QBUS2820 Assignment 2: Forecasting Solar Power Production from household Solar Panels

Overview

The assignment consists of forecasting the time series of solar power productions for 4 households in Australia, at different levels of temporal frequency (half-hourly, daily, monthly).

You will create and validate a methodology that considers the forecast models seen in the lectures, compute point, probabilistic forecasts and estimate the errors of the model.

Context and Data

Each time series measures a half-hourly frequency solar power production (KWh) from a solar panel situated in a rooftop of a house.

The dataset (solar_halfhourly.csv) comes in a csv format, with columns:

- `Customer`: Integer that identifies the house/solar panel.
- `datetime`: Date in datetime format.
- `value:` energy production, measured in KWh.

From this data, you would create several time series, for each Customer, and then forecast them all.

The dataset comes from Ausgrid, a subset from the original 300 clients that were made available. Data ranges from 2010 to 2013.

https://www.ausgrid.com.au/Industry/Our-Research/Data-to-share/Solar-home-electricity-data

Temporal Aggregation

The dataset comes measured at half-hourly frequency. As part of the assignment, you will forecast the half-hourly time series, but you will also generate time series at lower frequencies (Daily and Monthly) from the original series. This process is called 'temporal aggregation', where a group of observations are aggregated in some form (usually via the mean or the sum function). The groups are generated according to lower frequencies, for example, we could move from a half-hourly freq. Time series to a hourly time series by taking the two measurements for each hour in the day, and computing the mean of those two measurement to serve as the measurement for one hour.

In this assignment, the aggregation will be at the levels of Daily frequency (aggregating all observation in a natural day) and Monthly, aggregating all observations per calendar Month. Finer details such as daylight saving will be ignored.

A notebook that showcases temporal aggregation is available on canvas in the Assignment 2 module.

Analyzing a time series at different levels of temporal aggregation is typical in industry, and you might find that the best model at a given aggregation level might not be the best at another level. Interesting situations such as discrepancies between forecasts at different levels tend to happen. For example, the half-hourly forecast for the next day might not have a 'mean' that coincides with the forecast of the daily time series... What to do when that happens?

The forecast problem

The 'forecasts' that have to be computed and reported are:

- Point forecasts: You will forecast the next 48 half-hours for the half-hourly time series, the next 7 days for the daily time series, and the next month for the monthly time series. If you use a validation set and test set, pay attention that these have to be adjusted according to each level of temporal aggregation. You will do this for each time series, so there would be 12 time series (4 customers x 3 aggregation levels).
- Probabilistic forecasts: For each time series, forecast the 95% prediction intervals.
- Expected Performance: Estimate of the mean absolute error of the predictions for each time series. This is the estimation of what is going to be the prediction error on the test set, based on the training, before looking at the test set.
- Actual performance: The prediction error MAE calculated on the actual on a test set. Compare then actual vs expected performance. This implies creating a proper test set, using the horizon that depends on the aggregation level, for each time series.

The Comparison problem (Half-hourly vs Daily)

You will compare and discuss the performance of the best model at the half-hourly level for the next day (horizon=48) vs the performance of the daily forecast for the next day (horizon=1). The comparison can be in terms of prediction error, overpredictions vs underpredictions, etc. For example,

Based on that comparison and additional experiments, you will discuss if there is a way of using the half-hourly forecast to improve the quality of the daily forecasts, and vice-versa (using daily forecast to improve the quality of the half hourly). This is an open-ended question.

You need to do the comparison for **one** Customer of your choosing. (e.g. half-hourly vs daily time series from Customer 3) not all four Customers.

Comments and tips on the methodology

There are several time series in the dataset, and you do not need to forecast 'manually' by manually several models and eyeballing the best model. You can of course try whatever exploratory process you want to get a grasp of the data, but the objective is to create an **automatic methodology**, that given a time series, compares several models and chooses the best model for that time series according to some metric of 'best'. Then it computes the final forecasts using the best model. This likely implies some for of loop where you compare models, seasonalities, preprocessings/transformations, etc.

Once the methodology has been designed, then it can be applied to each time series in the dataset.

You can fine tune the methodology, for example by considering the frequency of a time series (special versions of the methodology for half-hourly, daily and monthly). The models that consider seasonality, the seasonality should be another parameter that is adjusted given the data, so finding a good seasonality should be part of the methodology.

Remember to document in writing your methodology, your design decisions: which models to compare, seasonalities to compare, adjustments and preprocessing (e.g. what to do with too long or too short series), necessary tradeoffs for computing speed if they become relevant, etc. Explain what are you doing and why.

Computing Time restrictions

You will aim towards the whole notebook taking less than 10 minutes to run. This might force you to make some sacrifices in the 'completeness' of the methodology (remember that these are always implicit, otherwise we would be trying every idea possible...). This is a soft restriction, but if the notebook deviates 'a lot' from 10 minutes of running time, it might get penalized.

What you need to submit

- A **notebook (.ipynb file)** that runs the methodology, creates the forecasts and documents the decisions and results along the way.
    - Divide it into sections and document clearly what you are doing using markdown cells before the code of each section. You can use one (or more) cells for the methodological discussion, this should be separated from the cells that clarify technical (programming parts). Failure to explain/Incomplete sections of the notebook might lead to strong penalties for those sections (this is, do not just have 'code').
    - The filename of the notebooks should be 'STUDENTID_ ASG2_nbook_QBUS2820.ipynb'
    - No pdf document is needed, make sure that the notebook is as clear as possible, you can find many examples online that interweave code and analysis.
- The notebook must have a **Forecast Results** section where you will report the forecast items (the four points in the forecast problem section of this document), using plots for the point and probabilistic forecast and text output for the performance. **You will also report on which model was used to point forecast each series**, to get a rough idea of the dynamics of the series (is it seasonal? Does it have a trend?, etc.).
- The notebook must have a **Half-hourly vs Daily Comparison** section, where you report on the performances for that problem (see the Comparison problem section above) and discuss potential ways of combining the information at these two levels to improve forecasts.

Marking

Percent of the total grade that is dedicated to each part of the assignment.

- Visual aspect of the notebook (10%). Proper sectioning, plots, text and code cells structure, etc.
- Methodology and Results: Point forecast (40%), Probabilistic(25%), Estimate the error (10%)
- Comparison Half-hourly vs Daily: 15%
- Other errors might penalize the final grade, e.g. Notebook does not run, the format of submission is not correct, etc.so percentages might vary because of this.