ETM
ENGINEERING & TECHNOLOGY MANAGEMENT

Title:

Reflection and Learning in Energy Demand Forecasting

Course Title: Energy Demand Forecasting

Course Number: ETM 510

Instructor: Dr. Sule Balkan, Dr. Tim Anderson

Term: Winter

Year: 2017

Author:

Nicole Wehner

File Name: 2017-W-510-01_NWehner

ETM OFFICE USE ONLY
Report No:
Type:
Note:

Introduction

Every 2-3 years, a global competition is held to forecast the energy demand in ISO New England's territory, called the Global Energy Forecasting Competition (GEFCom). The competition is organized by Dr. Tao Hong of University of North Carolina and sponsored by the IEEE Power and Energy Society. The department of Engineering and Technology Management at Portland State University held a course in the winter quarter of 2017 centered on this competition and teaching the students about forecasting with a hands on, practical application. Taking this class was a unique experience and an interesting crash course on forecasting, energy grid operations, and utilizing a new tool: the computer program and language called R.

Each day, electrical grid system operators adjust power generation resources to meet real time load demands. Because electricity cannot be easily stored, generation sources have to be adjusted constantly to meet current demands. Throughout the US, certain regions have organizations called independent system operators (ISO) to do this. ISO New England is one of these organizations and covers a service territory of Main, Vermont, New Hampshire, Massachusetts, Connecticut, and Rhode Island. ISO NE covers 7.1 million retail electricity customers, which serves a population of 14.7 million with a generation capacity of 30,500 MW [1]. To effectively serve their customers, they create forecasts of energy usage to be prepared for upcoming demand.

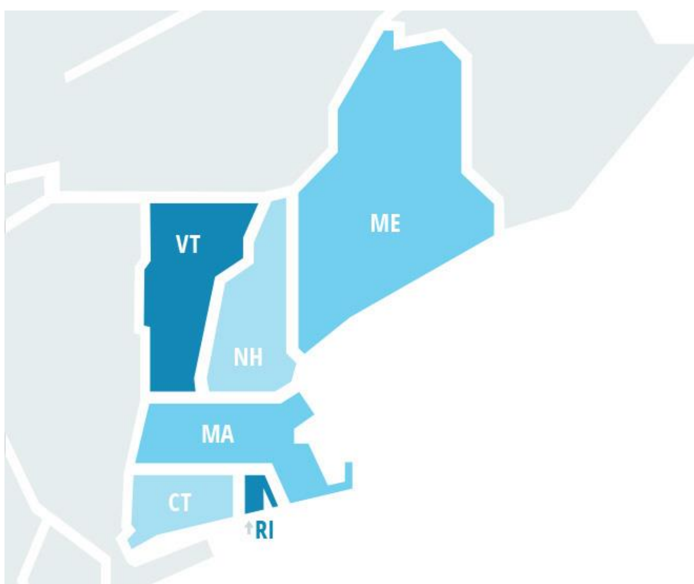

Figure 1. ISO New England's Operating Area. [1]

Prior to taking this class, I was aware of ISOs and their role in the power industry, but did not know the class would be based on GEFCom. My own experience in the power industry had me interested in the class based on the name, and peaked my curiosity enough to sign up for it despite the fact there was no course description. My job as a design engineer for power plants and substations has me interacting very little with grid operators, although I have submitted documents to other ISOs, such as ERCOT in Texas and California ISO. When designing and building a power plant, there is planning for operations by completing reactive power and load flow studies up to the point of interconnection to show the utility and system operators that the plant can support the necessary voltage and reactive power schedules.

Once the plant goes online however, there is no feedback to the design engineer, unless we are involved in the testing and start-up of the plant, of which I am usually not.

After the first day of class, there was minor panic in realizing what I had gotten myself into. My basic knowledge of the power grid would help me in the class, but no background in probability, statistics, or data analysis, would put me at a severe disadvantage. In addition, my last run in with any sort of programming, short of using formulas in Excel, was over five years ago. This probably wouldn't have been such an issue, if it weren't for the fact that I also happen to struggle with it immensely and dislike it heavily. This all added up to a big hesitancy to continue the class. I approached Dr. Balkan that evening after the first class, and she tried to assuage my fears by telling me it would be very collaborative and I would have access to a lot of help.

I decided to try it, and complete my forecasts utilizing R, a language and programming environment created for data and statistical analysis. Downloading R Studio went smoothly, but trying to load certain packages did not. It unearthed the feeling of being very unprepared and ignorant again. After meeting with Dr. Anderson to discuss useful references however, I felt back on track and like I should see the class through. Talking with another student in the class and sharing resources with her helped me to know I was not the only one with a lack of experience. Hearing other students talk about their forecasts and learning R strengthened that feeling.

I committed to the class because I felt it would help me achieve the following goals:

1. Learn more about the electrical grid and how utilities and independent system operators conduct their daily operations.
2. Give me an introduction to forecasting that would tie in with my career goals of getting involved in energy policy. I believe it is important to understand the details of all aspects of the energy industry – design, construction, and operations – in order to make educated decisions for the future of power generation, and that this class would help me understand the operations side of the power grid.
3. Learn a new programming environment that would help me in my future Engineering and Technology Management classes, such as Operations Research.

Research

Because I felt I had enough to take on as is, I chose the defined data option for the competition rather than the open data track. This meant the only data I was allowed to use to forecast was calendar data, historical demand, and temperature data. The open track allowed for any and all sources of data, such as economic data or weather forecasting. This kept me focused on reviewing forecasting methods and researching R's capabilities rather than searching for external data, despite the fact that I find the usage trends in relation to new distributed energy and economic impacts very interesting.

