5- Modelling

https://github.com/y-fawzy/energy-demand-forecast/blob/master/notebooks/4-modelling.ipynb

The data is almost ready for use in machine learning models. Before running any models, the following steps must be considered:

- Feature engineering can be used to intuitively add features that may be beneficial as the dataset has 4 features which will not be enough for a model this dynamic.
- Cross validate models where possible.
- Compare the best model of each model
- Dividing the training and testing set.
-

The dataset contains observation from **2015 to 2018**.

- **2015 to 2017** was used as the training set
- **2018** was used as the testing set.

RMSE will be used to compare the different models as this can tell us by how many Mega Watts our prediction is off by. To get a perspective of how this compares to the total, the metric: RMSE/Mean of Demand, will be used as a reference as well.

A) Feature Engineering

Set 1 features are weather related and have not been transformed from the dataset.

Set 1: Current Features (All aggregated weekly)

1. Maximum temperature
2. Minimum temperature
3. Mean temperature
4. Mean pressure
5. Mean humidity
6. Maximum windspeed

Feature sets that were added include the following:

Set 2 = Percentage Changes from Week to Week of all the weather features in set 1. Qty = 6

Set 3 = Datetime features. (Week of year, Month of year and Quarter of year). Qty = 3

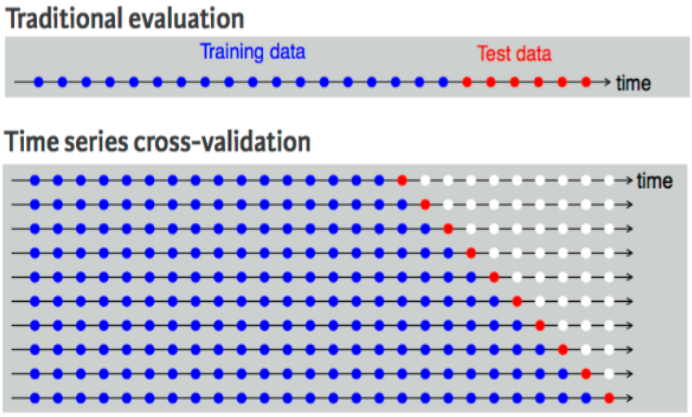
Set 4 = Difference of demand between current week and n-weeks ago. (1 & 2 weeks used). Qty =2

Set 5 = Difference of max, min and mean temperatures between current week and 1 week ago. Qty=3

Total features= 6 (Original) + 14 (Engineered)

B) Time Series Cross Validation Techniques

To cross validate a Time Series, the training set must be split using TimeSeriesSplit(n_splits=n).



Traditional splits will not work as each datapoint is dependant on the previous.

The figure visualizes the difference between TimeSeriesSplit() and Traditional Train-Test Splits.
There will be many combination of parameters to loop through. Thankfully there are tools that help such as

- Randomized Search CV. **Inputs**:
  - Estimator (Random Forest in this example)
  - Range of Parameters to randomly search through.
  - Cross validator (TimeSeriesSplit)
  - Number of iterations

- Grid Search CV: **Inputs**:
  - Estimator (Random Forest in this example)
  - Range of Parameters to randomly search through.
  - Cross validator (TimeSeriesSplit)

Randomized Search CV	Grid Search CV
Relatively Faster Computation	Slow/Expensive Computation
May not get the best parameters	Better to minimize loss function
Reasonable to accommodate larger range of values	A large range of values can cause memory problems

Most of this project utilized Randomized Search CV but Grid Search was used only once just for testing purposes as it takes much longer to fit a model.

C) Modelling

1) SARIMAX

1.1) Parameter Choice

- Parameters = (p,d,q)(P,D,Q,m)
- q/Q = Trend/Seasonal autoregressive Order
- d/D = Trend/Seasonal differencing order
- p/P = Trend/Seasonal Moving average order
- m= Seasonal cycle = 52 weeks
- All orders are integers

D is the significance of seasonality and was tested using the dicky fuller test. It returned a low p-value and negative test statistic which implies the differencing order can be set to 0.

The other parameters (q/Q/p/P) can range from 0 to infinity. Our limits will be set between 0 and 5 and the model that returns the lowest AIC score (Akaike information criterion).

