Energy Demand Forecasting Report

Group 7 – Nicolas Argyrides, Eaton D'Souza, Tharun Srikant, Will Flynn

Introduction

Energy demand forecasts play an important role in power production and distribution planning in the energy sector. There are a number of different techniques available, such as exponential smoothing, ARIMA, linear regression, etc. As part of a forecasting competition, the group has built a few models and selected one to generate point forecasts for specific dates in the future. The purpose of this report is to discuss the methodology in creating the team's forecasts and discuss the advantages and disadvantages of each model. Furthermore, the report will elaborate on the reasoning behind each choice of model in every stage of the forecasting competition.

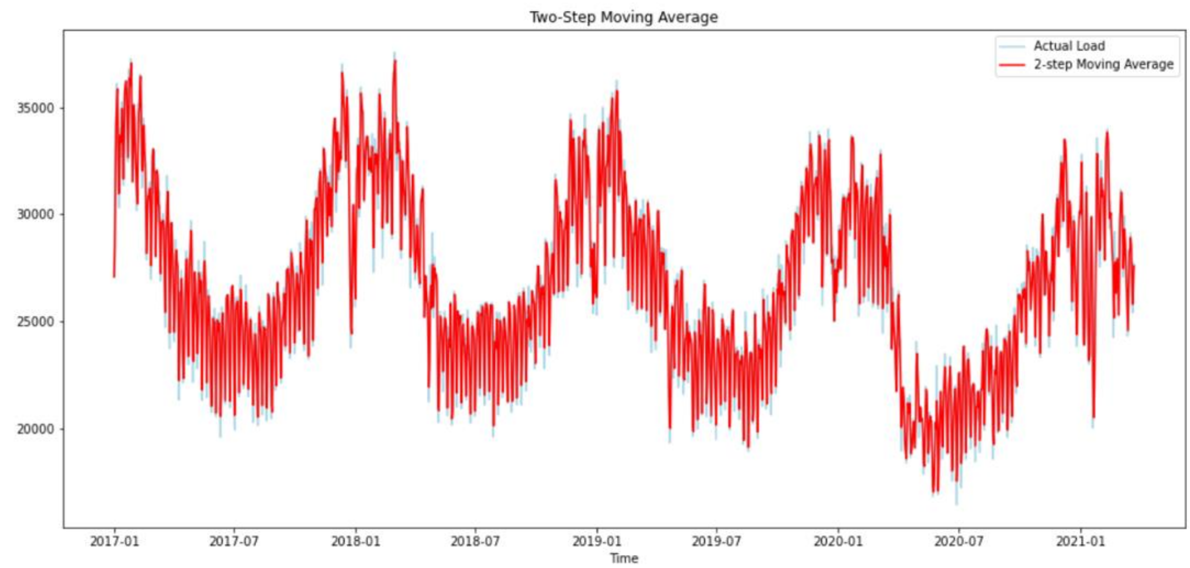
Forecasting Models

In this section, all six forecasting models will be discussed with an emphasis given to how the models were constructed and their performance during forecasting period. The models use a combination of both demand and weather data to forecast the electricity demand in the UK on a specific day. The accuracy of the models will be evaluated by the root-mean-squared error values.

Moving Average

The simplest forecasting method is the Naïve forecasting method which assumes the energy demand at time t is the same as t-1 i.e. the demand is the same as the day before. However, as the competition requires forecasting a day ahead without knowing the current day's forecast, this method was not used. The moving average method takes the average of a number of days and uses this to forecast the demand for the following day. Initial exploratory data analysis found that there are both strong seasonality and weekly trends, therefore simply using the average annual demand as a forecast would lead to significant inaccuracy. The advantage of using the 'moving average' model is that it finds the average load of the most recent days. As later discussed, one key finding in our analysis was that the demand is closely related to the demand of the previous day. Hence a 2-day moving average model was built using the equation below; the algorithm forecasts the next day's demand using the average demand of the two previous days.

$$Demand_t = \frac{1}{2}\sum_{i=1}^{2} Demand_{t-i}$$



A plot of the actual daily load dating back to 2017 has been plotted, along with the forecasted load using the 2-day moving average. As shown by the strong overlapping of the actual load (light blue) by the forecasted load (red), the 2-day moving average does predict the daily load fairly well. The findings suggest that demand is closely related to the average of the previous two-day demand.
By subtracting the forecast from the actual load, the prediction error was found and thus the RMSE (Root Mean Squared Error) of the model calculated. The 2-day moving average model had an approximated RMSE value of **2376.36** MWh.