My two primary sources for learning R were the books *R in a Nutshell* by Joseph Adler, and *R in Action* by Robert Kabacoff. These two were supplemented by Stack Overflow, an online forum for programmers. R is an open source program so there were many free sources easily obtainable online. *R in a Nutshell* showed me the basics of working in R and how to manage data and *R in Action* introduced me to linear models and regression.

I found the class guest speaker the most interesting and helpful in directing my thinking. In his presentation he discussed the differences between residential, commercial, and agricultural loading profiles. These differences are caused by usage differences. Residential areas are characterized by high loads during the summer due to air conditioning units. Daily residential loads are highest in the morning and evening while commercial loads are largest during the day and run year round. Agricultural loads are higher in the spring and fall because of planting and harvesting [2].

For forecasting sources, I started with the book that Dr. Anderson recommended, *Practical Time Series Forecasting with R: A Hands on Guide.* This introduced me to the Forecasting package in R, which had many different modelling methods. A source I relied heavily on was the online textbook *Forecasting: Principles and Practice,* by Rob J. Hyndman and George Athanasopoulos. I found that many of the papers specific to the competition or to energy demand forecasting were not as helpful to me because I didn't have enough of the basic knowledge to understand them, but that the online book was introductory enough for me to follow along.

Observations and Analysis

Round 4

My time prior to submitting a forecast for round 4 focused on learning R, how to read and write the data, and learning the trends of the data. This brought the normal challenges of using a new software and learning a new programming language. In general, learning the syntax of a new language has been very difficult for me. R however was fairly straightforward, because it is a higher level language that already has a lot of syntax and functions built in. Because it is open source, many people have created packages to help with data processing and analysis, which made it easier to use.

I started with the code that was given on the first day of class. This utilized a more sophisticated package of reading data in from Excel called XLConnect. This package relies on Java, which created some issues when using because of my computer settings. Once these were straightened out, it worked quite well for reading in data and writing it back to an Excel sheet. This saved a lot of time with copying and pasting data into different sheets. The code also uses the tibble package, which is a way of storing data in a more organized way than the standard vectors, arrays, and matrices in R. I found I was not able to use many functions with the data stored as a tibble, so I converted the data to a numeric list to more easily utilize in the various functions of the forecasting package.

R in a Nutshell discussed time series, which I thought to be the key to using data sets for forecasting. Storing a list as a time series meant you didn't have to store date or hour, and could assign a specific period to the series. For the demand data the natural choice for this was to put it in terms of hours and set the period to be 24 hours in one day. To access and pull specific entries in a time series meant thinking in terms of index number. It complicated things slightly to use 2016 data along with other years because it included a leap year. The month of March in 2016 started with entry (31+ 29) * 24 = 1,440 but the index of the forecast started 24 entries sooner because 2017 was not a leap year.

Coming from a science background, I thought of my forecast in terms of variables, and I knew that the variable that would have the highest effect on energy usage was temperature. A graph of demand vs temperature using the package ggplot2 showed this to be true, as Figure 2 shows.

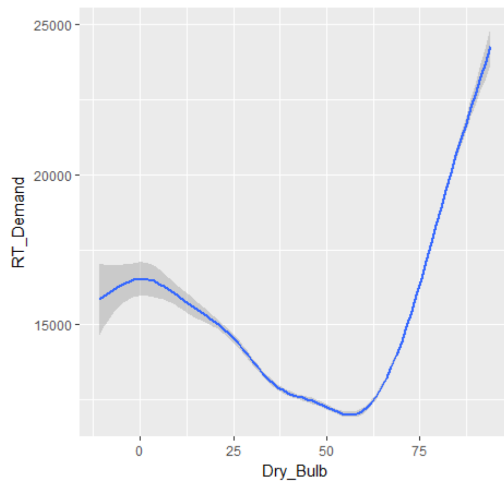

Figure 2. ISO NE total load plotted against temperature for 2016.

Once I confirmed this relationship to be strong, I started to attempt to fit a curve to this, so that my forecast would take temperature, and hopefully hour, and come up with a forecast number. I tried linear, polynomial, and sinusoidal curves but nothing came close. I then decided to limit my data in an attempt to make it easier, and looked at just March of 2016. This relationship is shown in Figure 3. The trend was not as clear with the much smaller data set however, and a linear fit yielded a line right in the middle of the plot.

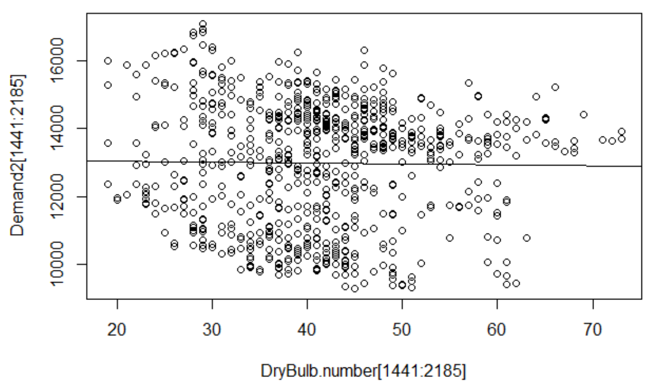

Figure 3. Temperature vs Demand for March 2016 with linear fit.