Pyramid Arima has provided Auto Arima which runs through the different combinations of parameters and retrieves the best model. After the first model was run, features and residuals were analyzed using summary statistics. After running the test the best model with the lowest AIC retrieved was **(3,0,0)(1,0,0,52).**

1.2) Feature Choice

Part of SARIMAX summary statistics retrieves the table on the right. The column of interest is **P>|z|.** This signifies how important the features is in prediction. Features with values greater than 0.05 were removed from the analysis and the model was rerun.

The model with less features performed better on the training and testing set.

	coef	std err	z	P>|z|	[0.025	0.975]
intercept	-1625.5304	5.13e+04	-0.032	0.975	-1.02e+05	9.89e+04
x1	2.946e+04	3.85e+04	0.764	0.445	-4.61e+04	1.05e+05
x2	-4.177e+04	2.12e+04	-1.973	0.048	-8.33e+04	-279.658
x3	1465.8423	2.22e+04	0.066	0.947	-4.21e+04	4.5e+04
x4	5314.9267	900.701	5.901	0.000	3549.585	7080.269
x5	-4449.8159	6589.184	-0.675	0.499	-1.74e+04	8464.748
x6	1.641e+04	6698.624	2.450	0.014	3280.348	2.95e+04
x7	8.41e+05	3.55e+04	23.659	0.000	7.71e+05	9.11e+05
x8	-2.607e+05	1.22e+05	-2.145	0.032	-4.99e+05	-2.25e+04
x9	-2.158e+05	1.22e+05	-1.774	0.076	-4.54e+05	2.26e+04
x10	-1.243e+06	7356.881	-169.016	0.000	-1.26e+06	-1.23e+06
x11	9.19e+04	2.67e+05	0.344	0.731	-4.32e+05	6.16e+05
x12	-6.901e+04	3.29e+04	-2.100	0.036	-1.33e+05	-4601.616
x13	-972.2392	2256.845	-0.431	0.667	-5395.573	3451.095
x14	1.209e+04	3.28e+04	0.368	0.713	-5.23e+04	7.65e+04
x15	-4.466e+04	1.22e+05	-0.365	0.715	-2.84e+05	1.95e+05
x16	0.9922	0.125	7.930	0.000	0.747	1.237
x17	-0.2824	0.066	-4.247	0.000	-0.413	-0.152
x18	-3.881e+04	2.25e+04	-1.725	0.084	-8.29e+04	5274.291
x19	1.491e+04	1.14e+04	1.311	0.190	-7386.614	3.72e+04
x20	1831.8145	1.34e+04	0.137	0.891	-2.44e+04	2.81e+04
x21	1.397e+04	1.48e+04	0.943	0.346	-1.51e+04	4.3e+04

1.3) Residual Analysis

The following are the results available to analyze residuals.

1. Prob(Q) & Prob(JB)

Prob(Q):	0.46	Prob(JB):	0.06

Prob(Q) = p-value for the null hypothesis that residuals are uncorrelated. If greater than 0.05 then the residuals uncorrelated. The best model has P(Q) = 0.46 which is good.

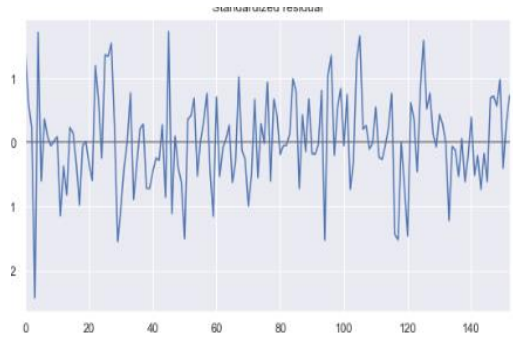
Prob (JB) = p-value for the null hypothesis that residuals are normally distributed. If greater than 0.05, the residuals are indeed normally distributed. The best model has P(JB) = 0.06