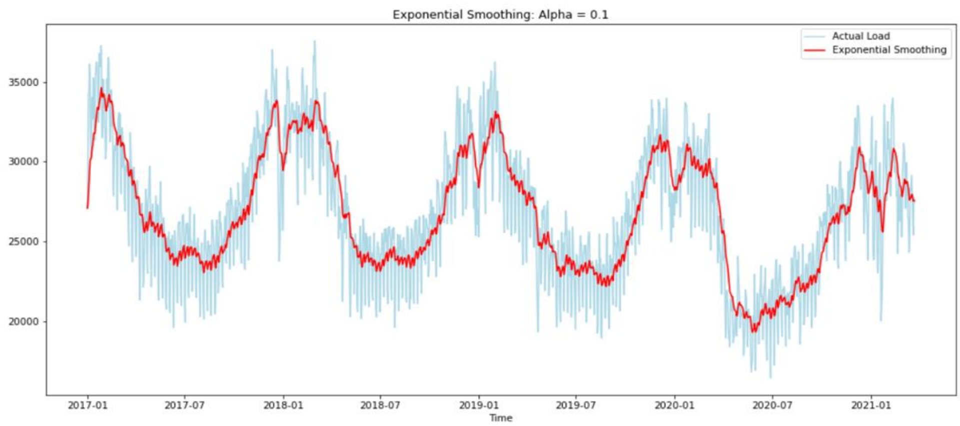
Single Exponential Smoothing

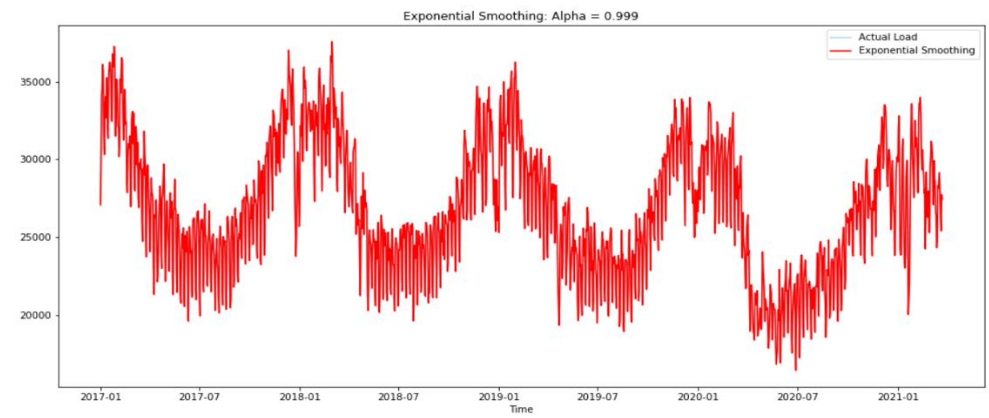
A commonly used forecasting method is exponential smoothing which estimates the current smoothed level using a weighted average of the previous estimate and a new point value. The equation below was used to calculate the smoothing constant, where 'a' (alpha) is the smoothing constant and S_t is the smoothing level at time t. The prediction for the next day's demand will simply be the current value of S_t.

$$S_t = ax_t + (1 - a)S_{t-1}$$

An iterative procedure was carried out where values of alpha were increased in step intervals of 0.001 from 0 to 1, and forecast values for the entire data set were calculated and the sum of errors between forecast and actual load recorded. This was carried out to find the optimal alpha value. A graph of the actual load values has been plotted against the forecasted values using the single exponential smoothing method, using alpha values of 0.001 and 0.999.

As shown by the two graphs, the optimal alpha value was found to be 0.999, which is the equivalent of using a small value of n in the moving average algorithm. The key finding is that the more weight is given to recent data as the previous days demand is closely related to the next day's demand. The (estimated) RMSE value for the simple exponential method was found to be **2066.24** MWh.



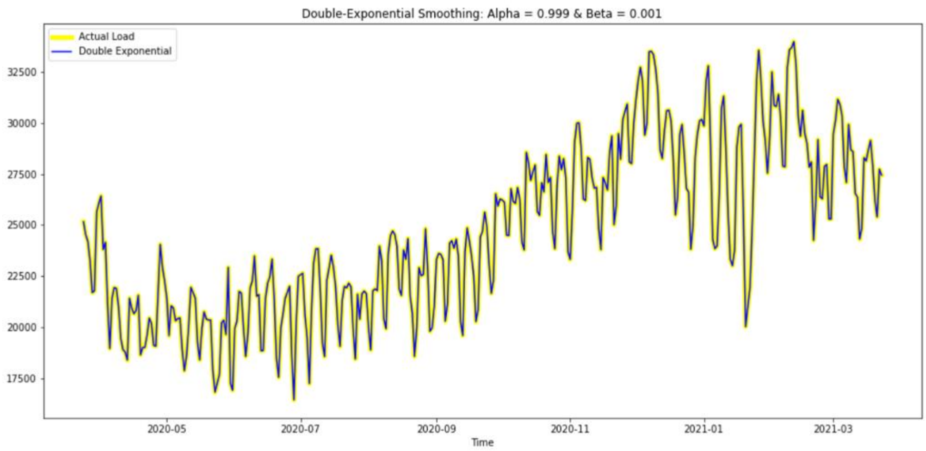


Double Exponential

Exponential smoothing tends not to perform as well when there is an underlaying trend in the data, as observed in our dataset. A solution is to control for the underlaying average (S_t) and the slope of the trend (δ_t). The equations below were used to calculate the smoothing level and the slope of the trend. The smoothing level is given by 'a' alpha and the trend smoothing constant 'B' beta. The forecasted load for n periods ahead is abbreviated as $F_{t+n} = S_t + n(\delta_t)$.