As can be seen from the plot, there are two groups of data, a set of points that ranges all temperatures but is at a higher demand, and group of range of temperature at a lower demand. I believe this is because in the month of March, in a region like the North East, there is a very wide range of temperatures at all hours of the day. The beginning of the month starts like winter, but the end of the month usually brings much warmer weather, so there is a wide range of temperatures at all hours of the day. Within the month of March, it could be 30 degrees in the middle of the day, at the beginning of the month, but then at the end of the month more likely be 30 degrees at night, which is what creates the two groups of points, one group at night and one during the day. This I think happens more so in a seasonal transitional month like March than a month in the middle of a season, like July or January.

After getting nowhere with my temperature based model, and getting closer and closer to the submission deadline, I turned to an autoregressive function, ar(), and prediction, predict(), from the Forecasting package utilizing March 2016 data. This gave me the demand shown in Figure 4. The code for this round can be found in Appendix A.

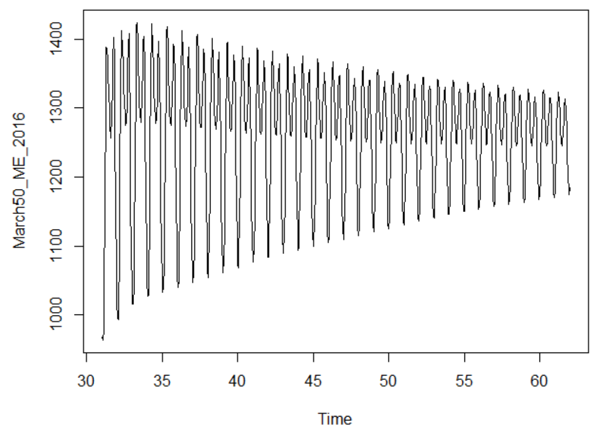

Figure 4. Round 4 results using ar() and predict() in the R Forecasting package.

Within the autoregressive function in R, there were three different models that could be used: Yule-Walker, Burg, and ordinary least squares. I tested each one, and ended up using Burg because it gave what I thought were the most realistic results.

In round 4, for my quantile distribution, I considered the spread of the maximum and minimum demands, which was about 500 MW. This correlated to about 20% above my calculated values and 20% below. I then split that between the four quantiles above 50 and four quantiles below 50, giving an 5% increase or a 5% decrease on each quantile one.

Round 5

Between round 4 and round 5 Reed Davis visited our class and presented several items to consider when building my model. One of which was the concept of heating days and cooling days. Because statistical models cannot utilize negative values, BPA separates their model into cooling days, in which they measure the number of degrees below 50˚F, and heating days, in which they use the number of degrees above 60˚F [2]. This gave me the idea to split my temperature variable into two, and then have two different models, rather than fitting one curve to the whole range of temperatures.

Something else Reed Davis said when he presented however, was that they cannot depend on temperature predictions more than three days in advance. BPA does three types of forecasts: short term, which is a couple days in advance, medium term, which is a couple months away, and long term, which is for the next year [2]. They do not use temperature for anything beyond short term, which made me realize that depending on temperature as a variable didn’t make sense in my case either. If I used temperature as an input to my model I was still stuck trying to predict it. If I still had to figure out how to predict temperature, why not just predict demand?

In Round 5 I was able to clean up my code some to make it run faster. In round 4, when compiling my finished results, I had to close out of R every time I ran a new region because I ran out of memory. To fix this, in round 5 I removed workbooks and the tibble variables after importing and converting to time series. I also ran the code in the console rather than sourcing the code.

For round 5 I used March 2015 in addition to 2016. When plotting 2015's data I noticed a 0 value, presumably a fault had occurred. To correct this, I found the value in the data in Excel, and copied the previous hour into that cell. After that, I added a line of code to check for zeros in any data and replace with the previous hour's data.

As a model, I tried using the forecast() function in the Forecast package. This function used stl() on the data, which decomposes the data into seasonal, trend, and irregular components and uses loess. Loess is the method of local polynomial regression fitting in which the fit for a point uses points local to it, with a weight to how far away it is from the point [4]. This model automatically creates upper and lower confident intervals at 95% and 80%, shown in Figure 5.

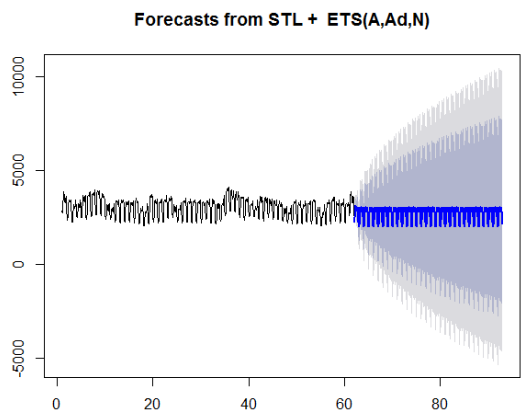

Figure 5. Round 5 prediction using forecast().

The fact that the upper and lower confidence intervals are over 5000 MW more than either the highest or lowest value shows me this is not a good model. Despite that, it helped me understand the different parts of a model: the trend, seasonality, and residual component. I was able to plot each separately in R, which helped me visual the difference between them and the affect they have on the end result.

