Capstone project #2 Report  
building energy forecast

Mike (Xiangnan) Shi

# Problem statement

## Project Background and Goal

Building energy consumption forecast is often of great interest to building owners for the purposes of tracking energy usage, revealing energy usage anomalies, forecasting operating cost and budget planning. When combined with utility rate structure, this can further guide building owners and management to operate the business in a way that minimizes expense and maximizes profit. However, there isn’t always a tool available for every building owner to forecast energy consumption. The scope of this project is to develop such a tool and evaluate its performance when predicting energy consumption.

## Project Overview

This project is conducted using Python in a Jupiter Notebook environment. The source code can be found at Github1. The dataset is from a Kaggle competition2.

The project follows a Data Science Method that consists of 6 steps as below:

1. Problem identification
2. Data wrangling
   1. Examine the data quality and fix missing values, outliers and wrong data
3. Exploratory data analysis (EDA)
   1. Explore the pattern of target data
   2. Explore the potential relationship between target data and relevant features
4. Pre-processing and Training Data Development
   1. Join the weather data and electricity data
   2. Define a holiday schedule and add it to the data as a feature
   3. Generate train and test data set
5. Modeling
   1. Build different models with both time series and random forest
   2. Tune the hyperparameters
   3. Run the models to forecast future energy usage based on test data features
   4. Evaluate the model performance
6. Documentation

1 Github link: <https://github.com/stonewatertx/Springboard_GitHub/tree/main/Capstone_Project_2_Building_Energy_Forecast>

2 Data link: https://www.kaggle.com/c/ashrae-energy-prediction/data

# data wrangling

## Data background

The dataset has three files:

* Building energy data. The data includes hourly metered data over a whole year timespan. There’re four types of energy consumption – electricity, natural gas, chilled water and steam. This project is only focused on electricity consumption. The original dataset from Kaggle has over 1400 buildings data with several million rows. To reduce the computing work, 1/3 of the buildings were randomly selected and used for the project.
* Weather data. It includes the hourly temperature, dew point temperature, precipitation, cloud coverage, etc. for each site.
* Building metadata. It shows the site id, building function, square footage and so forth for each building.

## Data cleaning

* **Zero readings**

Some electric meters show extended period of zero readings. This is considered wrong data because it’s almost impossible for a building to consume 0 electricity, especially for commercial buildings.

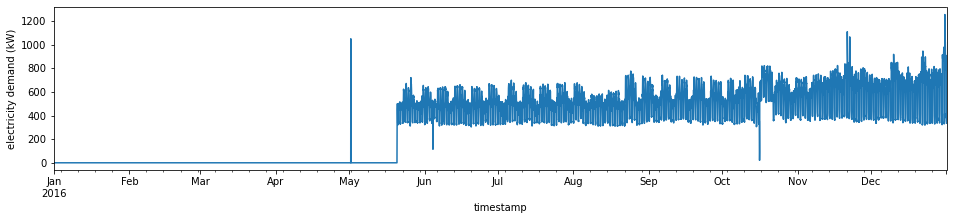


Figure 1. Zero readings example

It’s challenging to estimate the replacement for zero readings in this case because the missing period is long and there is no valid data during that period that can be used as reference for filling. As a result, those data points were dropped.

* **Frozen readings**

Frozen readings were also observed in the data as shown in below. They’re considered as sensor failures and dropped as well.

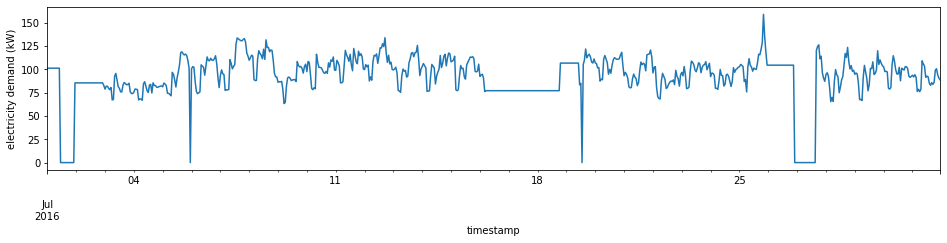


Figure 2. Frozen readings example

* **Anomalies**

Abnormal electricity swings were also observed in the data. It’s not clear whether it’s due to sensor issues or large building equipment dropped out of line. But regardless of the cause, it shouldn’t reflect normal operations. For our goal of forecasting energy consumption in normal operations, the outliers were dropped out. To do this, standard deviation was calculated and all data outside 3 times standard deviation range were considered outliers and dropped.