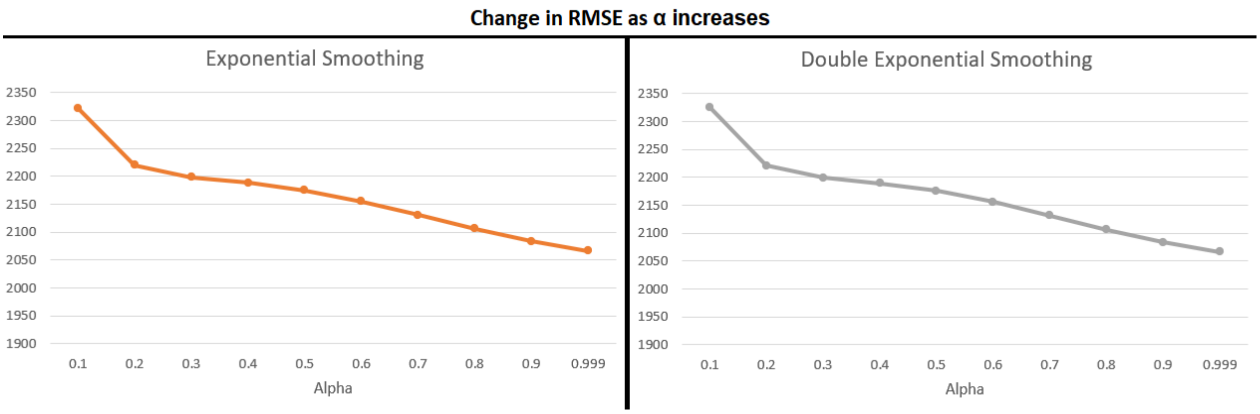
$$S_t = ax_t + (1-a)(S_{t-1} + \delta_{t-1})$$
$$\delta_t = \beta(S_t - S_{t-1}) + (1-\beta)\delta_{t-1}$$

Similar to the steps carried out for exponential smoothing, an iterative procedure was carried out to determine the optimal values of alpha and beta to create the best model. First the value of alpha was increased from 0 to 1 in steps of 0.001, beta was set to 0.001 and the sum of forecast errors recorded. The value of beta was then increased in steps of 0.001 to 1, and the process repeated before the next step increase of alpha. This procedure found the optimal values of alpha to be 0.999 and beta to be 0.001. The results suggest that the influence of the slope must be minimal as we observe strong fluctuations in demand which are not ideal.



A plot of the forecasted loads using the double exponential algorithm and actual demand load has been created for the data spanning back 365 days. The forecast uses a beta value of 0.001 and an alpha value of 0.999. One observes the influence of the weekend is quite significant, thus affecting the optimal weighting of the beta term. Once more, the strong overlap suggests the forecast matches closely to the actual demand. Calculation of the RMSE value for the double exponential method was found to be **2066.25 MWh**, a lower error than both the exponential and moving average methods.

The results of the iterative procedure were used to plot the model RMSE value for varying values of alpha. The two models compared were the simple exponential with varying alpha and double exponential with constant optimal beta value (0.001) and varying alpha values. As shown by the plots below, the simple exponential smoothing produces a more accurate forecast in general. However, as the alpha value continues to increase the difference in error between both models decreases. At the maximum value of 0.999, the difference in RMSE value between both models is negligible. Hence one can predict both the simple and double exponential models to have a similar performance during the forecasting competition.

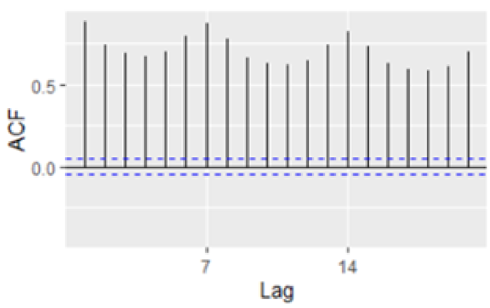


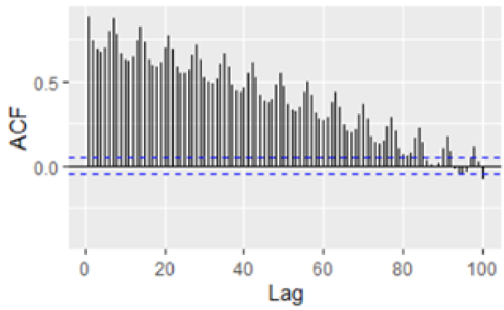
Auto Regressive Induced Moving Average (ARIMA)
One of the models that we have been using throughout the forecasting competition to forecast the load was the ARIMA model. Instead of describing the trend, we have used the ARIMA model as an approach to detect and describe the autocorrelations in the data. This implies that for this specific method we did not use any weather information linked to the load as the ARIMA model makes use only of a timeseries which consists of the historical load data.

Stationarity - When it comes to stationarity it is important to make sure that the properties of the time series are depended on the time. We have been expecting that the timeseries is going have seasonality and possibly, a trend implying that it is not stationary. This could adversely impact our forecasts and hence, we conduct tests with the aim of detecting stationarity.