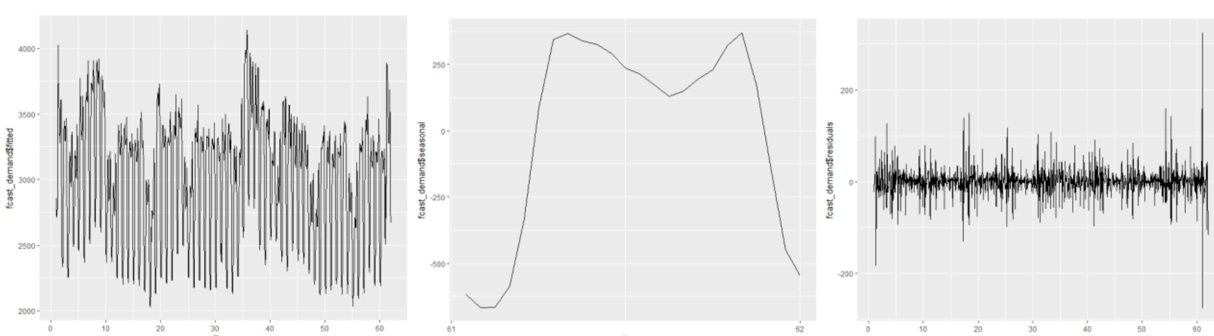

Figure 6. The fitted, seasonal, and residual components of stl() from left to right.

In round 5, I decided to utilize a more statistical approach to quantiles and used the quantile function in the forecast package. I used Rob J. Hyndman's blog to do this [5]. Because my confidence intervals were so large however, I had to create a loop to check for zero or a negative value in the 10th quantile and replace it with the 19th quantile.

Round 6

Between round 5 and round 6 Dr. Balkan showed a couple videos in class, one of which mentioned the Holt-Winter's model use in energy forecasting. After some Googling, poking around Rob J. Hyndman's website, and searching the forecasting package documentation, I found the function for a double seasonal Holt-Winter method, or dshw(). I looked specifically for methods that allowed for multiple seasonal periods because it seemed like there are many different seasonal patterns in energy demand: there is a daily cycle, a weekly cycle, and also a yearly/quarterly cycle.

I struggled some with whether a week could be considered a period. I felt that there was a similar trend each week, and similar energy usage every Monday, and then every Tuesday and so on, which would help to include a decline of usage on the weekends. I also considered a seasonal period of an entire year, and thought that if you could designate the entire year as a period, and feed several years of data into the model, then it would automatically look at last year's value on that day and hour and take that into account.

With some testing, I found that using a year as a period wasn't going to work with my programming. It may have been the memory available on my computer, or the way I was setting it up, but I got error after error. Using a week period helped capture the swing of weekends, as can be seen in Figure 7.

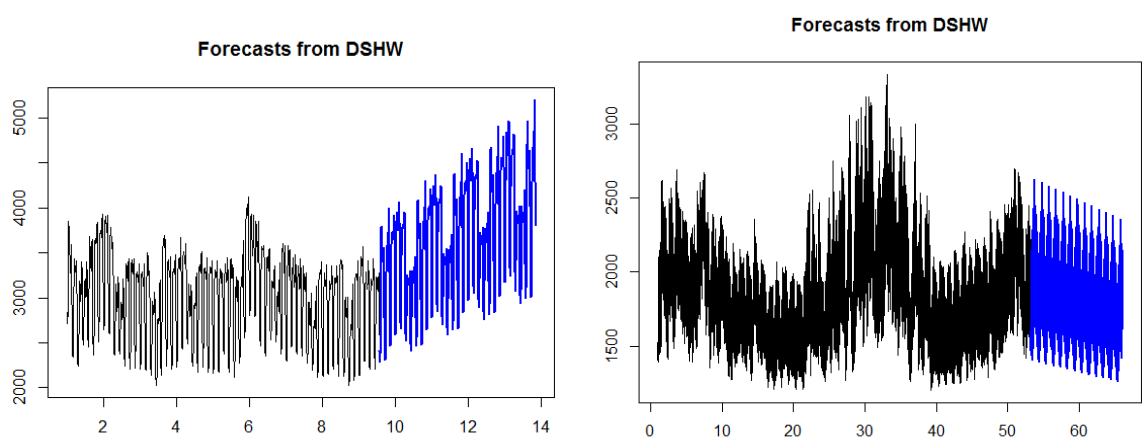

Figure 7. Prediction using dshw() with two years of April data and a period of one day and one week (left), and predition using 2016 data with a period of one day and one week (right).

Using a period of on week captured the weekends, but did not capture the trends of the year, which resulted in a steady trend up or steady trend down. This seasonality between the year, I thought was more important, as it would yeild more realistic results. After meeting with Dr. Balkan, I altered my period to be quarterly, which captured the seasons of the year. This seemed to cause a direct shift in forecast, which I think was because the data didn't start at the true beginning of a season so it was starting the prediction also in the middle of a season. By adding on the rest of winter 2015 at the beginning of 2016, it created a full season and showed the prediction to be more continuous, as shown in Figure 8.

