Capstone Project Summary

Implementation of New Model in Time Series Forecasting Dashboard

Cheng, Alec Jan Valenciano

122160

ACADEMIC YEAR | 2018 - 19
CAMPUS | Budapest
COURSE | MSc in Business Analytics
SUBMISSION DATE | 19 June 2019

Background

The client is one the largest operator of hotels, condotels, resorts, serviced suites, and branded residences in Indonesia. One of the KPIs that they follow closely is the **Occupancy Rate**, meaning what percentage of the rooms in a specific hotel are occupied. Forecasting this in an accurate manner could help in planning financials, logistics (personnel, facilities, etc.), and other things.

Goal

Currently, there are already existing models to do forecasting. The plan is to create a new model that may have a better performance. Specifically, I've been working on an implementation of **Recurrent Neural Network**. After creating the model, the next step is to integrate this to their existing shiny dashboard.

Existing Solution

The existing shiny dashboard contains 4 tabs ("Explanatory Analysis", "Model Builder", "Back Testing", and "Forecasting"). My part will mostly affect the last 3 tabs. **Model Builder** tab is where you create a model based on which algorithm you want (regression, ARIMA, ETS) and which predictors you would want to include**. Back Testing** shows plots of historical data compared to the predictions of the model you created / chose. Finally, **Forecasting** tab shows plots of historical data and forecasted values.

Tasks

I had work on managing the workflow of the shiny dashboard to incorporate the model that will be created. Here are the steps that were taken to accomplish the goal:

- Data preparation.
  - Create the function that will prepare the data suitable for the model to be created.
- Modelling
  - Create the function that will take in the data and output predictions.
  - Create also the functions for back testing and forecasting.
- Integration
  - Integrate all of these into the shiny dashboard.

Outcome

Implementing Recurrent Neural Network has proved to be useful as it generates more accurate predictions compared to existing models. In all hotels that were tested, with error function Root Mean Square Error being used, the new model always gets lower error values.

The only down side to the implementation of this is that generating the prediction using the RNN model can take more around 2 minutes. Using the other models, it is usually around a few seconds. We must take into consideration the balance between accuracy and the running time of the models. If you're going to run the model on several hotels, this would be a problem, but the use of this dashboard is not for running multiple hotels at the same time. Analysis is usually done by hotel so there would be no need to run it for multiple hotels at once. And with this, the wait time will not be as big of a hindrance.