The School of Mathematics

THE UNIVERSITY
of EDINBURGH

Forecasting United Kingdom Electricity Demand

by

Aditya Prabaswara Mardjikoen, s2264710

Dissertation Presented for the Degree of
MSc in Statistics with Data Science

May 2022

Supervised by
Prof Simon Wood and Dr Maarya Sharif

Executive Summary

Between 2009 and 2021, the United Kingdom (UK) began to increase electricity generation from renewable energy (wind, solar, and biomass) and reduce electricity generation from fossil fuels [7]. However, the proportion of the electrical load generated by renewable energy sources (such as wind, wave, and solar) is impossible to estimate and control one day in advance. Furthermore, technological advancements such as the smart grid and electric cars make forecasting electricity demand difficult.

Forecasting electricity load has gotten a lot of interest from academics and businesses in recent years. Electricity supply management is critical to ensuring that the electricity generated meets demand at all times. We will use UK half-hourly energy demand data from 2011 to 2016 for short-term (hourly and daily) electricity load forecasting in this report (excluding the data in the Christmas and New Year holiday periods).

The major goal of this report is to develop a model for forecasting UK electricity demand one day in advance for each half-hour of the day (24 hours). For this, we will utilise a generalized additive model (GAM), which can take into account for the interaction between covariates that have a nonlinear relationship with the response variable (in this case electricity load) [4].

The model that we proposed in this report manages to obtain mean absolute percentage error (MAPE) around 1.68% and root mean square error (RMSE) around 774 megawatts when we implement the model to predict the electricity load 24 hours in advance from January 2016 until June 2016 after we fit the model using the UK national grid data from January 2011 until December 2015 (excluding the Christmas and New Year holiday period) at each half hour. We discover that the highest forecasted error in the entire half hour period of the day occured during spring. In addition, we also found out that the typical forecasted error during weekends can be higher than during weekdays in the morning. Moreover, we also figure out that the typical forecasted error in Friday is the highest over the entire half hour period of the day.

Acknowledgments

I am grateful to the supervisors of this project, Prof Simon Wood and Dr Maarya Sharif, for their support and valuable advice. I would also like to thank Prof Simon Wood for providing the background material for this project. Moreover, I would like to thank to the United Kingdom (UK) national grid operator for providing the data for this project.

University of Edinburgh – Own Work Declaration

Name: Aditya Prabaswara Mardjikoen

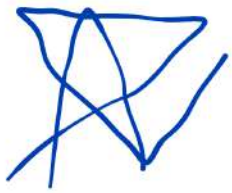
Matriculation Number: s2264710

Title of work: Forecasting United Kingdom Electricity Demand

I confirm that all this work is my own except where indicated, and that I have:

- Clearly referenced/listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Not sought or used the help of any external professional academic agencies for the work
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Complied with any other plagiarism criteria specified in the Course handbook

I understand that any false claim for this work will be penalised in accordance with the University regulations (https://teaching.maths.ed.ac.uk/main/msc-students/msc-programmes/statistics/data-science/assessment/academic-misconduct).



Edinburgh, 29 June 2022

Contents

1 Introduction **1**
1.1 Background and Motivation . . . 1
1.2 Objective . . . 1
1.3 Literature Review . . . 1
1.4 Terminology . . . 2
1.5 Data . . . 2

2 Exploratory Data Analysis **3**
2.1 Electricity Grid Load Data Analysis . . . 3
2.2 Average Daily Temperature Data Analysis . . . 5

3 Models **8**
3.1 Generalized Additive Model . . . 8
3.2 Short-Term Electricity Load Forecasting Model . . . 8
3.3 Model Fitting and Diagnostic . . . 9
3.4 Forecasting Metrics . . . 10

4 Results **10**
4.1 Forecasted MAPE and RMSE . . . 10
4.2 Daily and Half Hourly Residuals . . . 10
4.3 Half Hourly Forecasting Error Based on Day of Week . . . 13
4.4 Half Hourly Forecasting Error During Weekdays and Weekends . . . 15
4.5 Half Hourly Forecasting Error Based on Season . . . 16

5 Conclusion **18**

Appendices **20**

A R Code **20**

List of Tables

1 Comparison of model fit in training and test data 10

List of Figures

1 Load on the UK national grid, half hourly in megawatts against day since January 5th, 2011. 3
2 Average load on the UK national grid, in megawatts against half hour period of the day based on day type (weekdays and weekends) and seasons. 4
3 Average Daily Temperature on the UK, half hourly in Celcius against day since January 5th, 2011. 5
4 Mean of the average daily temperature on the UK national grid, in Celcius against half hour period of the day based on day type (weekdays and weekends) and seasons. . . . 6
5 Load against average daily temperature on the UK. 7
6 Some model residual diagnostic plots for the short-term electricity load forecasting model fitted to the training dataset. 9
7 Fitted half hourly megawatts load for the final 178 days of the data overlaid on the observed load. 11
8 Residual in megawatts for the final 178 days of the data. 12
9 Daily residuals against half hour period of the day for the final 178 days of the data. . 13
10 Residual MAPE against half hour period of the day based on day of the week. 14
11 Residual RMSE against half hour period of the day based on day of the week. 15
12 Residual MAPE and RMSE against half hour period of the day based on weekdays and weekends. 16
13 Residual MAPE and RMSE against half hour period of the day based on season. . . . 17

1 Introduction

1.1 Background and Motivation

We all use electricity as a source of energy on a daily basis. It is frequently produced using either nonrenewable or renewable energy sources. In the United Kingdom (UK), there is a significant change in electricity generation between 2009 and 2021. Wind, solar, and biomass electricity generation begin to increase, while fossil-fuel electricity generation declines [7]. The change in electricity generation during this period is for reducing greenhouse gas emissions and overall electricity consumption by approximately 67 terawatt-hours between 2010 and 2050 [7].

In the UK, especially Great Britain, larger electrical power plants and interconnectors are directly linked to the higher voltage electrical transmission system, where they must adhere to a code of operation defined in the balancing and settlement code [1]. Furthermore, companies that supply significant demand on the system, such as end-user suppliers, must follow this code. The code encourage that all parties are required to report their requested electricity generation or demand to the national grid in advance of real time to ensure that they can take several actions to ensure the maintenance of the electrical system [1, 7].

The electric company in Great Britain has different blocks where balancing electricity is managed continuosly under a 48 half-hourly settlement period (period for trading and balancing electricity) in a standard days [1, 7]. Throughout this period, the electrical system operator has numerous extra authority to escalate or reduce supply or demand on the electrical system to ensure the stability of the system at all times [7]. Thus, it is critical to manage electricity supply to ensure that the electricity generated meets energy demand at all times [6, 3].

Short-term (hourly and daily), medium-term (monthly to yearly), and long-term (5 to 30 years) electricity load forecasting has aroused the interest of industry and academia in recent years [3]. Forecasting electricity demand is critical for electricity providers for business on electricity markets. Furthermore, future electricity demand was required by electrical network administrators for the network's reliability and investment objectives.

The recent development of technology such as smart grids and electric cars has sparked a slew of new perspectives on energy management and electrical demand forecasting [3]. Furthermore, the rapid increase in the percentage of electricity generated from renewable sources makes electricity load forecasting difficult. Hydroelectric power can be easily and quickly managed, whereas tidal power output is highly predictable and reliable [2]. However, while weather forecasts can be used to anticipate wind, wave, and solar power in the short-term, they cannot be predicted more than a few days ahead of time and are difficult to control [2].

Against this background, it is critical to predict short-term electricity demand so that electricity generation can be matched to demand and incentives may be offered to large energy users, as well as small customers, to cut or increase demand to better match expected supply.

1.2 Objective

The purpose of this report is to build a forecasting model to estimate UK electricity demand one day ahead. We are aiming to observe the trend of the forecasting error in the half hour period of the day based on day of the week, day type (weekdays and weekends) and seasons.

1.3 Literature Review

We now give a review of the literature regarding electricity load forecasting. The earliest study we consider is conducted by [6], who used a generalized additive models (GAM) [9, 8, 4] on the French electricity load hourly data (collected from September 2000 to August 2005) and fitted one forecasting model for each hour. They find out that their model forecasted error is smaller when they take into account the summer break.

Another study conducted by [3] managed to formulate a GAM model to predict short-term and middle-term electricity demands using the French grid electricity load data. Their study suggest that a good temperature forecasting is essential based on the lower forecasting error that they obtained when using real temperature as explanatory variables (covariates) in the model instead of forecast temperature. In addition, they also manage to propose two model for short-term and middle term electricity load forecasting respectivel.

One of the most difficult challenges in electricity load forecasting is dealing with massive datasets. [9] manages to propose an alternative for this issue. In addition, they manage to proposed a single model to predict electricity demand using French national grid data after fitting the model at each half hour. Overall, most published literature regarding electricity load forecasting (such as [6, 3, 9]) showed the prediction results from a model that was build using non-UK electricity load data by considering meteorological variable and calendar effects. Thus, we are aiming to build a model for predicting electricity load using the UK data. Some of the analysis in this report is inspired by [9].

1.4 Terminology

We use terminology in [3] regarding the type of covariates in this report. There are two type of covariates that we will be using in this report. The first covariates is related with weather forecasting. We call this covariates as meteorological variables or meteorological predictor. The second covariates is related with calendar period or date and time variable. We call this covariates as calendar effects or calendar variable.

In this report, we used two terminology to identify a certain period of time based on the day of the week (Monday to Sunday). These terminology are weekdays and weekends. From Monday to Friday, we referred this period of day as weekdays. As for from Saturday and Sunday, we called this period of day as weekends.

Next, we will introduce four time period that we used to classify a certain period of time based on month of the year. These time period are summer, spring, autumn and winter. December, January and February are considered to be the winter period; March, April and May are considered to be spring period; June, July and August are considered to be summer period; September, October and November are considered to be autumn period.

Lastly, we will also mention one important terminology that will be used in this report. We will refer to the time of day in half hour intervals as half hour period of the day.

1.5 Data

The dataset that we are using in this report was collected by the UK national grid operator. It includes 90419 observations of electricity load (in megawatts) and temperature (in Celsius or ℃), which were measured in each half hour of a day from January 2011 until June 2016. The electricity load is physically measured directly from the grid, while the temperature data is from the UK meteorological office. The dataset does not contain electricity load and temperature data during the Christmas (25th until 31st December) and New Year (1st until 4th January) holiday periods because obtaining this data during these two periods is difficult and generally has to be measured separately.

The dataset that we are using in this project does not contains any missing values. There are some data preparation steps required before build our predictive model. First, we convert the day of week into an integer with the following descriptions: 1 represents Sunday, 2 represents Monday, 3 represents Tuesday, 4 represents Wednesday, 5 represents Thursday, 6 represents Friday, and 7 represents Saturday. Next, we add the day type (weekdays and weekends) and seasons data at each observations in the UK national grid load data that we used for this report.