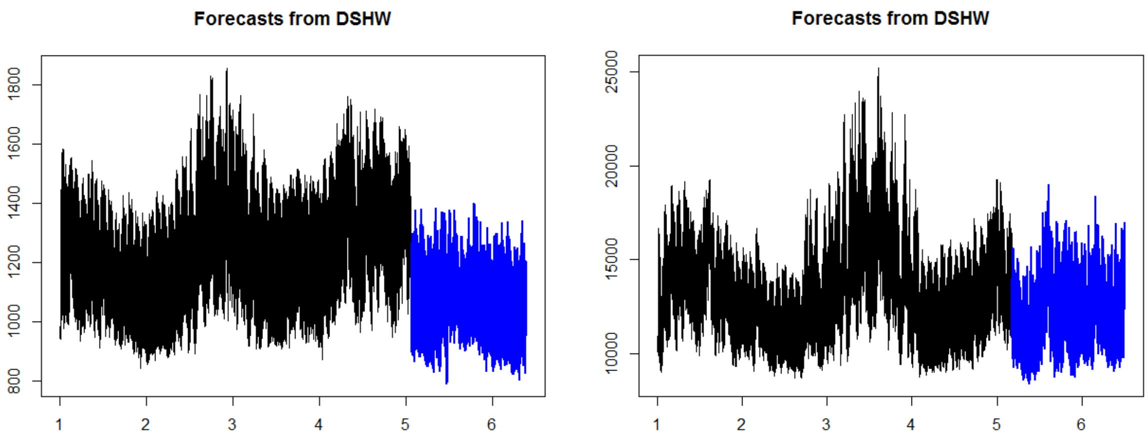

Figure 8. Double seasonal Holt Winter's method prediction without seasonal data (left) and with (right).

After getting the result shown in Figure 8 for the total New England load, I tried it on other regions. About half the time, the prediction looked realistic and similar to Figure 8, but then half the time the prediction looked like Figure 9. In these cases, I used only April data to and the weekly period to get a prediction like the left side of Figure 7. Although this also was clearly not correct, it was close to the deadline and looked better than the other options.

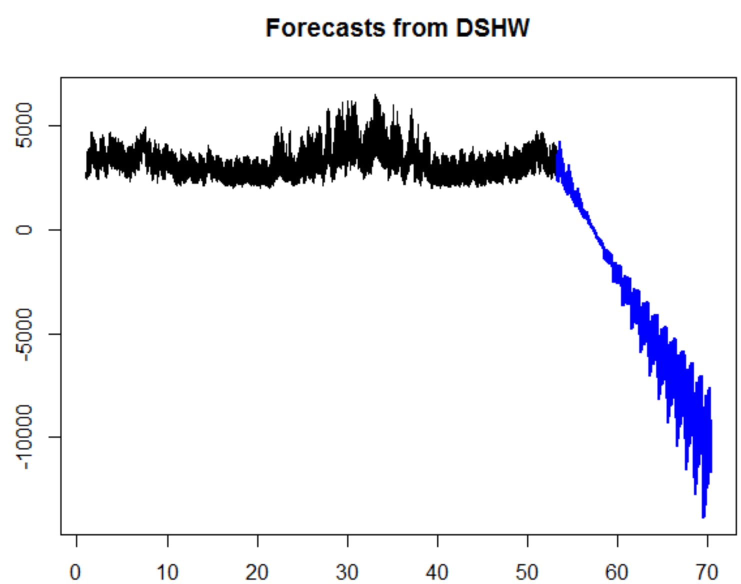

Figure 9. Connecticut demand forecast using the Double Seasonal Holt Winter's method.

Because using the upper and lower limits of the prediction gave me such a large, unrealistic spread, I decided to take a look at the maximum and minimum energy demand values in recent years and assume a normal distribution instead. I felt this would account for the likely range of temperatures and give it a realistic spread of predicted values.

Results and Discussion

Throughout the forecasting process, several data trends were found. The seasonality aspect was discussed above, including seasonal trends between daily loads, weekly loads, and quarterly seasons. Within a day, it was found consistently that the times of peak load where in the morning and in the evening. This was presumably because the loads were primarily commercial and residential, meaning that people would get up and get ready to work and some would already be at work and out and about while some were still at home, versus during the day where fewer people are at home so the load shaves off a little. This happens again in the evening when some people are still out and at work, while some are already home. Then of course at night, as people wind down and go to bed, the load drops off heavily. This can be shown by the seasonal component break out in Figure 6.

The weekly trend was that the energy usage was high Monday – Thursday, slightly lower on Friday, and lower still on Saturday and Sunday. Again, this was likely because there were fewer commercial loads on the weekend. This trend can be seen in Figure 7.

Seasonally, it was very easy to tell that the loads peaked in the summer, were the lowest in fall and spring, and had a slight peak in winter because of heating and cooling loads. The weather in the North East is mild in fall and spring so heaters and air conditioners were running less of the time.

To test my three models, I plotted them all on one graph for the same day against the actual load for two days in March. One, a weekday, March 10, and the other a weekend day, March 5. The results can be seen in Figures 10 and 11.