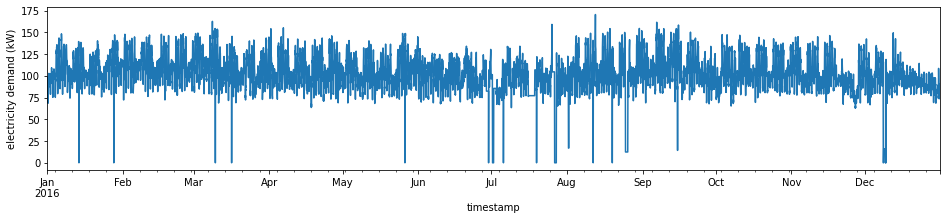


Figure 3. Outliers example

# exploratory data analysis

An Exploratory Data Analysis (EDA) was conducted to help us better understand the data, explore the relations between features and target value and see if there’s initial findings of pattern in the data.

## Monthly usage

In the chart below, electricity usage peaks in July and August. This is very common as the high temperature in summary often results in more air conditioning usage and therefore more energy consumption.

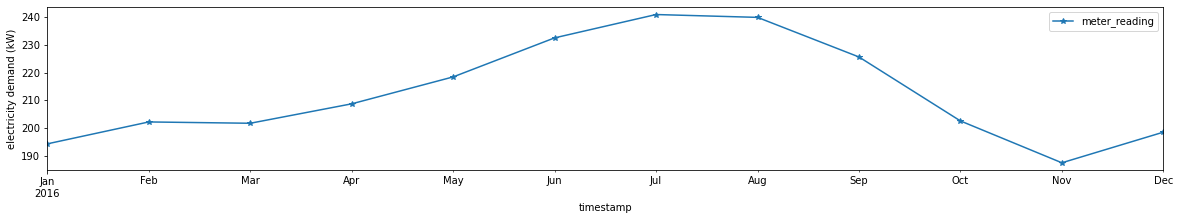


Figure 4. Monthly Electricity Consumption

## Impact of weather

To look closer into the relationship between electricity consumption and weather, two scatter plots were generated in figure 5. In those charts, daily average electricity consumption, air temperature and dew temperature were plotted. There is a clear pattern that once the temperature is above a threshold, the electricity consumption almost increases linearly. As result, the weather-related features will be considered as inputs in the modeling process.

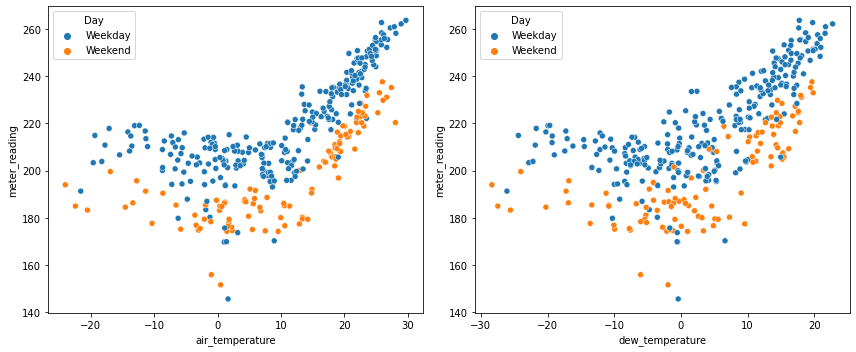


Figure 5. Electricity consumption vs air temperature and dew point temperature

## Seasonality

In order to investigate the patterns in daily average electricity demand, a decomposition analysis was conducted to investigate on the trend and seasonality. As shown in figure 6, there is a strong pattern of seasonality. This suggests that in the modelling session we should consider a model that incorporates seasonality.