2 Exploratory Data Analysis

2.1 Electricity Grid Load Data Analysis

We started by observing the half hourly electricity load trend from January 5, 2011 until June 30, 2016. Figure 1 shows the megawatts load on the UK national electricity grid at half-hour intervals, starting on January 5th, 2011. Between January 2011 and June 2016, we can see that the electricity load demand from every January after New Year holiday period until summer tends to decrease until around 20000 megawatts, whereas the electricity load demand tends to increase around 50000 megawatts after summer until December.

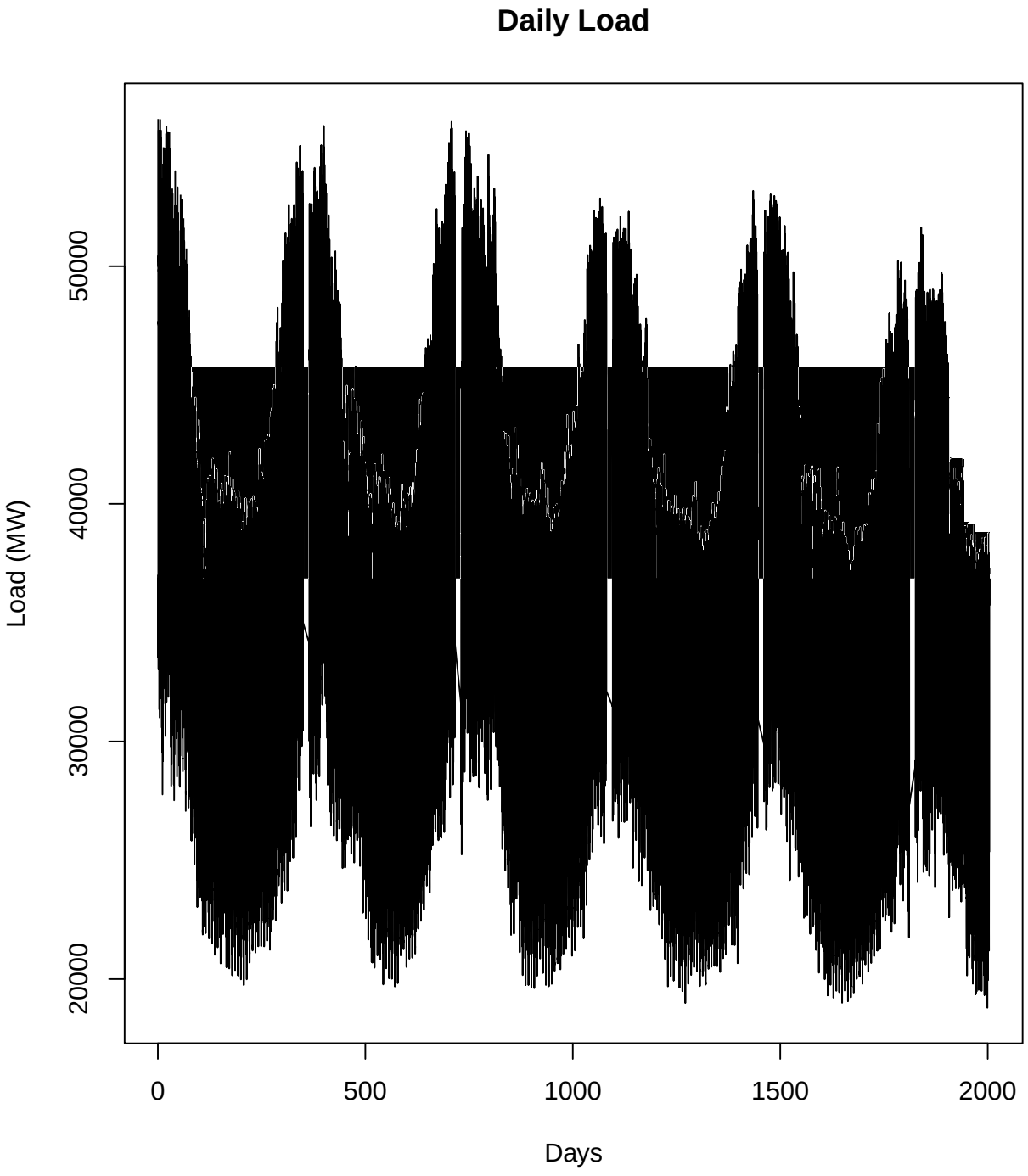


Figure 1: Load on the UK national grid, half hourly in megawatts against day since January 5th, 2011.

According to Figure 2, it appears that the average electricity demand in fhe UK increased between morning and evening, whereas it decreased in night. Overall, the peak of the average electricity demand in the half hour period in a day occured in the evening. We can see that the average electricity demand is higher during weekdays than during weekends. Moreover, the demand for electricity is the highest during winter, while it is the lowest during summer.

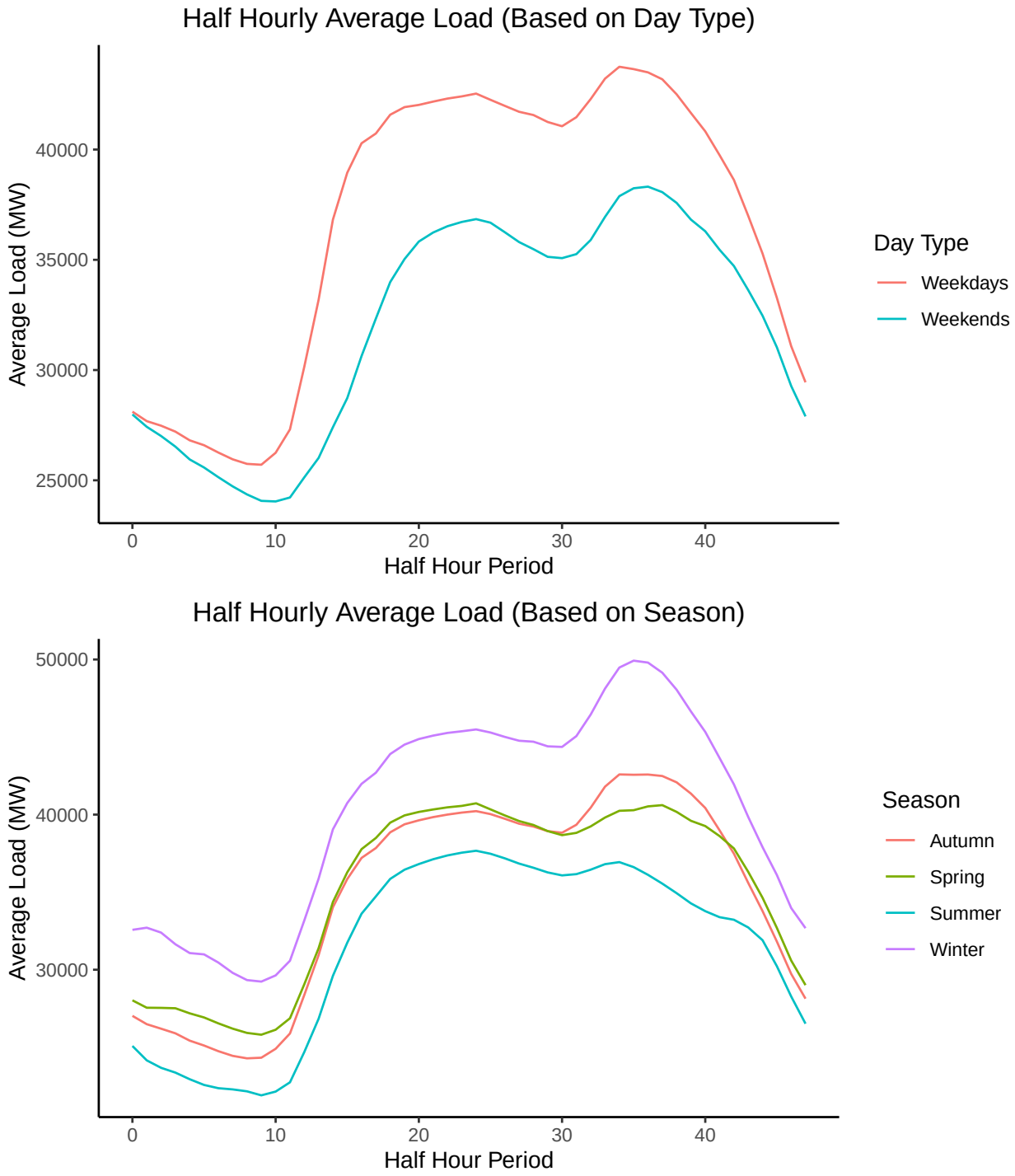


Figure 2: Average load on the UK national grid, in megawatts against half hour period of the day based on day type (weekdays and weekends) and seasons.

2.2 Average Daily Temperature Data Analysis

We will examine the half hourly average daily temperature trend from January 5, 2011 until June 30, 2016. Figure 3 shows the average daily temperature (in Celcius) on the UK at half-hour intervals, starting on January 5th, 2011. Between January 2011 and June 2016, we can see that the average daily temperature from every January after New Year holiday period until summer during this period tends to rise until it reaches roughly around 30 °C, whereas the average daily temperature tends to drop from around July until December before the Christmas holiday period til it reaches approximately less than or equal 0 °C.