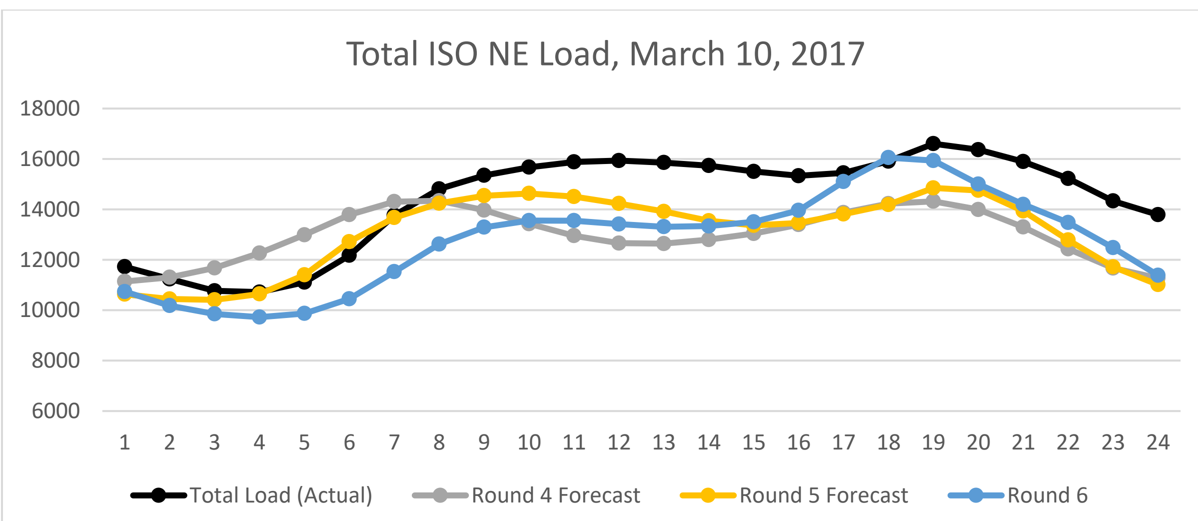

Figure 10. Three rounds of forecasts for the total ISO NE load compared against actual weekday value.

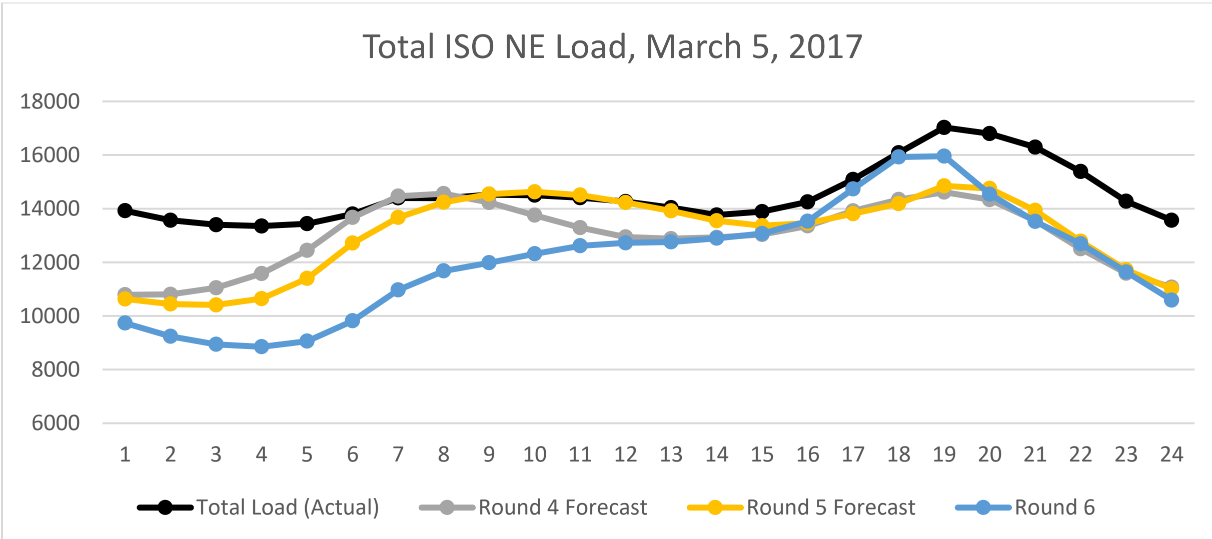

Figure 11. Three rounds of forecasts for the total ISO NE load compared against actual weekend day value.

For these two days, I found that the total squared error for all hours was as Table 1 shows. Surprisingly, it shows that both my round 4 and round 5 predictions were closer than round 6. My hypothesis is that round 6 would do better over the entire year because it would have the seasonal peaks of winter and summer, but that since the forecast period was March, when the temperature and loads were milder, my more uniform method in round 5 was a safer, middle-of-the road prediction. I think that round 4 would have showed more error as the month went on because of its eventual convergence at one value.

Table 1. Total Squared Error			
	Round 4	Round 5	Round 6
Sunday, March 5, 2017	79,879,969	83,281,806	188,062,930
Friday, March 10, 2017	109,320,585	61,460,850	74,422,182

I also wanted to see the difference between my quantiles for the last two rounds, since they both assumed normal distribution, but one used the standard deviation of all values (round 6) versus the standard deviation of the upper and lower prediction limits (round 5). As expected, Round 6's distribution looks much more realistic.