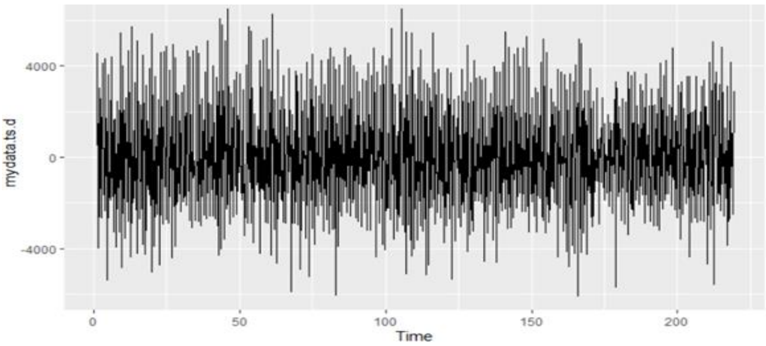
```
        Augmented Dickey-Fuller Test

data:  mydata.ts
Dickey-Fuller = -3.1198, Lag order = 11, p-value = 0.1043
alternative hypothesis: stationary

p-value smaller than printed p-value
        Phillips-Perron Unit Root Test

data:  mydata.ts
Dickey-Fuller Z(alpha) = -124.08, Truncation lag parameter = 7, p-value = 0.01
alternative hypothesis: stationary

p-value smaller than printed p-value
        KPSS Test for Level Stationarity

data:  mydata.ts
KPSS Level = 1.578, Truncation lag parameter = 7, p-value = 0.01
```

To test for stationarity, we used the Augmented Dickey-Fuller Test, Phillips-Perron Unit Root Test and the KPSS Test for Level Stationarity. Given the results in the figure above, there is an indication that according to the Augmented Dickey-Fuller Test that the timeseries is non-stationary while the Phillips-Perron Unit Root Test suggests otherwise. Given that the result of these two tests is not consistent, a KPSS Test was carried out which in contrast to the other tests the null hypothesis indicates that the timeseries is stationary. Thus, we concluded that the time series is non-stationary and differencing could allow us to make the data stationary prior using it to create our load forecasts. This is because differencing will allow us to reduce the trend and seasonality around the data and preserving a constant mean that is stationary. Another way to identify non-stationarity is by plotting the Auto Correlation Function and Partial Autocorrelation Function Plots:



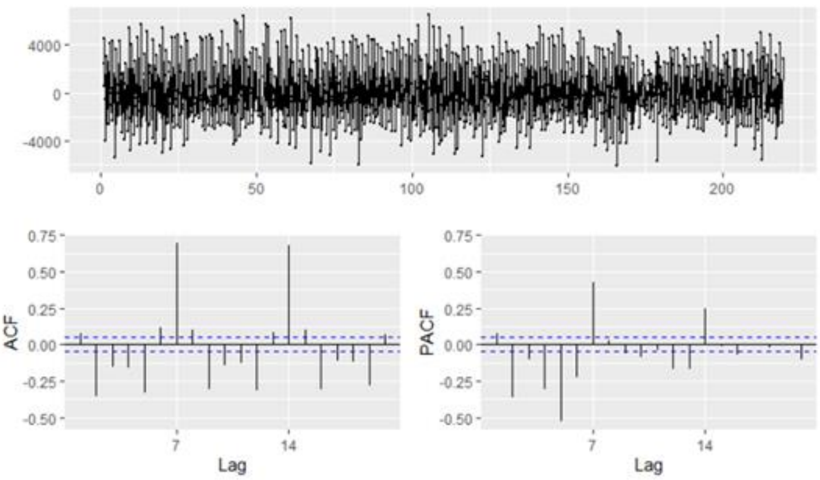


Given that the ACF values in the first figure with 20 lags are not converting to zero is another indication that the timeseries could be non-stationary and thus we plotted the 100 lags ACF where we can see that eventually the values converge to zero. Given the stationarity tests and ACF plots we have used an R build in function that allows us to detect the number of differences we need to take in order to make the timeseries stationary.



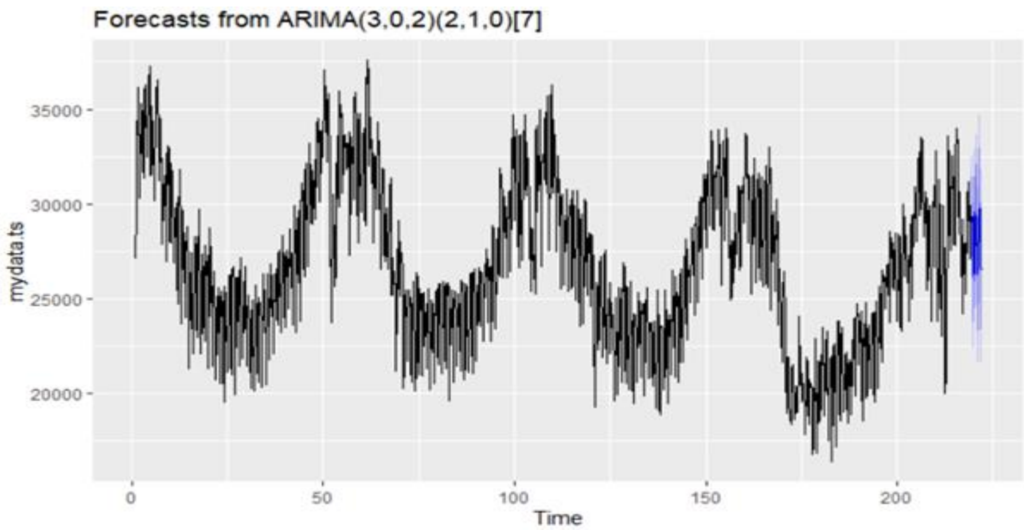
It is suggested that we take the first order difference and consequently after carrying out the same stationarity tests, this time all 3 tests conclude that the timeseries now is stationary. The figure (left) shows the data with a first order difference. Given that the values converge to a mean around 0 and that the variance is fairly constant overtime, we can confirm that the time series has been differenced and it is thus, stationary.