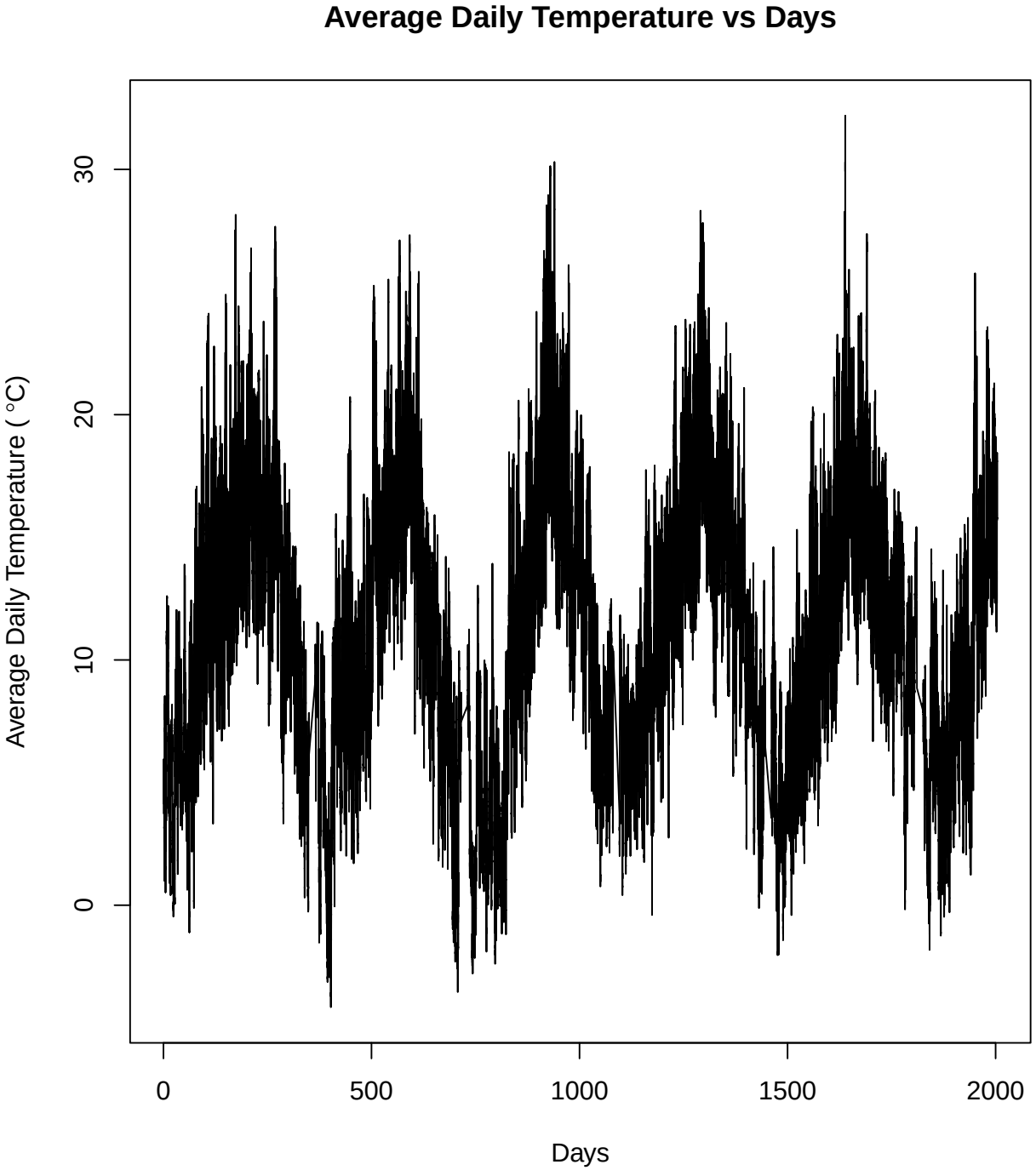


Figure 3: Average Daily Temperature on the UK, half hourly in Celcius against day since January 5th, 2011.

Next, we will examine the mean of the average daily temperature trend against half hour period of a day based on day type (weekdays and weekends) and season. Based on Figure 4, over the entire half hour period of a day, the mean of the average daily temperature during weekdays is slightly higher than during weekends. Over the entire half hour period of a day, the mean of the average daily temperature is the highest during summer, whereas it is the lowest during winter.

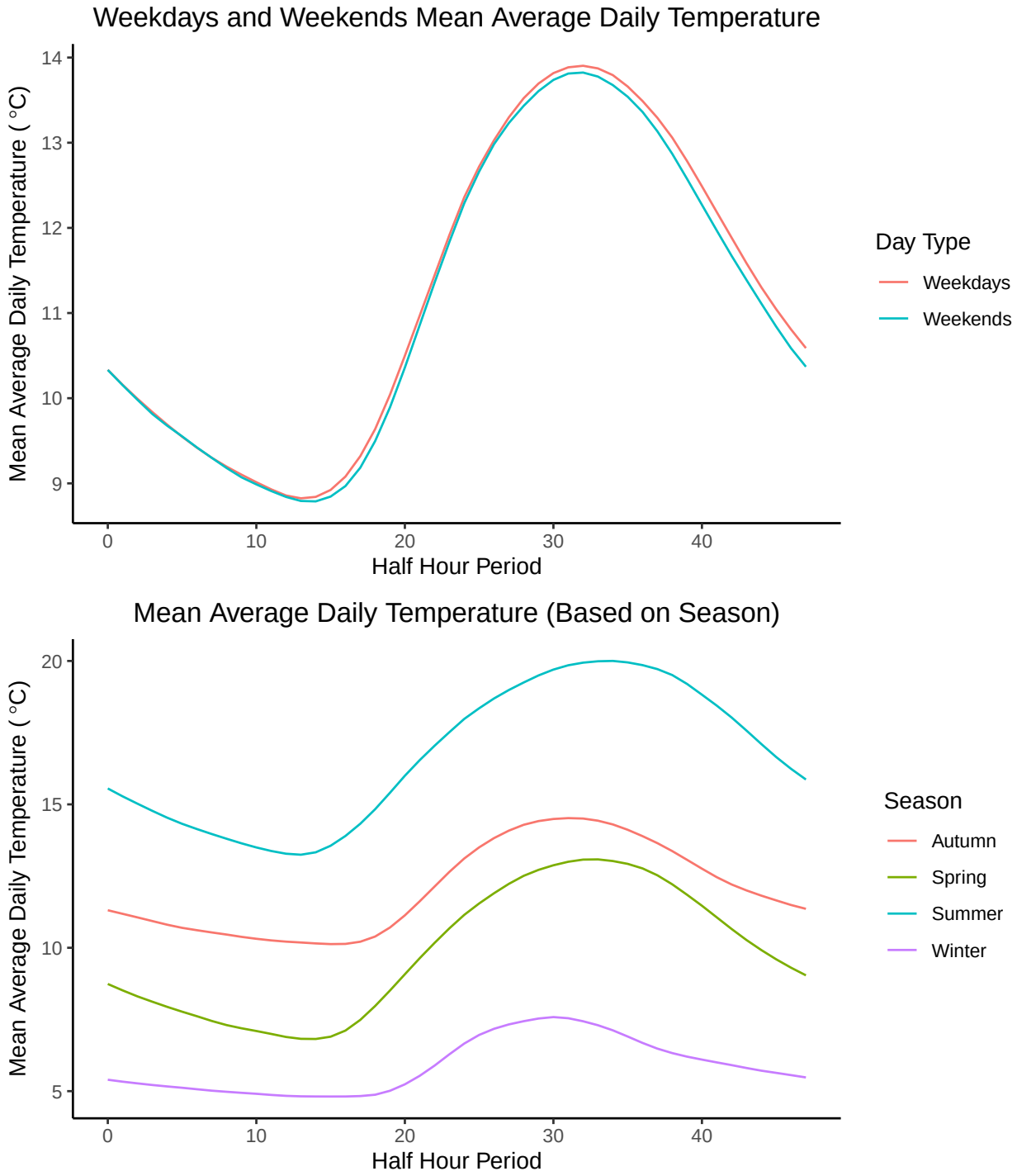


Figure 4: Mean of the average daily temperature on the UK national grid, in Celcius against half hour period of the day based on day type (weekdays and weekends) and seasons.

Lastly, we are going to investigate the relationship between the electricity load (in megawatts) in the UK national grid and average daily temperature (in Celcius). Figure 5 illustrate the nonlinear relationship between these two variables. We can see that the electricity load in the UK national grid

tends to decline as the average daily temperature increased. On the opposite, the electricity load tends to increase when the average daily temperature tends to drop. This correspond to what we observe in Figure 1 and 3. We can see that during the days where the average daily temperature is smaller, electricity demand tends to be higher compare. Therefore, it is recommended that the national grid operator should increased the electricity load supplies between after summer and before the start of Christmas holiday period.

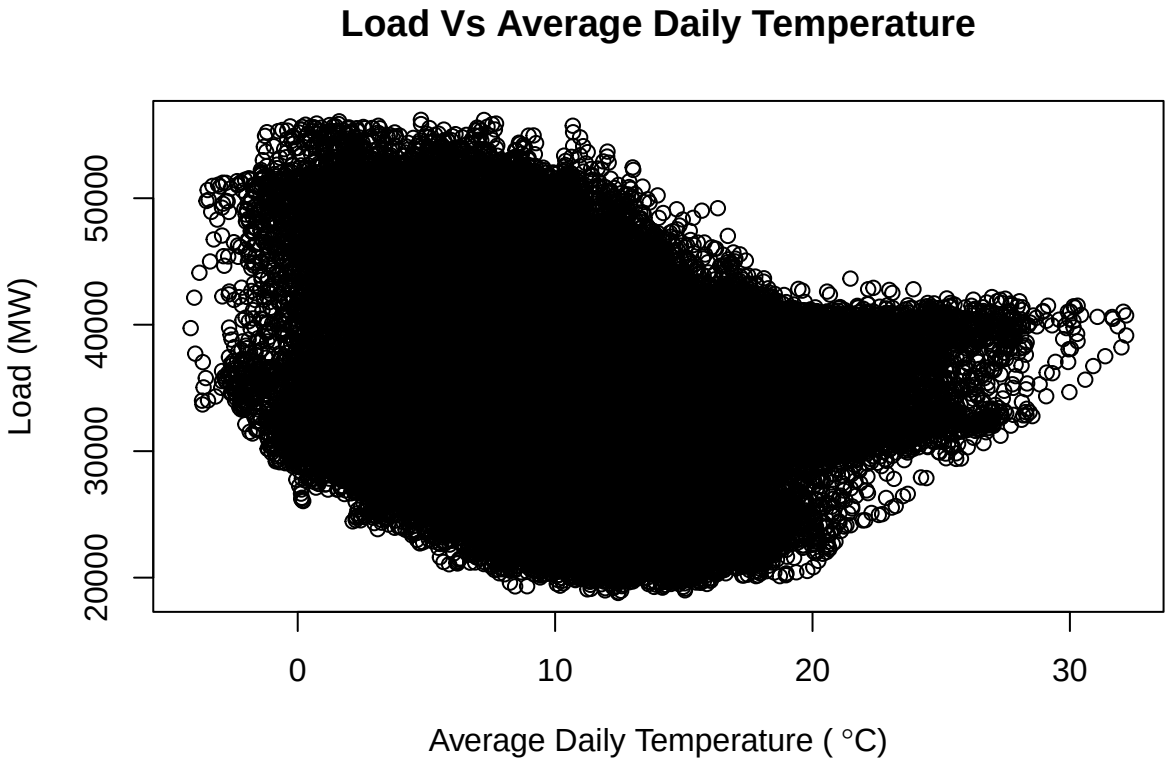


Figure 5: Load against average daily temperature on the UK.

3 Models

3.1 Generalized Additive Model

In real life problems, the relationship between the response variable and its covariates are often not linear. Traditional linear regression model is frequently failed in this circumstance since they assumed that the response variable and its covariates have a linear relationship. As a result, we need to construct a more adaptable statistical model to identify and quantifying nonlinear regression effect. These model are known as generalized additive model (GAM) [4]. In this model, the response variable is modelled as the sum of some unknown smooth functions of its covariates and a zero mean random error term [8]. Consider that we want to fit the following generalized additive model (GAM) [3]:

$$y_i = f_1\left(x_{1,i}\right) + f_2\left(x_{2,i}\right) + \cdots + f_p\left(x_{p,i}\right) + \epsilon_i,$$

where y_i is a univariate response variable, $x_{q,i}$ is a covariate, f_q are smooth functions of the covariates x_q, and ϵ_i represents the model error at time i. The error term ϵ_i is modelled as independent $N\left(0, \sigma^2\right)$ random variables, which is a normal random variables with mean zero and variance σ^2.

In GAM, we can take account the interaction between two or more covariates in the form of unknown smooth functions [8]. We denote $f\left(x_{1,i}, x_{2,i}\right)$ as the smooth function of the interaction terms between the covariates x_1 and x_2. As for the case of the interaction terms between three covariates, we denote $f\left(x_{3,i}, x_{4,i}, x_{5,i}\right)$ as the smooth function of the interaction terms between the covariates x_3, x_4 and x_5. Sometimes, when we want to take into account the interaction between several covariates in our GAM model, we can modelled it as a tensor product [8].