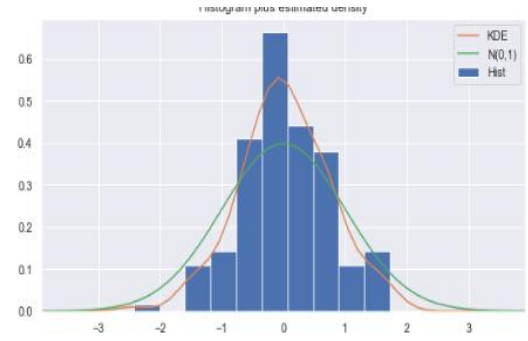
2. Residual Plots

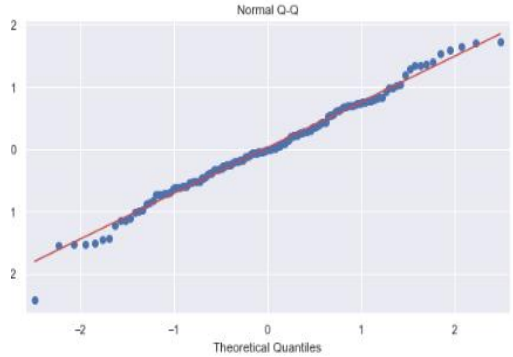
Residuals over time showing no pattern which is good

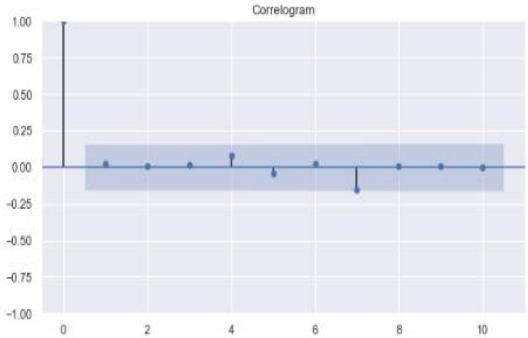
Residual distribution(Orange) vs normal distribution (Green).
- Mean is at 0
- Std is narrower than normal

This is acceptable







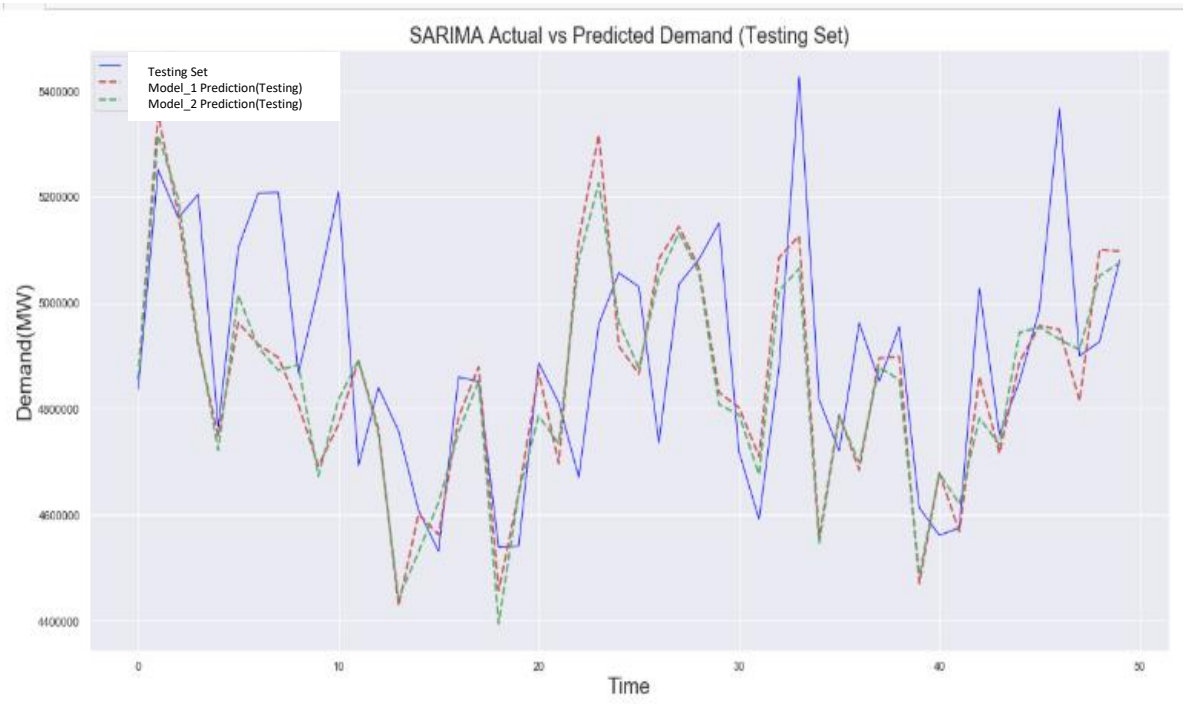


Residuals lining up well with normal distribution line

Residuals do not show significance with each other so they are uncorrelated
This is good

Residuals showing no reasons for concerns about prediction. Parameter choice looks like the best choice with current dataset and computation.