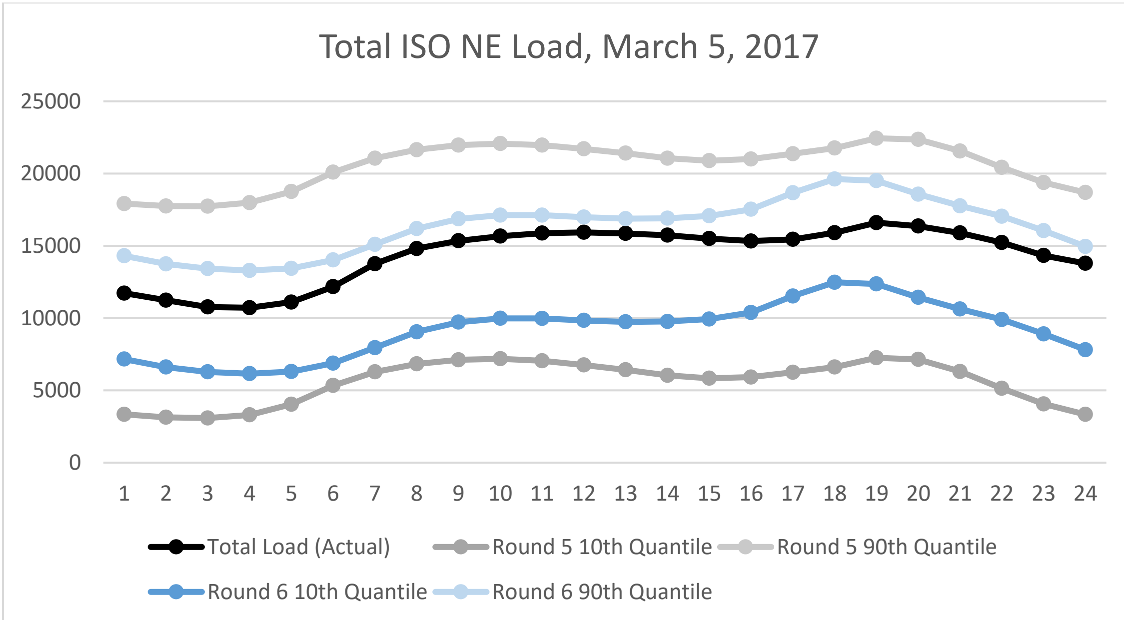

Figure 12. Quantile comparison between round 5 and round 6.

Conclusion

If I were to continue forecasting and analyzing, I would add in the demand from 2017 that was known, from the months leading up to the forecast period. I would also try and streamline my code and look into the data errors, so that I could use more years of data than just 2015 and 2016. I think it would be interesting to analyze trends over specific areas/states, and would maybe consider different models for different regions.

This course and the competition were a crash course in Forecasting and in R. It was a method to learn concepts by doing them and throwing myself into the fire. I feel that I usually excel at this, as my job puts me in a sink or swim environment quite often, but I do not know if this was the most effective way for me to learn forecasting. I would have benefited a lot from another couple hours of class time at the beginning to introduce and help me brush up on statistical analysis and forecasting. At the beginning, although I did research, it was very hard to even know what direction to go in, when it seemed like there were so many different possibilities. It felt like the course was structured to allow the freedom to explore those different possibilities, which I appreciated, but I felt like it was so much more of a trial and error process than actually educating myself on something and then going in that direction. After meeting with Dr. Balkan at the end, I realized I probably should have reached out more throughout the course, but again, without really knowing what to ask, I didn't think to schedule time. I'm more of a come-with-solutions person than a come-with-problems person, so unless I had something specific to tackle, I didn't reach out for help.

From this course, I learned the basics of forecasting, such as the different parts of a model: the seasonality, trend, and residual components. I also learned how quantiles worked and how they are used to evaluate forecasts. I was introduced to different algorithms by other students in the class:

neural networks, which I had heard of before, and support vector machines, which I had not. I was re-introduced to data analytics and statistical concepts like normal distribution and standard deviation.

Going forward, I would like to learn more about energy pricing structures and the market itself.

References

[1] ISO New England, "Key Grid and Market Stats", *Iso-ne.com*, 2017. Accessed on: 20- Mar- 2017. [Online]. Available: https://www.iso-ne.com/about/key-stats.

[2] R. Davis, "BPA Forecasting", Portland State University ETM Department, 2017.

[3] R. Hyndman, “Forecasting Functions for Time Series and Linear Models,” R Documentation Version 7.3. [Online] Available: http://github.com/robjhyndman/forecast

[4] R Core Team and Contributors Worldwide, “R Statistical Functions,” R Documentation Version 3.4.0, [Online]. Available: https://stat.ethz.ch/R-manual/R-devel/library/stats/html/stats-package.html

[5] R. Hyndman, "Generating quantile forecasts in R", *Hyndsight*, 2017. Accessed on: 20- Mar- 2017. [Online]. Available: http://robjhyndman.com/hyndsight/quantile-forecasts-in-r/.

[6] Rob J. Hyndman and George Athanasopoulos, *Forecasting: Principles and Practice,* Otexts. 2013, October. [Online]. https://www.otexts.org/fpp

[7] R.J. Hyndman, “Forecasting Time Series with Multiple Seasonal Patterns,” Accessed on: 20 – Mar- 2017, [Online]. Available: http://www.robjhyndman.com/papers/multiseasonal.pdf

[8] H. Tao, "GEFCom2017: Hierarchical Probabilistic Load Forecasting", *Blog.drhongtao.com*, 2017. Accessed on: 20- Mar- 2017. [Online]. Available: http://blog.drhongtao.com/2016/10/gefcom2017-hierarchical-probabilistic-load-forecasting.html.