3.2 Short-Term Electricity Load Forecasting Model

In this report, the response variable will be the electricity grid load. As for the covariates, it will be meteorological variables (such as temperature) and calendar effects (such as day of the week, month of the year, and etc). We suggest the following model for short-term (one-day ahead) electricity grid load forecasting at the ith half hour period:

$$\begin{aligned} L_i = {} & f_1\left(L_{i-48}\right)\mathrm{day}_i + f_2\left(\mathrm{toy}_i\right) + f_3\left(t_i\right) + f_4\left(\mathrm{day}_i\right) + f_5\left(\mathrm{month}_i\right) + f_6\left(\overline{T}_i\right) + f_7\left(\theta_i\right) + f_8\left(\mathrm{year}_i\right) + \\ & + f_9\left(\mathrm{day}_i, L_{i-48}\right) + f_{10}\left(\mathrm{day}_i, \overline{T}_i\right) + f_{11}\left(\mathrm{day}_i, \theta_i\right) + \epsilon_i. \end{aligned} \tag{3.1}$$

Here L_i is the electricity grid load in megawatts at the ith half hour period; L_{i-48} is the electricity grid load 24 hours (48 half hours) previously; t is the cumulative time (total time elapsed since the start of the data), scaled to between 0 and 1; toy is time of year, as a proportion from 0 to 1; day_i represents the day of the week (an integer from 1 to 7 decribed in Section 1.5; $\overline{T}$ is the average daily temperature in Celcius; month_i is the month of the year (an integer from 1 to 12); year_i is the year; θ_i is an exponential smoothing of the real temperature T_i at the ith half hour period:

$$\theta_i = \sum_{j=1} T_{i-48j}(0.95)^j,$$

with T_{i-48j} is the real temperature at the ith half hour period, j days before. The term ϵ_i in Model 3.1 represents the model error at the ith half hour period. It is modelled as an independent $N\left(0, \sigma^2\right)$ random variables. The f_j are all smooth function that represented as penalized regression spline [8, 9]. f_1, f_3, f_6, f_7, f_8 are cubic regression splines; f_2, f_4, f_5 are cyclic cubic regression splines; f_9 is tensor product of cubic regression splines (cyclic in L_{i-48}); f_{10} is tensor product of cubic regression splines (cyclic in $\overline{T}_i$); f_{11} is tensor product of cubic regression splines (cyclic in θ_i).

3.3 Model Fitting and Diagnostic

We start by splitting the UK national grid data into two dataset: training dataset and test dataset. The training dataset will consist the UK national grid data from January 2011 until December 2015, whereas the test dataset will consist the UK national grid data from January 2016 until June 2016. The purpose for splitting the data into two dataset is to evaluate the performance of Model 3.1 when they are used to predicting electricity demand on the data that are not used to build this model. Let $\hat{\epsilon}_i$ denote Model 3.1 residuals [8] at the ith half hour period, where

$$\hat{\epsilon}_i = L_i - \hat{L}_i \tag{3.2}$$

with L_i and $\hat{L}_i$ is the actual and predicted electricity demand at the ith half hour period respectively. Figure 6 illustrate the residuals diagnostic plots [8] for Model 3.1 after we fitted it with the training data at each 48 half hours. The upper left normal $QQ-$plot is almost close to a straight line even though some point fall below the straight line, suggesting that the assumption of the residual is normally distrubuted is almost reasonable. The upper right plot (residuals against fitted values) show that the residuals of Model 3.1 approximately constant as the fitted values (in this case, the estimated electricity demand computed using Model 3.1) increased, which suggest that the assumption of constant variance is reasonable. The histogram of residuals plot at the lower left confirms that the distribution of Model 3.1 residuals appears almost approximately close with a normal distribution even though the residuals distribution looks like a left-skewed distribution. The lower right plot, the response (actual electricity demand) against fitted values (predicted electricity demand), illustrates a postive linear relationship between the response and fitted values. This would indicate that the actual electricity demand and forecasted electricity demand computed using Model 3.1 almost match up perfectly.

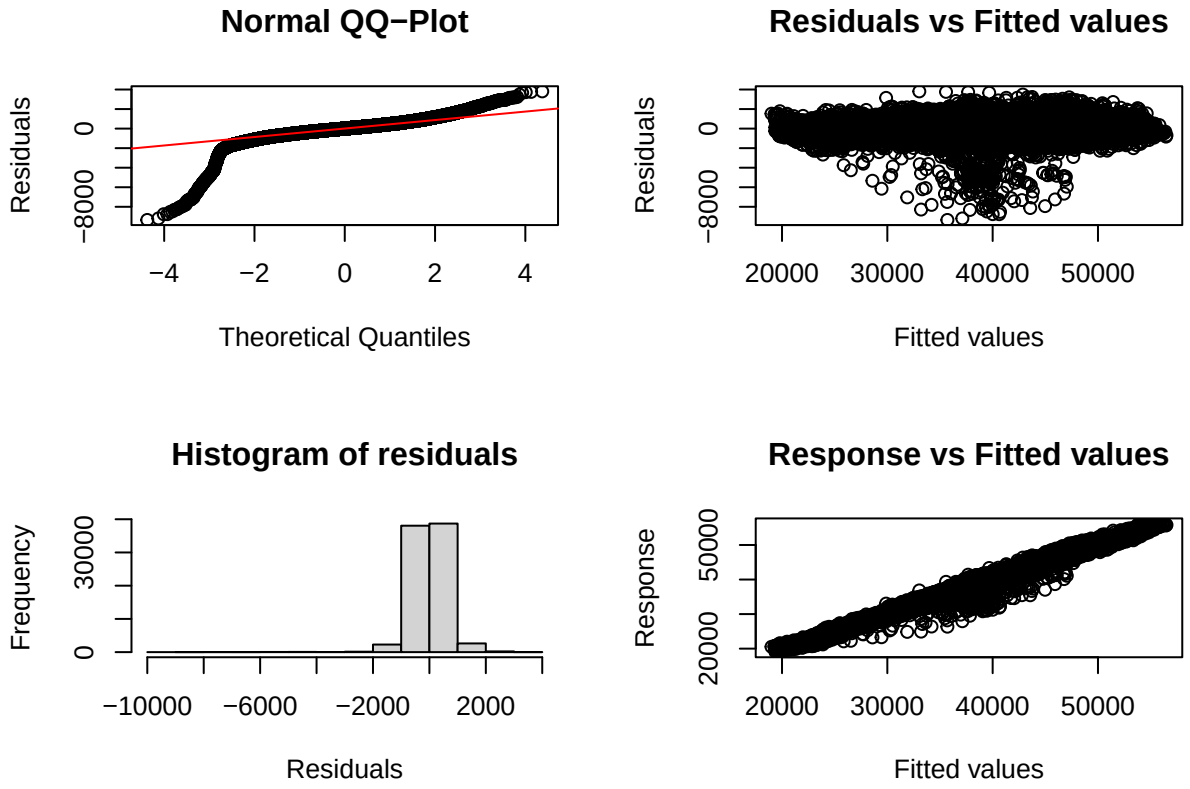


Figure 6: Some model residual diagnostic plots for the short-term electricity load forecasting model fitted to the training dataset.

3.4 Forecasting Metrics

We introduce two statistical metrics that will be used to evaluate the performance of Model 3.1. This two metrics are root mean squared error (RMSE) [10] and mean absolute percentage error (MAPE) [5]. Let $\hat{L}_i$ denoted the predicted electricity load in the ith half hour period which was estimated using Model 3.1. The RMSE metric is given by

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{L}_i - L_i\right)^2}, \tag{3.3}$$

where N is the number of observations estimated in the forecasting period. This metric is used to measure the overall forecasting error throughout the entire forecasting period. On the other hands, the MAPE metric is used to describe the average difference between the forecasted electricity demand and the actual electricity demand. For the case of electricity load forecasting in our report, this metric can be expressed as

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{L_i - \hat{L}_i}{L_i}\right|. \tag{3.4}$$

4 Results

4.1 Forecasted MAPE and RMSE

We provide the MAPE and RMSE for the training and test data that we obtained using Model 3.1 in Table 1. Through the entire forecasting period in the test dataset (January until June 2016), the MAPE and RMSEon the test dataset is around 1.68% and 774 megawatts respectively. This implies that the average predicted error and the typical error is roughly around 1.68% and 774 megawatts respectively for the test dataset.

As for the training dataset, we obtain the MAPE and RMSE through the entire forecasting period in the training dataset (January 2011 until December 2015) around 1.14% and 572 megawatts respectively. Therefore, the average predicted error and the typical error is roughly around 1.14% and 572 megawatts respectively for the training dataset.

Table 1: Comparison of model fit in training and test data

	MAPE	RMSE (MW)
Training data	0.0114	571.7836
Test data	0.0168	774.2662

4.2 Daily and Half Hourly Residuals

We observe the trend of the actual and predicted electricity demand between January 5th, 2016 until June 30th, 2016. Figure 7 illustrates the actual and predicted electricity demand during this period. Overall, the predicted electricity demand for this period manages to met with its actual demand. However, we could not see the trend of the daily residuals clearly. Therefore, we need daily residuals plot for the predicted electricty demand in the test dataset.

Figure 8 illustrates the one-day ahead predicted residuals from January 5th, 2016 until June 30th, 2016. We can see that the peak of the predicted daily residuals in this period (roughly around 4000 megawatts) occured between day 1900 until 1950, with day 1 counted from January 5th, 2011. Overall, the difference between the actual and forecasted electricty demand in between early January until end of June 2016 roughly around 2000 megawatts. As we observe the daily residuals during this period over the entire half hour period of the day using Figure 9, we discover that the worse predicted residuals occured between around 10 AM until 8 PM.