Arima Model Specification - To further investigate the models' specifications, we used the differenced timeseries to see the lags that are correlated and thus find the AR and MA components.



The figure above shows significant spikes at lags 2 and 5 for both the ACF and PACF, while the PACF has also a significant spike at lag 3. This is an indication that the non-seasonal components of MA(2) and AR(3). When it comes to the seasonal components, we can see that at lags of multiples of 7 we see significant spikes indicating that we have seasonal components of MA(1) and AR(1). We have tried various combinations of models and used an R build in function that allows us to iterate between various ARIMA models and detect the best model to use based on the model that has the lowest AIC value. Given that we have used the model in various occasions during the forecasting competition and updated the data each time, the best model most of the times appears to be ARIMA(3,0,2)(2,1,0)[7]. This indicates that our model has non-seasonal components of AR(3) and MA(2) as well as seasonal components of AR(1) and MA(0) and a seasonal difference.

After deploying this model, we made sure to check the residuals as an additional indication whether our model is fully robust and could be used for forecasting future load. We concluded that we are going to monitor how the ARIMA model performs throughout the forecasting competition although the prediction intervals may not be very accurate given some residuals appear to be correlated. For this reason, we chose not to report any forecasts with the ARIMA model but use it as an indication to compare the forecasts with the rest of the models used as well as the actual load once that becomes available.



One limitation that the ARIMA model has is that it only considers historical data and finds the characteristics that are most likely to be replicated in the future and does not accounts for drastic changes in consumer behaviour or weather conditions. As discussed earlier we believe that weather conditions and the current Covid-19 pandemic have most certainly changed the behaviour and habits of people thus drastically affecting the current and future load. This could imply that data from 2017 is not appropriate to use given a critical change in the load data has been occurred over the past year. Perhaps multiple linear regression or double exponential smoothing models can be better predictors of future load. The RMSE value using the ARIMA model was found to be **1174.935** MWh**.**

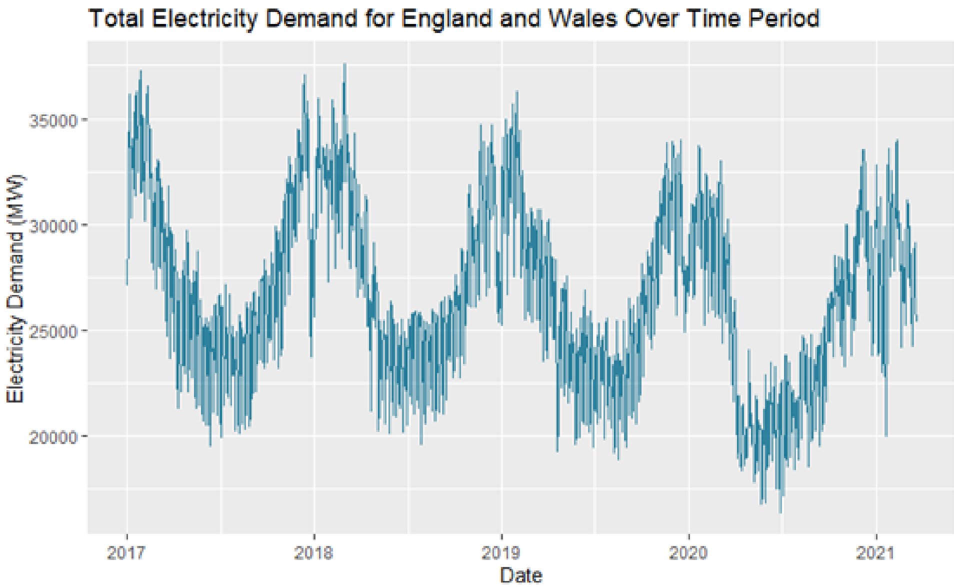
Linear Regression Model

As a novice approach, our first choice was to use a linear regression model to forecast demand. To create the regression, we need to determine which independent variables should be included in the model. Observing the data, we are given the temperature measured in 6-hour intervals for London, Leeds, and Bristol. The daily temperature for each city could be modelled in two ways; by computing the daily mean or by calculating the difference between the maximum and minimum temperature recorded for that day.

To determine which measure to use, the correlations are calculated between demand, mean and difference for each city. Demand is more heavily influenced by the average temperature with a correlation of 0.705 as opposed to the difference, 0.464. Therefore, the daily average for each city was chosen as it has a greater effect on the load. Furthermore, 3 variables to model the effect between the amount of sunlight each city receives, and demand are added to the regression. The linear regression as of now can be written as:

$$\begin{aligned}\text{demand}_t \;=\; & \beta_0 + \beta_1 \text{mean temp. London} + \beta_2 \text{mean temp. Leeds} + \beta_3 \text{mean temp. Bristol} \\ & + \beta_4 \text{sun London} + \beta_5 \text{sun Leeds} \;+\; \beta_6 \text{sun Bristol} \;+\; \epsilon_t\end{aligned}$$