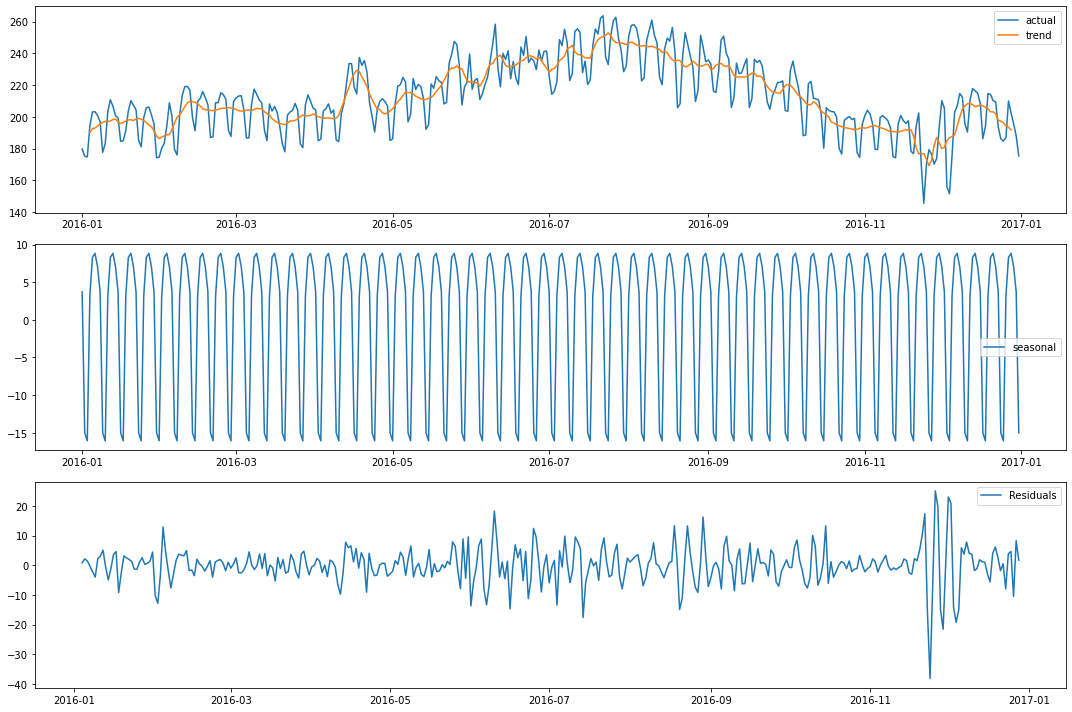


Figure 6. Daily average electricity demand decomposition analysis

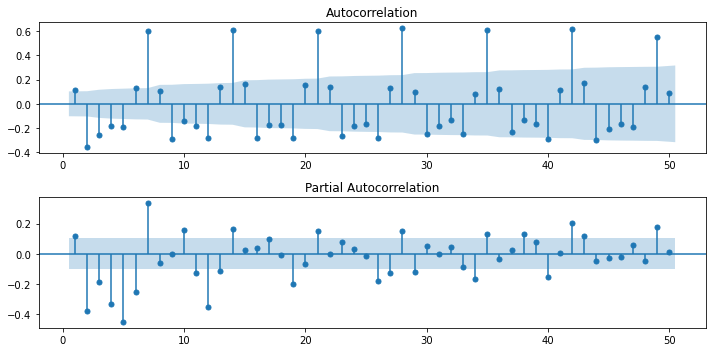
Another way to look into the seasonality is autocorrelation plot (ACF) and partial autocorrelation plot (PACF). Both plots clearly show that 7 days seem to be a cycle that electricity consumption follows. 

Figure 7. ACF and PACF plots

# modeling

Several time series models and one random forest model were explored in this project. The dataset was split into training set (Jan – Sep) and test set (Oct – Dec). Models were first built and fit based on trending set. Then the models make forecast for October to December and the forecast was compared to test set. Two metrics, R squared and Mean Absolute Error (MAE), were used to evaluate the performance.

## ARIMA Model

A basic ARIMA model was first tried out. This is the simplest model that has no seasonality. An auto arima function was called to find the best auto regression and moving average order. It turned out to be (2,1,5).