1.4) Results



The graph above shows the plot of the *testing set(Blue line*), the initial model with all features *Model_1(Dotted Red line*) and *the model with reduced features, Model_2*
Model_2 returned the better results on the training and testing sets. Here are the results on the training and testing sets.

Training Set

The RMSE is 98,000 MW.

$$\frac{RMSE \times 100}{Mean\ of\ Training\ Set} = \mathbf{2.04\%}$$

Testing Set

RMSE = 196,000 MW

$$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set} = \mathbf{3.99\%}$$

Cross validation was not be performed on this SARIMA model as parameter choice was not complex.

2) Random Forest Regression

2.1) Parameter Tuning

- n_estimators = Number of trees use in the model
- max_depth = Max depth of each decision tree used
- max_features = Max number of features used to decide each split

Randomized Search CV was run at 3000 iterations and the below results were found.

RANDOMIZED SEARCH CV	
n_estimators	Range(100,800) . Interval = 50
max_depth	Range(1,150). Interval = 1
max_features	Range(2,15). Interval = 1

The above ranges were used for Randomized Search CV. Grid Search CV could not handle running these ranges so they were reduced to the below table.

GRID SEARCH CV	
n_estimators	Range(100,500) . Interval = 50
max_depth	Range(1, 50). Interval = 2
max_features	Range(2,12). Interval = 1

2.2) Results

i) *Best Parameters*

n_estimators	100
max_depth	9
max_features	4

ii) *Errors*

The following results are those on the testing set. Both models achieved the same parameters. Even though Grid Search took less time to run, less range of values were tested due to memory issues.

Randomized Search CV (Iterations=3000)	**Grid Search CV**
Run Time = **1 hour 53 minutes**	Run Time = **50 minutes**

RMSE = **171,600 MW**	RMSE =**171,600 MW**
$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set} = \mathbf{3.5\%}$	$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set} = \mathbf{3.5\%}$

3) Gradient Boost Regression

3.1) Parameter Tuning

n_estimators = Number of trees use in the model
max_depth = Max depth of each decision tree used
max_features = Max number of features used to decide each split
subsample = Fraction of training data to be randomly samples for each tree.
learning_rate = Controls weight of new trees added to the model based on errors

n_estimators	Range(100,800) . Interval = 50
max_depth	Range(1,150). Interval = 1
max_features	Range(2,15). Interval = 1
subsample	Range(0.001,0.99). Number of points = 50
learning_rate	Range(0.001,1). Number of points = 30

Randomized Search CV was run at 3000 iterations and the below results were found.

3.2) Results

i) Best Parameters

n_estimators	600
max_depth	52
max_features	3
subsample	0.057
learning_rate	0.03

ii) Errors

Training - Iterations=3000	**Testing - Iterations=3000**
RMSE = **99,300 MW**	RMSE = **154,800 MW**
$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set} = \mathbf{2.07\%}$	$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set} = \mathbf{3.16\%}$

4) XG Boost Regression

4.1) Parameter Tuning

n_estimators = Number of trees use in the model
max_depth = Max depth of each decision tree used
max_features = Max number of features used to decide each split
subsample = Fraction of training data to be randomly samples for each tree.
learning_rate = Controls weight of new trees added to the model based on errors
colsample_bynode = subsample ratio of columns for each split

n_estimators	Range(100,800) . Interval = 50
max_depth	Range(1,150). Interval = 1
max_features	Range(2,15). Interval = 1
subsample	Range(0.001,0.99). Number of points = 50
learning_rate	Range(0.001,1). Number of points = 30
colsample_bynode	Range(0, 1). Interval = 0.05

Randomized Search CV was run at 3000 iterations and the below results were found.