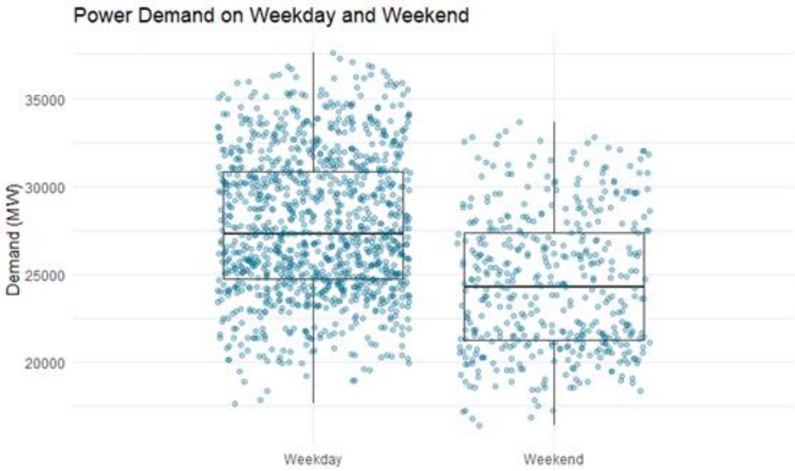
To observe seasonal fluctuations, electricity demand is plotted over the time frame:



The graph above shows that demand follows a seasonal pattern where the load is lower during the summer and high during the winter. The pattern seems logical as more electricity is used for heating during lower temperatures in December, January, and February. Less electricity is used during the summer months as temperature rises and air conditioning is rare in the UK. The general trend in demand reduces slightly from winter in 2020. Due to seasonal fluctuations, a dummy variable for each season is included in the regression, taking a value of 1 if the forecasted date is within that season, and 0 if otherwise:

$$\begin{aligned}\text{demand}_t \;=\; & \beta_0 + \beta_1 \text{mean temp. London} + \beta_2 \text{mean temp. Leeds} + \beta_3 \text{mean temp. Bristol} \\ & + \beta_4 \text{sun London} + \beta_5 \text{sun Leeds} \;+\; \beta_6 \text{sun Bristol} + \beta_7 \text{winter} + \beta_8 \text{spring} + \beta_9 \text{fall} \\ & +\; \epsilon_t\end{aligned}$$

Summer is the base group for the season dummy variables and takes the value of the intercept. Furthermore, detailed analysis indicates that there are differences in demand during weekdays and weekends.



The box plots indicate that weekdays have a higher average demand around 27,500 MW compared to a lower demand of around 25,000 MW on the weekends. The difference is expected as many shops, companies, factories and so forth close during the weekends. Therefore, a dummy variable is added to the linear model which takes the value of 1 if the forecasted day is a weekend, and 0 if otherwise.

A lagged dependent variable of one day is included in the regression model. It is highly likely that the electricity demand does not differ significantly between 2 consecutive days. Due to the seasonal effect, and the slight changes in weather between any 2 days, demand should remain similar. Therefore, the previous day's demand can help us predict the demand for the following day. The final regression can be written as:

$$\begin{aligned}\text{demand}_t = \; & \beta_0 + \beta_1 \text{demand}_{t-1} + \beta_2 \text{mean temp. London} + \beta_3 \text{mean temp. Leeds} \\ & + \beta_4 \text{mean temp. Bristol} + \beta_5 \text{sun London} + \beta_6 \text{sun Leeds} + \beta_7 \text{sun Bristol} \\ & + \beta_8 \text{winter} + \beta_9 \text{spring} + \beta_{10} \text{fall} + \beta_{11} \text{weekend} + \epsilon_t\end{aligned}$$

To assure robustness of the results, moving averages of 7-day intervals are calculated for demand and its autoregressive term, along with the mean temperature and sunlight variables for each city. This is because we make predictions given estimated weather forecasts for the following day. However, demand data for the previous two days prior to the forecasted day (t-2) is unavailable. Therefore, using moving averages we can use the demand value from two days ago in our forecasts for the following day. The sharp drop in demand on the weekend will affect prediction on Mondays negatively, because of this we can expect a stronger correlation of demand with a seven-day average rather than a one-day lag. Finally, a moving average is calculated for the autoregressive term as it is simply the moving average of demand(t-1). An AIC test is performed on the linear regression model to determine the best selection of significant variables to include in the regression. The purpose of using the AIC test was increase model complexity to improve the accuracy of the model whilst preventing overfitting. The model with the lowest AIC score will be used to forecast demand. The AIC test is run each time the data is updated, although the best model outcome is unlikely to change much over the month.

AIC Score	Regression
25317.23	$\text{demand}_t = 1$
17290.07	$\text{demand}_t = \beta_0 + \beta_1\text{demand}_{t\text{-}1} + \beta_2\text{mean temp. London} + \beta_3\text{mean temp. Leeds} + \beta_4\text{mean temp. Bristol} + \beta_5\text{sun London} + \beta_6\text{sun Leeds} + \beta_7\text{sun Bristol} + \beta_8\text{winter} + \beta_9\text{spring} + \beta_{10}\text{fall} + \beta_{11}\text{weekend} + \epsilon_t$
17288.09	$\text{demand}_t = \beta_0 + \beta_1\text{demand}_{t\text{-}1} + \beta_2\text{mean temp. London} + \beta_3\text{mean temp. Leeds} + \beta_4\text{mean temp. Bristol} + \beta_5\text{sun London} + \beta_6\text{sun Leeds} + \beta_7\text{sun Bristol} + \beta_8\text{winter} + \beta_9\text{spring} + \beta_{10}\text{fall} + \epsilon_t$
17286.11	$\text{demand}_t = \beta_0 + \beta_1\text{demand}_{t\text{-}1} + \beta_2\text{mean temp. London} + \beta_3\text{mean temp. Leeds} + \beta_4\text{mean temp. Bristol} + \beta_5\text{sun London} + \beta_6\text{sun Leeds} + \beta_7\text{sun Bristol} + \beta_8\text{winter} + \beta_9\text{spring} + \epsilon_t$
17284.96	$\text{demand}_t = \beta_0 + \beta_1\text{demand}_{t\text{-}1} + \beta_2\text{mean temp. London} + \beta_3\text{mean temp. Leeds} + \beta_4\text{sun London} + \beta_5\text{sun Leeds} + \beta_6\text{sun Bristol} + \beta_7\text{winter} + \beta_8\text{spring} + \epsilon_t$