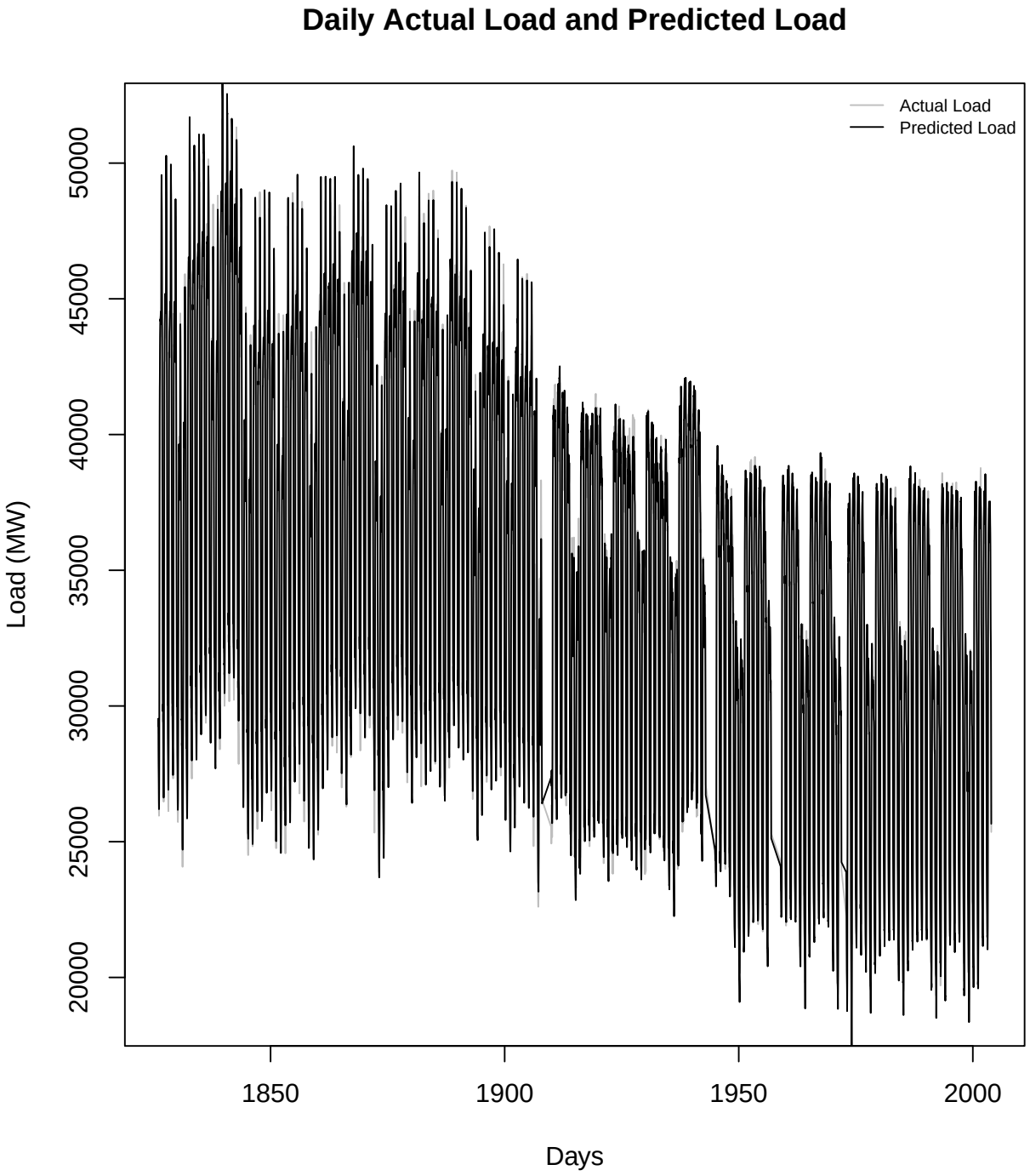


Figure 7: Fitted half hourly megawatts load for the final 178 days of the data overlaid on the observed load.

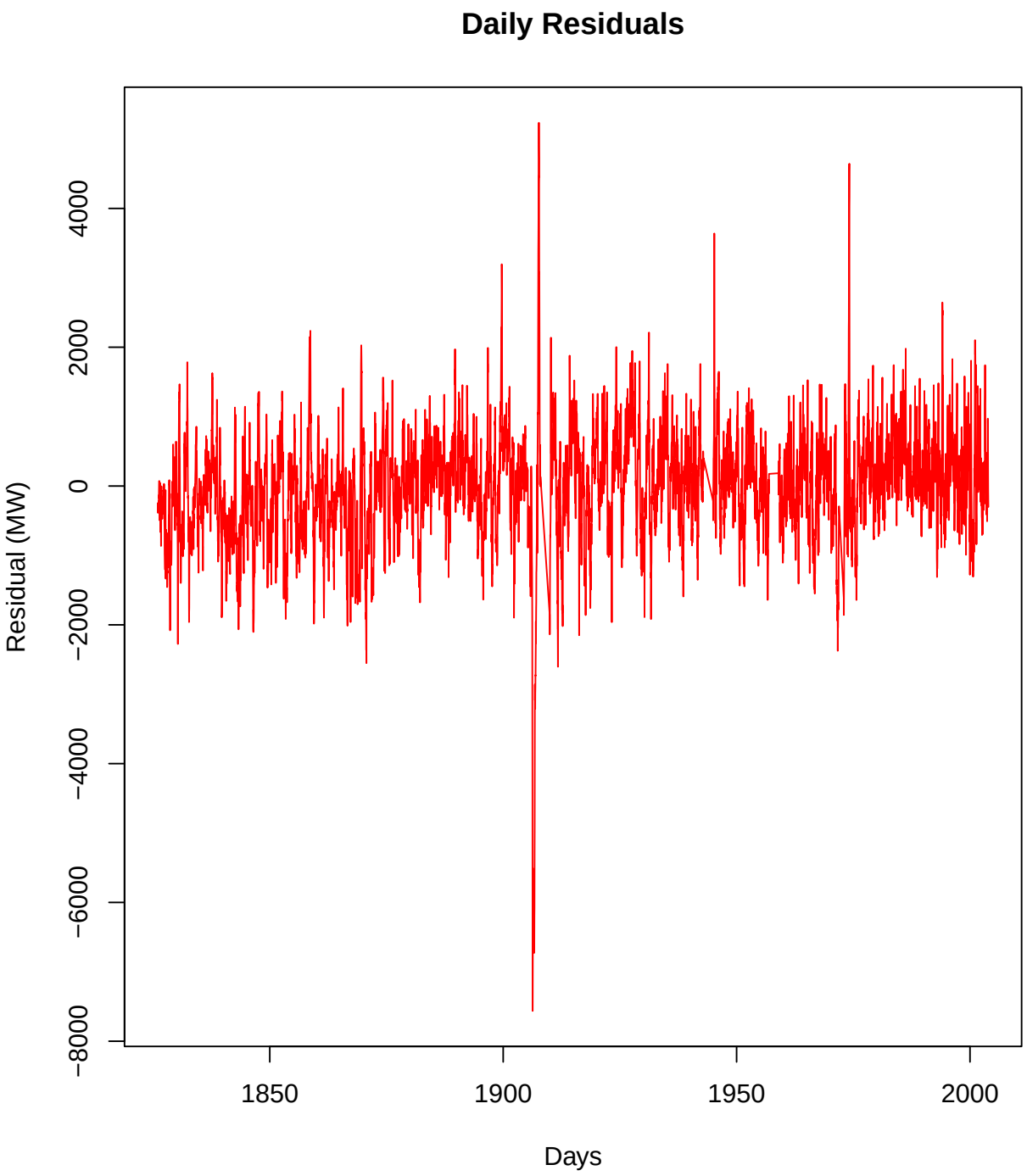


Figure 8: Residual in megawatts for the final 178 days of the data.

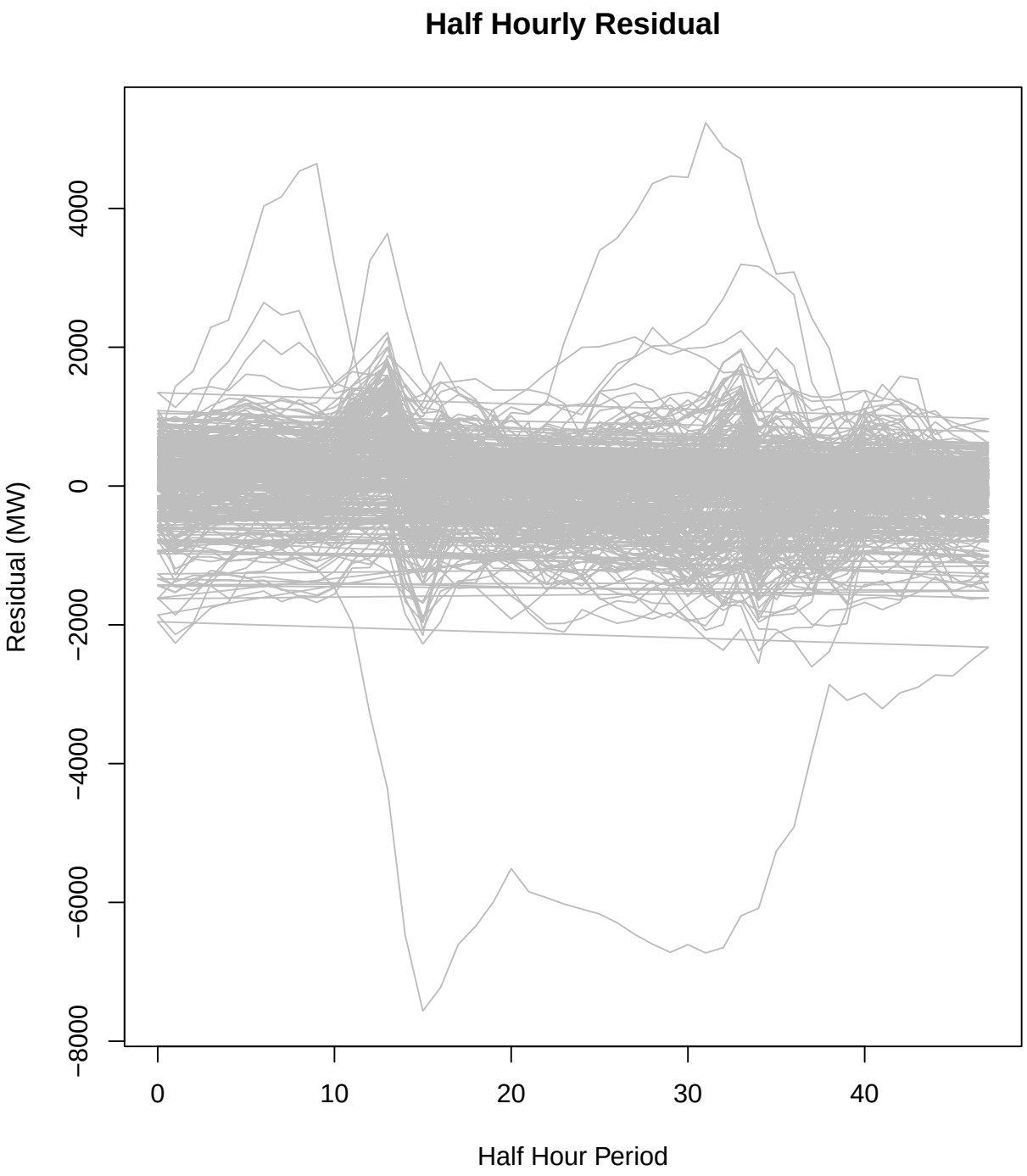


Figure 9: Daily residuals against half hour period of the day for the final 178 days of the data.

4.3 Half Hourly Forecasting Error Based on Day of Week

We are going to observe the the MAPE and RMSE against the half hour period of the day based on the day of the week (Monday to Sunday). Figure 10 illustrates the residual MAPE against half hour period of the day based on the day of the week. We can see that the major prediction error in the mornings occured in Sunday. According to the MAPE against half hour period of the day plot in Figure 10, the average difference between the forecasted electricity demand and the actual electricity demand in Sunday morning is roughly around 2% until 4%. In addition, there is also a sharp increase in the forecasted error in Saturday between 10 AM and 3 PM.

