE).

3.3 Repeat steps 3.1 and 3.2 by increasing a each time by 0.1, until $a = 0.9$.

3.4 Plot $RMSE$ vs value a. Select value a based on the lowest RMSE value.

3.5 For the selected value of a plot the predicted values against the original values, and visually inspect the accuracy of the forecasting model.

3.6 Comment on your results.

Task 4: Fit an AR(p) Model (use the training set)

4.1 First select the order p of the AR model by plotting PACF in order to determine the lag k at which PACF cuts off, as discussed in Section 6.4.4.

4.2 Estimate the parameters of the AR(p) model. Provide RMSE value and a plot the predicted values against the original values.

4.3 Carry out a residual analysis to verify the validity of the model.

a. Do a Q-Q plot of the pdf of the residuals against $N(0, s^2)$. In addition, draw the residuals histogram and carry out a χ^2 test that it follows the normal distribution $N(0, s^2)$.

b. Do a scatter plot of the residuals to see if there are any correlation trends.

4.4 Comment on your results.

Task 5: Comparison of all the models (use the testing set)

Run the above three trained models on the test data. (That is, do not retrain the models, but simply run them as is on the testing set). Select the best model among the three, based on their performance on the testing data. Comment on your results.

What to Submit?

1. Submit a report with detailed procedures of your coding or the packages you use, or the AI tools that assist you for all three Tasks.

Sharing code is not allowed and constitutes cheating, in which case both students (the one that aids and the one that receives) will get a zero for the project and will be reported to the student conduct office.

2. Your results (graphs, snapshots, tables, etc) and your conclusions.

Grading:

- Task 1: 20 points.
- Task 2: 20 points.
- Task 3: 20 points.
- Task 4: 20 points.
- Task 5: 20 points.

Remember that you will be graded mostly on your ability to interpret the results