The results of the AIC test are shown above, with the various models and their respective scores. The lowest AIC score was found to be 17284.96 for the best model. The final linear regression model is shown below. Surprisingly, the 'weekend' variable was not included in the model. However it is important to note that we did not take the previous days load as a variable rather, a 7-day moving average was chosen. By doing so, the 'weekend effect' is accounted for by the moving average with respect to the actual demand.

$$\text{demand}_t = \beta_0 + \beta_1\text{demand}_{t\text{-}1} + \beta_2\text{mean temp. London} + \beta_3\text{mean temp. Leeds} + \beta_4\text{sun London} + \beta_5\text{sun Leeds} + \beta_6\text{sun Bristol} + \beta_7\text{winter} + \beta_8\text{spring} + \epsilon_t$$

The RMSE value of the linear regression model is **824.5.**

Forecasting Competition Approach

Round-1: (09-March) – Prediction (standard deviation): 10.30168 (0.09072) [Score: 1.395429]

To begin with, the model with the highest accuracy was found to be the linear regression model and the standard deviation was found by examining the variation in logged loads from the previous month. The double-exponential method was not chosen as it was felt the 'weekend' effect (negative value of the beta parameter accounting for the trend, on Sunday the 7th of March) might contribute towards giving an estimate that is considerably below the actual value of the load. At this point, we had not found optimal values of the alpha & beta parameters of the model which had gave the smallest RMSE.

Round-2: (11-March) – Prediction (standard deviation): 10.25442 (0.0702039) [Score: 1.272528]

For our next prediction, the group decided to use the double-exponential method to make the prediction with parameter values of alpha & beta being equal to 0.99 & 0.5, respectively. The reason for this choice was that we wanted to see how this model fared in comparison to the regression model. The previous week's variation in log loads were analysed to get an estimate for our prediction's standard deviation.

Round-3: (13-March) – Prediction (standard deviation): 10.17104 (0.06462) [Score: 1.193485]

The group had noticed when fitting the linear regression model that the binary variable for the weekend was not significant at a 5% significance level, so the model was not trusted to accurately predict the load on the day of a weekend. We had observed that on average in the last four Saturdays, the load had decreased by approximately 10% from the previous day and it was agreed among the group that the double exponential model would not be able to account for this drop-off in load values during weekends. Thus, the decision was made to predict the load on Friday using the regression model and subsequently, the group would reduce this load by 10% to make prediction for Saturday, the 13th of March. The standard deviation estimate was obtained by inspecting the variation in log loads from Thursday-to-Sunday for each of the last four weeks and taking the average of these values.

Round-4: (16-March) – Prediction (standard deviation): 10.19998 (0.05460) [Score: 1.649710]

At this point, the ARIMA model had been fitted & had given a prediction of 10.3282 for the log load on this day, while the (unoptimized) double exponential method (with parameters alpha=0.99 & beta=0.5) had given a prediction of 10.084. Since the regression model had given an estimate that was in between the two afore-mentioned predictions, the decision was made to submit this estimate. The standard deviation was acquired by analysing the variation in log loads for all the weekdays of the previous week.

Round-5: (18-March) – Prediction (standard deviation): 10.29414 (0.0412424) [Score: 2.212102]

On this occasion, it was the still yet to be optimized double exponential method's prediction which was neither the biggest nor the smallest of the three model estimates as the ARIMA & linear regression models gave predictions of 10.35 and 10.19, respectively. As the usage of the double exponential method in the previous week resulted in a good prediction, we decided to opt for double-exponential's output once again, for prediction on Thursday. The standard deviation estimate for this prediction was obtained by inspecting the variation in log loads from Monday-to-Thursday for each of the previous four weeks and taking the average of these values.

Round-6: (20-March) – Prediction (standard deviation): 10.08736 (0.0649625) [Score: 0.936937]

For similar reasons as before, the methodology used in **round 3** was once again implemented to obtain the prediction for Saturday, the 20th of March. The variation in log loads of the previous four Saturdays was used in getting the standard deviation.