Even though the residual MAPE in Saturday is higher than during Friday in this time interval,

however, the typical difference between the actual and predicted electricity demand in Friday is higher than Saturday between 5 AM and before midnight according to Figure 11. During this period, the typical forecasted error in Friday experienced a sharped increased around 800 megawatts until 1600 megawatts.

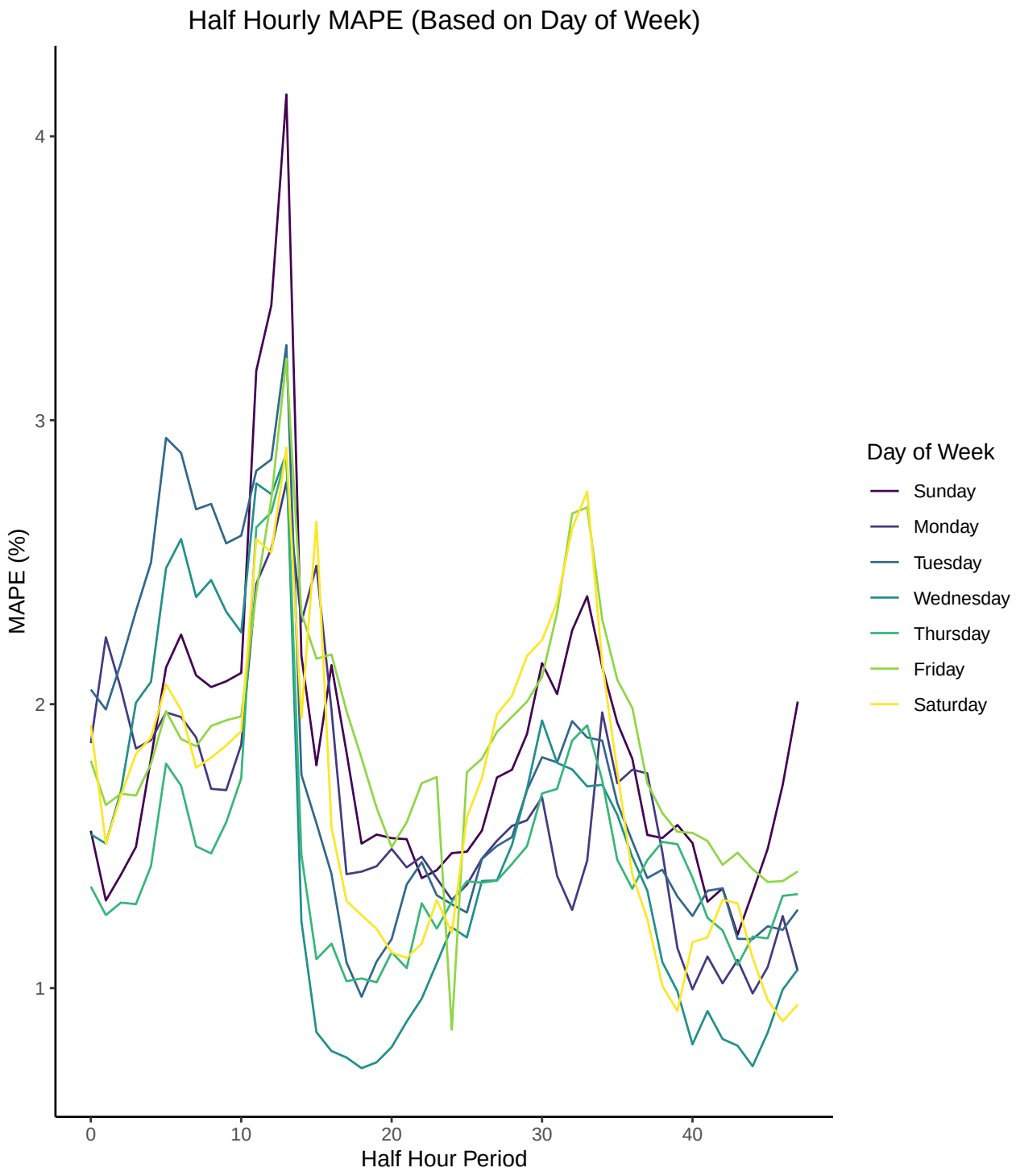


Figure 10: Residual MAPE against half hour period of the day based on day of the week.

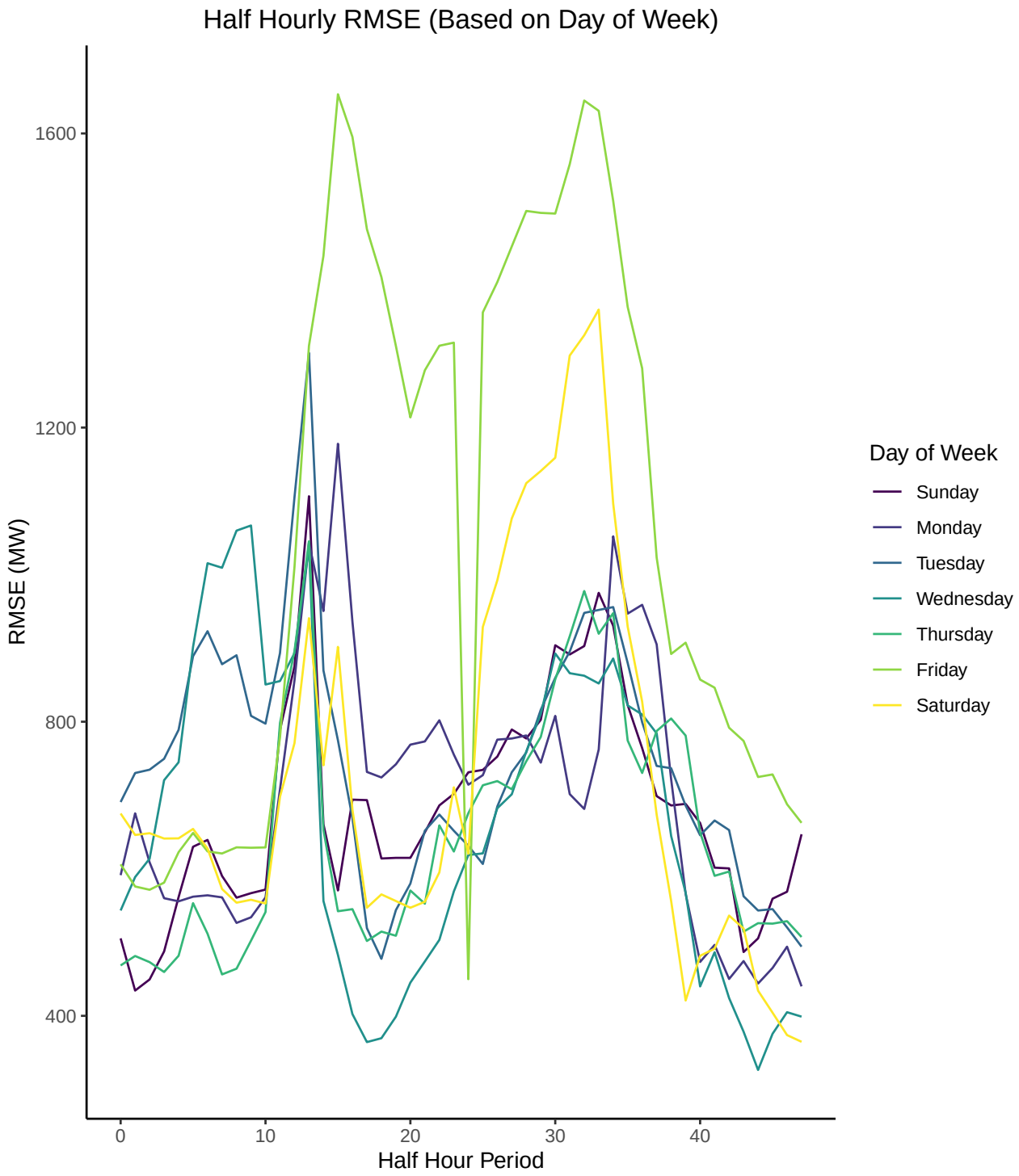


Figure 11: Residual RMSE against half hour period of the day based on day of the week.

4.4 Half Hourly Forecasting Error During Weekdays and Weekends

We are going to observe the the MAPE and RMSE against the half hour period of the day based on the type of day(weekdays and weekends). Figure 12 illustrates the residual MAPE and RMSE against half hour period of the day based on weekdays and weekends. It appears that the forecasting error during weekends over the entire half hour period of the days is higherr than during weekdays based on MAPE. The residual MAPE over the entire half hour period of the day plot in Figure 12 suggest that in the morning between around 3 AM and 9 AM the average difference between the actual and predicted electricity demand during weekends reaches its peak at around 3.5%.

Eventhough the residual MAPE during weekends is higher than during weekdays over the entire

half hour period , however the residual RMSE during weekdays is not always lower than during weekends as we observe in Figure 12. In weekdays morning,there exist a sharp increase in the typical difference between the actual and predicted electricity demand around 1200 megawatts. On the other hands, during the evening the worse difference between the actual and predicted electricity demand occured during weekends roughly around 1200 megawatts.

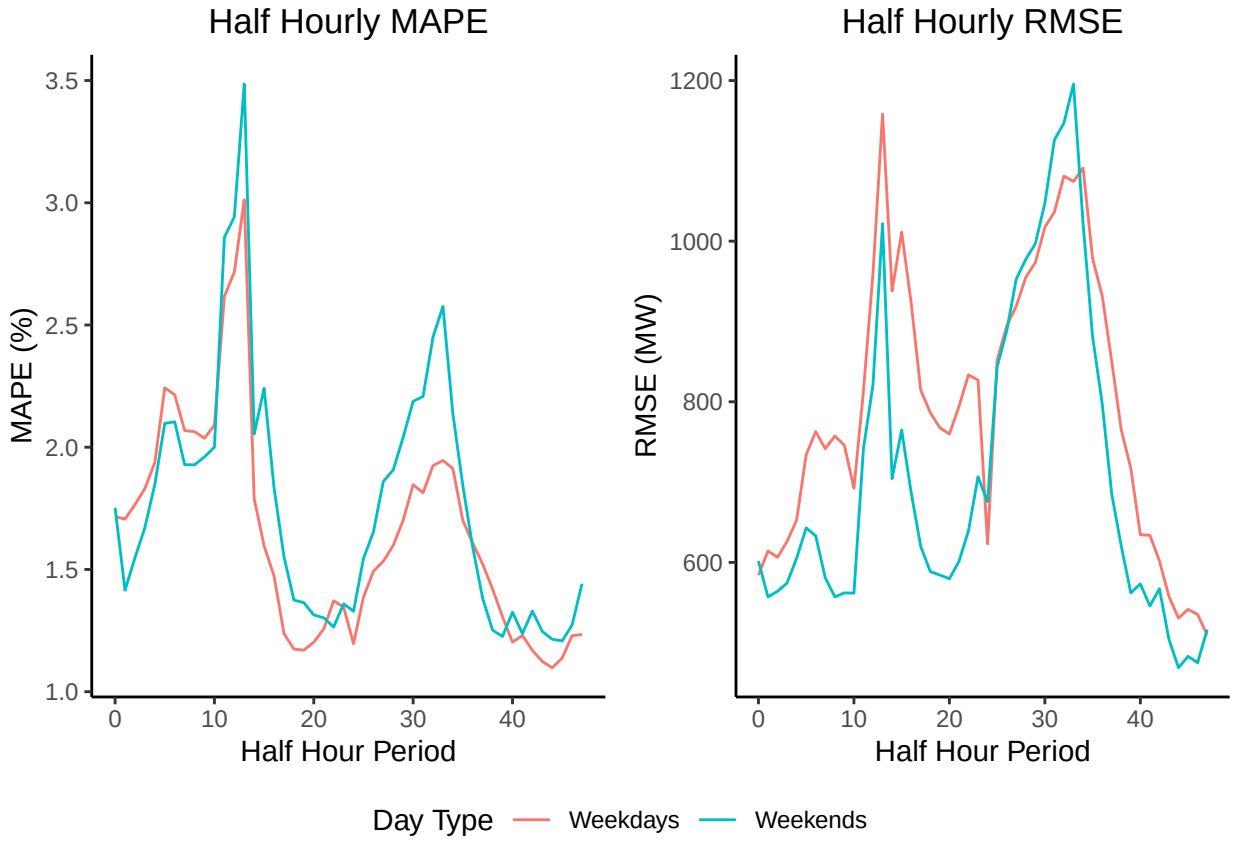


Figure 12: Residual MAPE and RMSE against half hour period of the day based on weekdays and weekends.