An arima model (2,1,5) was built and trained based the training data. Then a forecast method was called to forecast the energy usage during the test data time period. The forecast was compared to the test data. It shows significant error. The R squared was -10.9, which indicates the model performs even worse than simply using the average of targe value. The MAE is 22.73. The basic arima model is able to capture the weekly repetition, but the shape is a little bit off. Let's see if adding seasonality improves it.

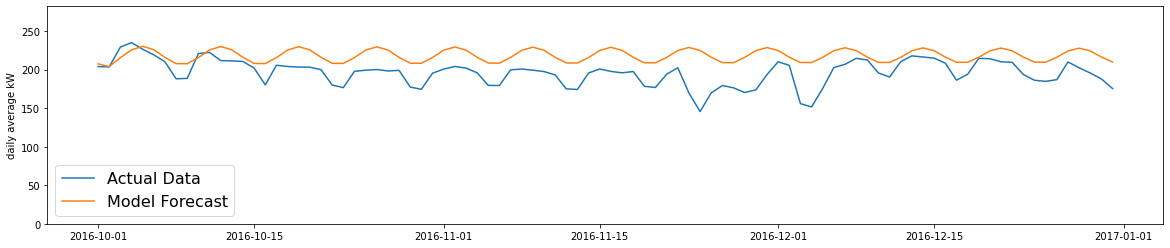


Figure 8. Basic ARIMA model forecast

## SARIMA Model

In this model, the seasonality component is added. Similar to the first model, auto arima is called to seek the best orders. It turns out to be ARIMA(5,1,0)(2,1,0)[7]. The model forecast result is shown in figure 9. R squared improved to -0.8, but it’s still not satisfactory. MAE is 25.47.

The shape of the forecast approximates the actual data much better now. However, the forecast shows a downward trend that seems to be wrong. This is because of the nature of SARIMA model that's based on previous few steps. If the previous data shows a trend, it will just pass on to the future forecast without correcting itself. To improve it, we'll try a rolling forecast.

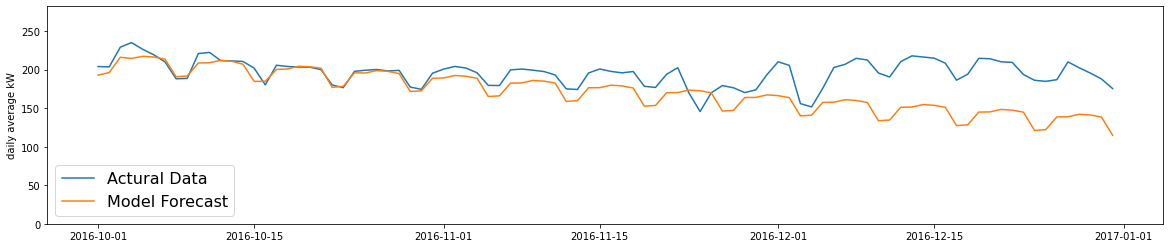


Figure 9. Basic SARIMA model forecast

## Rolling Forecast with SARIMA

In this model, we’re not doing a one-time forecast for three months ahead. Instead, we’re doing a rolling 1-week forecast. It means that we start with a one-week forecast. Once the week is past, we add the week to the training set, retrain and update the model and forecast the next week. This gets repeated over and over.

As shown in the figure 10, the forecast is able to match the test set much closer now. The downward trend from previous model was successfully corrected. R squared gets significantly improved to 0.25. MAE is 9.84

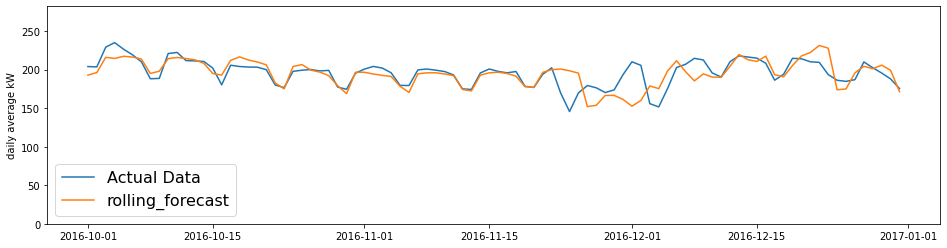


Figure 10. Rolling Forecast with basic SARIMA model

## Rolling Forecast with SARIMAX + Holiday and Weather as Exogenous Inputs

In this model we’re using all elements in a SARIMAX model, specifically the “X” component with holiday feature and weather added as exogenous inputs. The forecast gets further improved with R squared of 0.52 and MAE of 8.23.