Round-7: (23-March) – Prediction (standard deviation): 10.22682 (0.0504584) [Score: 2.058903]

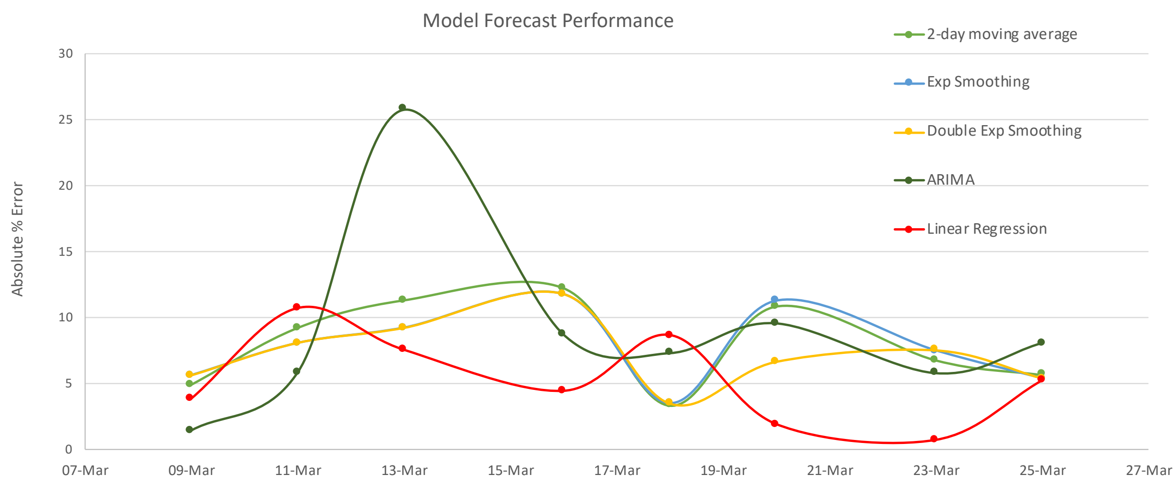
By this time, we were able to find optimal parameter values of the double exponential method and they were: alpha = 0.999 & beta = 0.001, whose prediction gave the smallest value for the three main error metrics that were considered: root mean squared error, mean absolute deviation and mean absolute percent error. The prediction it had given for Tuesday, 23rd of March, was 10.14227 while the ARIMA predicted log load value of 10.276. Since the regression model gave a prediction that was in the middle of these two predictions, we once again decided to submit this model's estimate as our prediction for the log load on the 23rd of March. The variation in log loads of the previous **seven** days was used to obtain the standard deviation.

Round-8: (25-March) – Prediction (standard deviation): 10.21845 (0.0411658) [Score: 1.517986]

On this occasion, the three main models we considered gave predictions which did not have major discrepancies. The ARIMA, (optimized) double-exponential & regression predicted 10.2453, 10.220213 & 10.21845, respectively, for the log load on the 25th of March. As the regression model's prediction had performed well on the previous occasion in comparison to the groups in this cohort we decided to make our prediction for Thursday, the 25th of March, with this approach. The variation in log loads of the previous **five** days was used to obtain the standard deviation.

Model Evaluation

In order to evaluate the performance of the models during the competition period, the values of predicted loads have been compared to the actual load for each forecast. Below, a graph of the five models and their respective absolute percentage error has been plotted for each forecast date. Both the 2-day moving average and exponential performed relatively poorly on a consistent basis, with the double exponential model performing marginally better. The ARIMA model appears to have performed reasonably well, but a large error was observed on the third forecast. The linear regression model appears to have performed very well throughout the competition's duration.



The table below shows the average absolute % error across all eight forecasted load values for each model. The ARIMA model performed the worst with an average absolute percentage error of 9.04%. The best performing model was found to be the linear regression model, with an average absolute percentage error of 5.35% across all 8 predictions. This finding coincided with the initial exploratory data analysis, where the linear model was found to have the lowest RMSE value.

Forecasting Model	Average Absolute % Error
Moving Average	8.03
Exponential	7.78
Double Exponential	7.20
ARIMA	9.04
Linear Regression	5.35

Conclusion

The initial exploratory data analysis found strong seasonality in the demand data, with demand decreasing significantly on weekends and during the summer months. Initially, low complexity models such as a two-step moving average and exponential smoothing were created to generate forecasts. One of the key findings was the very high optimal value of alpha, indicating that demand on a given day is heavily influenced by previous day's demand. Having constructed more intricate models such as the ARIMA and the linear regression model, the ARIMA was found to have a relatively high RMSE value. The initial linear regression model included variables such as the mean temperatures in UK cities, a rolling 7-day average of the demand and dummy variables for weekdays and the different seasons of the year. The Akaike information criterion (AIC) was then used to reduce overfitting and improve accuracy.

The results of the mean absolute percentage error in the forecasting period found the linear regression model to perform the best. During the competition, this model was used frequently and performed well comparatively to other groups forecasts and consequently resulted in the group finishing second overall in the competition. The use of a 7-day average of the standard deviation performed well during the competition as the demand during the week remains relatively stable. The 7-day standard deviation average was used opposed to a monthly average value, as it is unlikely that the load on a particular day would fluctuate by a great amount.

Given the above analysis and results, our recommendation is to use a linear regression model to forecast the daily demand of electricity in the UK. Furthermore, an estimated error of the forecast should be made using the five-day average standard deviation of the load. The model could have been improved having access to external data. For example, the negative relationship between energy prices and demand could be estimated by the model, as well as the impact that housing growth and population has on electricity demand. Additionally, the effect that closed offices and business have on demand due to covid-19 could be evaluated. Furthermore, a number of moving averages of the data should be implemented rather than one 7-day moving average across the period. Given no external data, a more detailed model would estimate the effect that temperature has on demand at each 6-hour time interval throughout the day to produce a more accurate daily forecast. Future models could implement more advanced machine learning techniques such as neural networks for more accurate demand forecasting. Finally, using a test-train split approach will help select the optimal model, as it tests the accuracy of the model based on difference segments of the data.