4.5 Half Hourly Forecasting Error Based on Season

We are going to observe the the MAPE and RMSE against the half hour period of the day based on season. Figure 13 illustrates the residual MAPE and RMSE against half hour period of the day based on season. Over the entire half hour period of the day, the forecasting error between midnight and 5 AM is the highest during summer. During this time period in summer, the average forecasted residuals can reaches around 5% while the typical forecasted error can reaches around 1250 megawatts.

Between 5 AM and 8 PM, the forecasted error during spring is the worse. On average, it can reaches around 2%. However, the typical forecasted error can be roughly around 1250 megawatts. As for the winter period, the forecasted error is reaches its peak between 10 AM and 3 PM. Typically the forecasted error in winer during this time period is roughly around 1000 megawatts, whereas the average difference between the actual and forecasted electricity demand can reaches around 2%.

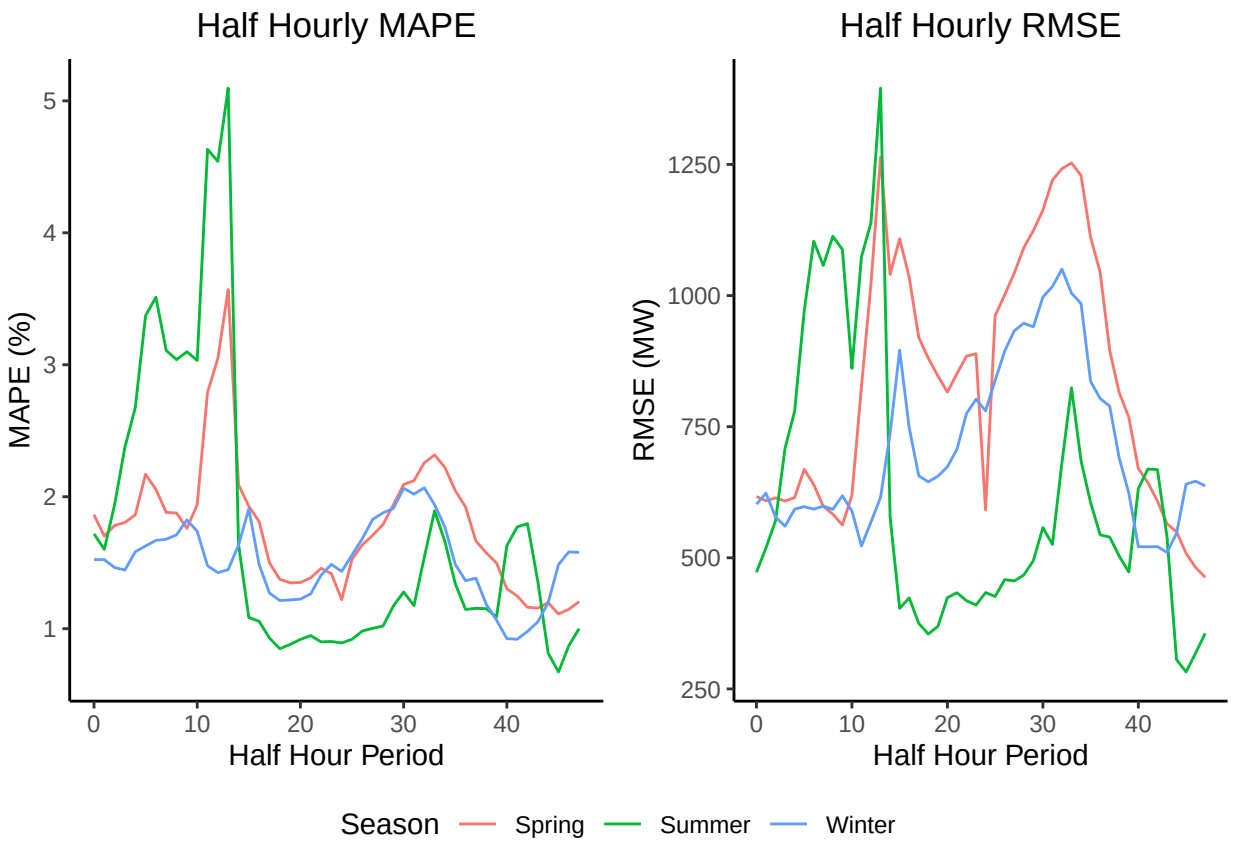


Figure 13: Residual MAPE and RMSE against half hour period of the day based on season.

5 Conclusion

We have already formulate a generalized additive model (GAM) to predict the electricity grid load (in megawatts) 24 hours in advance. Our model that we fit using the UK national grid data from January 2011 until December 2015 at each half hour manages to obtain mean absolute percentage error (MAPE) around 1.68% and root mean square error (RMSE) around 774 megawatts when we implement the model to predict the electricity demand from January 2016 until June 2016. There are three important results that we obtained after estimating the electricity demand on the UK national grid data from January 2016 until June 2016. First, we find out that the forecasted error during summer is the highest between midnight until morning, whereas the highest forecasted error through the rest of the half hour interval happened during spring. Second, even thoguh the MAPE during weekdays is higher than during weekends, however the RMSE during weekends can be higher than during weekdays in the morning. Lastly, we found out that the RMSE in Friday is the highest over the entire half hour period of the day. Moreover, the MAPE during morning is the worse in Sunday.

The limitations in the analysis that we proposed in this report is we does not take into account the electricity load demand during Christmas and New Year holiday period in the data that we fit for our GAM model and also the data that we does not fit to this model. If we take into account the demand for electricity during Christmas and New Year holiday period, then we can observe the forecasting error during the entire winter period.

References

[1] Balancing & settlement code. *Elexon BSC*, April 2022. https://www.elexon.co.uk/bsc-and-codes/balancing-settlement-code/. (Accessed 9 June 2022).

[2] G. e. Boyle. *Renewable electricity and the grid: The challenge of variability*. Earthscan, 2009.

[3] Y. Goude, R. Nedellec, and N. Kong. Local short and middle term electricity load forecasting with semi-parametric additive models. *IEEE Transactions on Smart Grid*, 5(1):440–446, 2014.

[4] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, second edition, 2009.

[5] S. Kim and H. Kim. A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, 32(3):669–679, 2016.

[6] A. Pierrot and Y. Goude. Short-term electricity load forecasting with generalized additive models. *Proceedings of ISAP power*, 2011, 2011.

[7] I. G. Wilson, S. Sharma, J. Day, and N. Godfrey. Calculating great britain's half-hourly electrical demand from publicly available data. *Energy Strategy Reviews*, 38:100743, 2021.

[8] S. N. Wood. *Generalized Additive Models: An Introduction with R*. Texts in Statistical Science. London: Chapman & Hall, second edition, 2017.

[9] S. N. Wood, Y. Goude, and S. Shaw. Generalized additive models for large data sets. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 64(1):139–155, 2015.

[10] J. Zhang, A. Florita, B.-M. Hodge, S. Lu, H. F. Hamann, V. Banunarayanan, and A. M. Brockway. A suite of metrics for assessing the performance of solar power forecasting. *Solar Energy*, 111:157–175, 2015.

Appendices

A R Code

The code for loading the used libraries and creating functions to perform the entire data analysis in this report.

```
## Loading required library and defined required function
library(mgcv)
library(dplyr)
library(tidyverse)
library(ggpubr)
library(patchwork)
library(data.table)
library(lubridate)

mape_compute <- function(ytrue,ypred){
  # Function to compute MAPE
  # Input: ytrue = actual values, ypred = predicted values
  # Output: MAPE

  error_ratio <- (ypred-ytrue)/ytrue
  mape <- mean(abs(error_ratio),na.rm = TRUE)
  return(mape)
}

rmse_compute <- function(ytrue,ypred){
  # Function to compute RMSE
  # Input: ytrue = actual values, ypred = predicted values
  # Output: RMSE

  mse <- mean((ypred-ytrue)^2,na.rm = TRUE)
  rmse <- sqrt(mse)
  return(rmse)
}

season_convert <- function(month_name){
  # Function to identify season in UK based on month
  # Input: month_name = month name
  # Output: name of season in UK based on month

  # Convert integer (1-12) to month name
  if(is.numeric(month_name)==T|is.integer(month_name)==T){
    month_name <- month.name[month_name]
  }

  # Identify winter season
  if(month_name %in% c(month.name[c(1,2,12)],month.abb[c(1,2,12)])){
    season <- "Winter"
  }

  # Identify spring season
  else if (month_name %in% c(month.name[c(3,4,5)],month.abb[c(3,4,5)])){
    season <- "Spring"
```

```
  }

  # Identify summer season
  else if (month_name %in% c(month.name[c(6,7,8)],month.abb[c(6,7,8)])){
    season <- "Summer"
  }

  # Identify autumn season
  else{
    season <- "Autumn"
  }
  return(season)
}

error_data <- function(ape,square_error,list_var){
  # Function to produces MAPE and RMSE summary table
  # Input:
  # ape = absolute percentage error, square_error = square error,
  # list_var = list of factor or categorical data
  # Output: MAPE and RMSE summary table
  error_df <- aggregate(cbind(mape=ape,mse=square_error), list_var, FUN=mean)
  error_df$rmse <- sqrt(error_df$mse)
  error_df$mse <- NULL
  return(error_df)
}
```