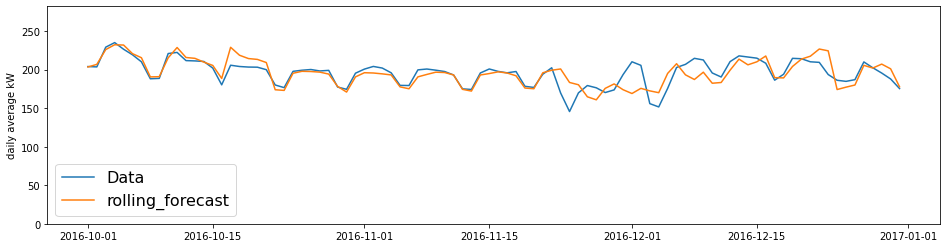


Figure 11. Rolling forecast with SARIMAX model with holiday and weather inputs added

## Random Forest

Above models are all based on time series. In comparison, a random forest model was also explored. The features in this model include holiday, weather (air temperature and dew point temperature) and day of week. With a grid search over a series of hyper parameters for number of trees, it turns out 12 trees seem to be the optimal number.

The R squared is 0.39 and MAE is 9.93. Though the result is not as good as previous model, but it has advantages of relative simplicity and capability of doing long term forecast.

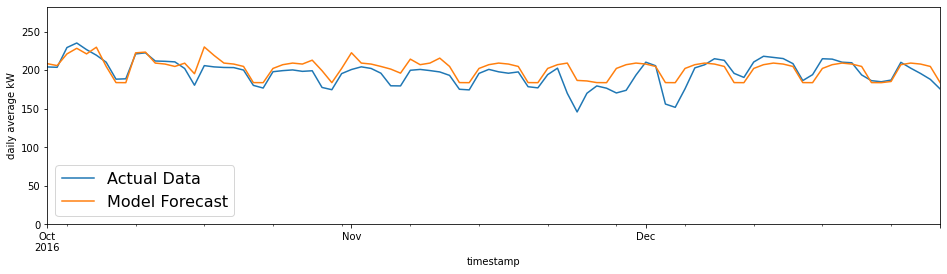


Figure 12. Forecast by random forest

## Performance summary

|  |  |  |
| --- | --- | --- |
| Model | R Squared | MAE |
| Basic ARIMA | -10.95 | 22.73 |
| SARIMA | -0.81 | 25.5 |
| SARIMA with rolling forecast | 0.25 | 9.84 |
| SARIMAX with rolling forecast and exogenous inputs | 0.52 | 8.23 |
| Random Forest | 0.39 | 9.93 |

# takeaways

* In this project, ARIMA and SARIMAX don't seem to perform well enough for a one-time long term forecast. The models tend to pick up a trend from prior steps and assume the trend will continue in the future. As a result, the forecast is very sensitive to the last few steps in the train data set.
* Time series models perform much better forecast if done in a rolling manner, which is more focused on short term forecast.
* Adding exogenous features, such as holiday schedule, weather data in this case can improve the accuracy of time series models.
* Random forecast can be an alternative solution to consider if a one-time long term type forecast is needed.

# future research

* Some of the data trend cannot be explained by the features used in the study. Additional features may be explored and integrated into the models. Those features may or may not be in the dataset we have.
* Additional models such as neural networks can be explored as well.
* In this study, our training is based on the first nine months of the year and is then used to forecast the last three months. There is a chance that some information in the last three months may be import inputs for forecast, but our training set don’t see that information and that limits the predicting power of our model. It might be worth using a whole year of data to train the model if the data is available.