4.2) Results

i) Best Parameters

n_estimators	620
max_depth	69
max_features	9
subsample	0.119
learning_rate	0.09
colsample_bynode	0.65

ii) Errors

Training - Randomized Search CV (Iterations=3000)	**Testing - Randomized Search CV (Iterations=3000)**
RMSE = **23,100 MW**	RMSE = **170,000 MW**
$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set}$ **= 0.48%**	$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set}$ **= 3.47 %**

D) Best Model Found

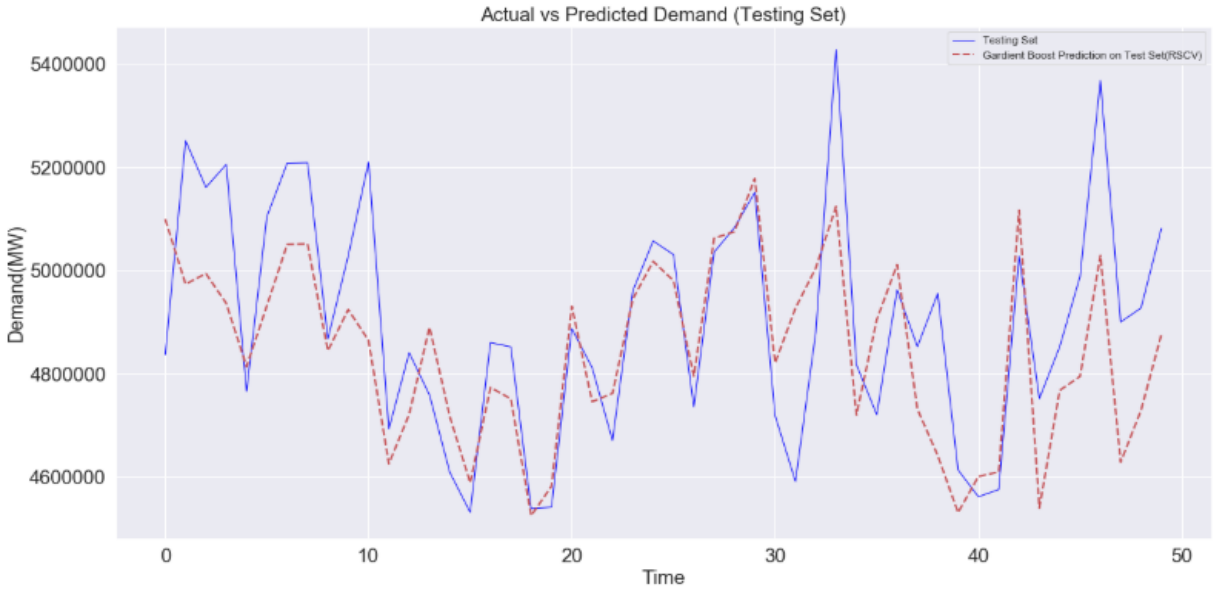
For best results, the best model out of all was remodelled and ran at 10,000 iteration(3.3x original iterations). The results are shown below.

Randomized Search CV (Iterations=10,000)
RMSE = 146,529 MW
$\frac{RMSE \times 100}{Mean\ of\ Testing\ Set} = 2.99\%$

n_estimators	775
max_depth	147
max_features	14
subsample	0.28
learning_rate	0.069

Here is a summary of all the models results achieved.

Model Name	**RMSE Train (MW)**	**RMSE/Mean Train (%)**	**RMSE Test (MW)**	**RMSE/Mean Test (%)**
SARIMAX	97,842	2.04	195,663	3.99
RFR	76,179	1.59	171,575	3.5
Gradient Boost	99,304	2.07	154,831	3.16
XG Boost	23,096	0.48	169,981	3.47
GB–10,000 iter	46	0	146,529	2.99



The above graph represents the best model achieved which is the Gradient Boost model. The pattern is being captured quite well but the magnitudes are off.

This is not a model that is ready for implementation. Limitations and improvements will be discussed in the next section.

E) Conclusion and Limitations

- The model is predicting total weekly demand, which is around 5,000,000MW.
- The best RMSE on the testing set found was 146,529 MW. This is 2.99% of mean.
- The graph shows capturing of the fluctuations well. The magnitude of the values predicted sometimes falls short of the actual value which is the reason for the error.

This error can be due to the following:

- The training set only contained data for 3 years. That is only 3 seasonal cycles. Given more data, the results will improve.

- Computational power is not strong to tune a larger number of model parameters as well as a wider range of parameter range.