The code for reading the data and perform data preparation step in Section 1.5:

```
## Loading the data and perform data preparation steps

# Load UK grid load data and convert date into julian day
load("UKL.RDA")
day <- julian(UKL$date, origin = min(UKL$date))

# Add full name of day of week
UKL$day_desc <- wday(UKL$date, label=TRUE,abbr=FALSE)

# Add month name data
UKL$month_desc <- factor(month.name[UKL$month],levels=month.name)

# Convert day of week to integer (1 to 7)
UKL$day_num <- as.numeric(UKL$dow)

#categorize day based on weekdays and weekends
weekend_day <- c("Saturday","Sunday")
UKL$day_type <- ifelse(UKL$day_desc %in% weekend_day,"Weekends","Weekdays")
UKL$day_type <- factor(UKL$day_type)

#Add season data based on month
UKL$season <- unlist(lapply(UKL$month,season_convert))
UKL$season <- factor(UKL$season)
```

The code for creating plots in Section 2 to perform exploratory data analysis:

```r
## Exploratory data analysis

# Plot load against day
plot(day,UKL$load,type='l',xlab='Days',ylab='Load (MW)',main="Daily Load")

# Average load against time of day based on day type
p1 <- ggplot(UKL, aes(x=tod, y=load)) + aes(colour = day_type) +
  stat_summary(fun = mean, geom="line")+
  labs(x="Half Hour Period", y = "Average Load (MW)",colour="Day Type")+
  theme_classic()+
  ggtitle("Half Hourly Average Load (Based on Day Type)")+
  theme(plot.title = element_text(hjust = 0.5))

# Average load against time of day based on season
p2 <- ggplot(UKL, aes(x=tod, y=load)) + aes(colour = season) +
  stat_summary(fun = mean, geom="line")+
  labs(x="Half Hour Period", y = "Average Load (MW)",colour="Season")+
  theme_classic()+
  ggtitle("Half Hourly Average Load (Based on Season)")+
  theme(plot.title = element_text(hjust = 0.5))

p1/p2

# Plot average daily temperature against day
temp_text <- expression("Average Daily Temperature ("*~degree*C*")")
plot(day,UKL$temp,type='l',xlab="Days",ylab=temp_text,
     main="Average Daily Temperature vs Days")


# Mean average daily temperature against time of day based on day type
tempavg_text <- expression("Mean Average Daily Temperature ("*~degree*C*")")
p3 <- ggplot(UKL, aes(x=tod, y=temp)) + aes(colour = day_type) +
  stat_summary(fun = mean, geom="line")+
  labs(x="Half Hour Period", y = tempavg_text,colour="Day Type")+
  theme_classic()+ggtitle("Weekdays and Weekends Mean Average Daily Temperature")+
  theme(plot.title = element_text(hjust = 0.5))

# Mean average daily temperature against time of day based on season
p4 <- ggplot(UKL, aes(x=tod, y=temp)) + aes(colour = season) +
  stat_summary(fun = mean, geom="line")+
  labs(x="Half Hour Period", y = tempavg_text,colour="Season")+
  theme_classic()+ggtitle("Mean Average Daily Temperature (Based on Season)")+
  theme(plot.title = element_text(hjust = 0.5))

p3/p4

# Plot load against average daily temperature
plot(UKL$temp,UKL$load,xlab=temp_text,ylab="Load (MW)",
     main="Load Vs Average Daily Temperature")
```

The code to fit the GAM model, produces residual diagnostic plots and comparison table in Section 3:

```r
## GAM model for electricity forecasting

# Split the data into training and test data
train.index <- which(UKL$year != max(UKL$year))
train.dt <- data.table(UKL[train.index,])
test.dt <- data.table(UKL[-train.index,])

# Initialize list to store GAM model, RMSE, MAPE
model_list <- list()
mape_test <- list()
rmse_test <- list()
mape_train <- list()
rmse_train <- list()

# Initialize list to store training and test data
test_list <- list()
train_list <- list()

# Fit proposed GAM model at each half hour
for (i in unique(UKL$tod)){
  tod_train <- which(train.dt$tod==i)
  tod_test <- which(test.dt$tod==i)
  model_list[[i+1]] <- bam(load ~ s(timeCount, bs = "cr",k=10)+
                             s(load48,by=day_num,bs = "cr",k=10)+
                             s(toy, bs = "cc", k = 10)+
                             s(day_num, bs = "cc", k = 7)+
                             s(month, bs = "cc", k = 12)+
                             s(temp, bs = "cr",k=10)+
                             s(temp95, bs = "cr",k=10)+
                             s(year,bs="cr",k=3)+
                             ti(day_num,load48,bs=c("cr","cc"),k=c(7,10))+
                             ti(day_num,temp,bs=c("cr","cc"),k=c(7,10))+
                             ti(day_num,temp95,bs=c("cr","cc"),k=c(7,10)),
                           select=TRUE,data = train.dt[tod_train,],family = gaussian)

  # Compute test data RMSE and MAPE at each half hour
  rmse_test[[i+1]] <- rmse_compute(test.dt$load[tod_test],
                                   predict.gam(model_list[[i+1]],test.dt[tod_test,]))
  mape_test[[i+1]] <- mape_compute(test.dt$load[tod_test],
                                   predict.gam(model_list[[i+1]],test.dt[tod_test,]))

  # Compute training data RMSE and MAPE at each half hour
  rmse_train[[i+1]] <- rmse_compute(train.dt$load[tod_train],
                                    model_list[[i+1]]$fitted.values)
  mape_train[[i+1]] <- mape_compute(train.dt$load[tod_train],
                                    model_list[[i+1]]$fitted.values)

  # Store the training data at each half hour
  train_list[[i+1]] <- train.dt[tod_train,]

  # Add the training data predicted load and residual
  train_list[[i+1]]$resid <- model_list[[i+1]]$residuals
  train_list[[i+1]]$pred_load <- model_list[[i+1]]$fitted.values
```

```
  # Store the test data at each half hour
  test_list[[i+1]] <- test.dt[tod_test,]

  # Add the test data predicted load
  test_list[[i+1]]$pred_load <- predict.gam(model_list[[i+1]],test.dt[tod_test,])
}

# Concatenate the training data at each half hour
train_set <- bind_rows(train_list, .id = "column_label")

# Order the training data by date and half hour period
train_set <- train_set[order(date,tod),]

# Concatenate the test data at each half hour
test_set <- bind_rows(test_list, .id = "column_label")

# Compute residual, squared error, absolute percentage error
test_set$resid <- test_set$load-test_set$pred_load
test_set$square_error <- test_set$resid^2
test_set$ape <- 100*abs(test_set$resid/test_set$load)

# Order the test data by date and half hour period
test_set <- test_set[order(date,tod),]

# Model residual diagnostic plots
par(mfrow=c(2,2))
qqnorm(train_set$resid, ylab="Residuals",main = "Normal QQ-Plot")
qqline(train_set$resid,col="red")
plot(train_set$pred_load, train_set$resid, xlab = "Fitted values",
     ylab = "Residuals", main = "Residuals vs Fitted values")
hist(train_set$resid, xlab = "Residuals", main = "Histogram of residuals")
plot(train_set$pred_load, train_set$load, xlab = "Fitted values",
     ylab = "Response", main = "Response vs Fitted values")
```

The code to produces plot and table in Section 4:

```
## Forecasting result table and plot

# Model fit comparison table
rmse_vec <- c(mean(unlist(rmse_train)),mean(unlist(rmse_test)))
mape_vec <- c(mean(unlist(mape_train)),mean(unlist(mape_test)))
error_df <- data.frame(mape = mape_vec, rmse = rmse_vec)
colnames(error_df) <- c("MAPE", "RMSE (MW)")
rownames(error_df) <- c("Training data", "Test data")
error_df

# Plot daily actual and predicted load in test data
test_index <- which(UKL$date %in% test_set$date)
plot(day[test_index],test_set$load,xlab="Days",ylab="Load (MW)",
     col="grey",type="l",main="Daily Actual Load and Predicted Load")
lines(day[test_index],test_set$pred_load,col="black")
legend("topright", legend=c("Actual Load", "Predicted Load"),
       col=c("grey", "black"),lty=1,bty='n',cex=0.7)
```

```r
# Plot residual vs day in test data
plot(day[test_index],test_set$resid,xlab="Days",ylab="Residual (MW)",
     type="l",col="red",main="Daily Residuals")

# Plot residual vs half hour period
plot(test_set$tod,test_set$resid,xlab="Half Hour Period",
     ylab="Residual (MW)",type="l",col="grey",main="Half Hourly Residual")

# Plot MAPE vs half hour period based on day of week
list_var <- list(tod=test_set$tod,dow=test_set$day_desc)
dow_error <- error_data(ape,square_error,list_var)
ggplot(aes(x=tod,y=mape,group=dow,color=dow),data=dow_error)+
  geom_line()+theme_classic()+
  labs(x="Half Hour Period",y="MAPE (%)",colour="Day of Week")+
  ggtitle("Half Hourly MAPE (Based on Day of Week)")+
  theme(plot.title = element_text(hjust = 0.5))

# Plot RMSE vs half hour period based on day of week
ggplot(aes(x=tod,y=rmse,group=dow,color=dow),data=dow_error)+
  geom_line()+theme_classic()+
  labs(x="Half Hour Period",y="RMSE (MW)",colour="Day of Week")+
  ggtitle("Half Hourly RMSE (Based on Day of Week)")+
  theme(plot.title = element_text(hjust = 0.5))

# Plot MAPE vs half hour period based on day type
list_var <- list(tod=test_set$tod,day_type=test_set$day_type)
day_error <- error_data(ape,square_error,list_var)
p7 <- ggplot(aes(x=tod,y=mape,group=day_type,color=day_type),data=day_error)+
  geom_line()+theme_classic()+
  labs(x="Half Hour Period",y="MAPE (%)",colour="Day Type")+
  ggtitle("Half Hourly MAPE")+
  theme(plot.title = element_text(hjust = 0.5))

# Plot RMSE vs half hour period based on day of week
p8 <- ggplot(aes(x=tod,y=rmse,group=day_type,color=day_type),data=day_error)+
  geom_line()+theme_classic()+
  labs(x="Half Hour Period",y="RMSE (MW)",colour="Day Type")+
  ggtitle("Half Hourly RMSE")+
  theme(plot.title = element_text(hjust = 0.5))

ggarrange(p7,p8,nrow=1,ncol=2,common.legend=TRUE,legend="bottom")

# Plot MAPE vs half hour period based on season
list_var <- list(tod=test_set$tod,season=test_set$season)
season_error <- error_data(ape,square_error,list_var)
p9 <- ggplot(aes(x=tod,y=mape,group=season,color=season),data=season_error)+
  geom_line()+theme_classic()+
  labs(x="Half Hour Period",y="MAPE (%)",colour="Season")+
  ggtitle("Half Hourly MAPE")+
  theme(plot.title = element_text(hjust = 0.5))

# Plot RMSE vs half hour period based on season
```

```
p10 <- ggplot(aes(x=tod,y=rmse,group=season,color=season),data=season_error)+
  geom_line()+theme_classic()+
  labs(x="Half Hour Period",y="RMSE (MW)",colour="Season")+
  ggtitle("Half Hourly RMSE")+
  theme(plot.title = element_text(hjust = 0.5))

ggarrange(p9,p10,nrow=1,ncol=2,common.legend=TRUE,legend="bottom")
```