Appendix A – Round 4 Code

```
## Load packages
library(XLConnect)
library(dplyr)
library(tibble)

## Load workbook and create numeric vector
wk1 = loadWorkbook("Misc/510/smd_hourly_2016.xls")
wk2 = loadWorkbook("Misc/510/smd_hourly_2015.xls")
## outwk <- loadWorkbook("Misc/510/TrackDRound4-PSU ETM 510.xls")
ME_2016 <- select(as_tibble(readWorksheet(wk1,sheet="ME")),Date, Hr_End, RT_Demand, Dry_Bulb)
ME_2015 <- select(as_tibble(readWorksheet(wk2,sheet="ME")),Date, Hour, DEMAND, DryBulb)

##read in variables
Demand_ME_2016 <- as.numeric(unlist(ME_2016[3]))
Demand_ME_2015 <- as.numeric(unlist(ME_2015[3]))
Total_Demand <- c(Demand_ME_2015, Demand_ME_2016)
##Date_2016 <- as.POSIXct(unlist(ME_2016[1]))   need origin?
##Dry_bulb_2016 <- as.numeric(unlist(ME_2016[4]))
##Hr_2016 <- as.numeric(unlist(ME_2016[2]))

## remove workbook once it is read in
rm(wk1)
rm(wk2)

## Using the entire year to predict the model
## ts_ME_2016 <- ts(data = Demand_ME_2016, start = 1, end = 366, frequency = 24)
## plot(ts_ME_2016)
## ts_ME_2016.ar <-ar(ts_ME_2016, aic = TRUE, order.max = NULL, method=c("burg"))
## ts_ME_2016.ar <-arima(ts_ME_2016)
## ts_ME_2016.predict <- predict(ts_ME_2016.ar,n.ahead = 8760)
## ts.plot(ts_ME_2016, ts_ME_2016.predict$pred)
## March = 31 days in Jan, 28 in Feb, 24 hours a day = 1416 through 2160
## March50_ME_2016 <- ts_ME_2016.predict$pred[1416:2160]

## Using just March to predict
Demand_ME_2016.march <- Demand_ME_2016[1441:2184]
ts_ME_2016 <- ts(data = Demand_ME_2016.march, start = 1, end = 31, frequency = 24)
plot(ts_ME_2016)
ts_ME_2016.ar <- ar(ts_ME_2016, aic = TRUE, order.max = NULL, method=c("burg"))
## ts_ME_2016.ar <- arima(ts_ME_2016)
ts_ME_2016.predict <- predict(ts_ME_2016.ar, n.ahead = 744)
## ts.plot(ts_ME_2016, ts_ME_2016.predict$pred)
##scatterplot3d(Hr_2016,Dry_bulb_2016,Demand_ME_2016,pch = 16, highlight.3d=TRUE)
March50_ME_2016 <- ts_ME_2016.predict$pred

##Quantiles
##quantile <- qnorm((10:90)/10, mean_Demand, std_Demand)


## Using March 50 and adding to either side, not to exceed minimum and maximum of previous year
March10_ME_2016 <- March50_ME_2016 * .8
March20_ME_2016 <- March50_ME_2016 * .85
March30_ME_2016 <- March50_ME_2016 * .9
March40_ME_2016 <- March50_ME_2016 * .95
March60_ME_2016 <- March50_ME_2016 * 1.05
March70_ME_2016 <- March50_ME_2016 * 1.1
March80_ME_2016 <- March50_ME_2016 * 1.15
March90_ME_2016 <- March50_ME_2016 * 1.2

## Write to workbook

writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March10_ME_2016, sheet = "ME", startRow = 2, startCol = 3)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March20_ME_2016, sheet = "ME", startRow = 2, startCol = 4)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March30_ME_2016, sheet = "ME", startRow = 2, startCol = 5)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March40_ME_2016, sheet = "ME", startRow = 2, startCol = 6)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March50_ME_2016, sheet = "ME", startRow = 2, startCol = 7)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March60_ME_2016, sheet = "ME", startRow = 2, startCol = 8)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March70_ME_2016, sheet = "ME", startRow = 2, startCol = 9)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March80_ME_2016, sheet = "ME", startRow = 2, startCol = 10)
writeWorksheetToFile("Misc/510/TrackDRound4-PSU ETM 510.xls", March90_ME_2016, sheet = "ME", startRow = 2, startCol = 11)
```

Appendix B – Round 5 Code

```
## Load packages
library(XLConnect)
library(dplyr)
library(tibble)
library(forecast)

## Load workbook and create numeric vector
wk1 = loadWorkbook("Misc/510/smd_hourly_2016.xls")
wk2 = loadWorkbook("Misc/510/smd_hourly_2015.xls")
data_2016 <- select(as_tibble(readWorksheet(wk1,sheet="CT")),Date, Hr_End, RT_Demand, Dry_Bulb)
data_2015 <- select(as_tibble(readWorksheet(wk2,sheet="CT")),Date, Hour, DEMAND, DryBulb)

##read in variables
Demand_2016 <- as.numeric(unlist(data_2016[3]))
Demand_2015 <- as.numeric(unlist(data_2015[3]))
##Date_2016 <- as.POSIXct(unlist(CT_2016[1]))   need origin?
##Dry_bulb_2016 <- as.numeric(unlist(CT_2016[4]))
##Hr_2016 <- as.numeric(unlist(CT_2016[2]))

## remove workbook once it is read in
rm(wk1)
rm(wk2)
rm(data_2015)
rm(data_2016)

## Using the entire year to predict the model
## ts_CT_2016 <- ts(data = Demand_CT_2016, start = 1, end = 366, frequency = 24)
## plot(ts_CT_2016)
## ts_CT_2016.ar <-ar(ts_CT_2016, aic = TRUE, order.max = NULL, method=c("burg"))
## ts_CT_2016.ar <-arima(ts_CT_2016)
## ts_CT_2016.predict <- predict(ts_CT_2016.ar,n.ahead = 8760)
## ts.plot(ts_CT_2016, ts_CT_2016.predict$pred)
## March = 31 days in Jan, 28 in Feb, 24 hours a day = 1416 through 2160
## March50_CT_2016 <- ts_CT_2016.predict$pred[1416:2160]

## Using just March to predict
Total_Demand <- c(Demand_2015[2185:2904], Demand_2016[2161:2880])

## make sure there are no zeros (month)
for (i in 1:1440)
  if (Total_Demand[i] <= 0 ) Total_Demand[i] <- Total_Demand[i-1]


ts_Demand <- ts(data = Total_Demand, start = 1, end = 62, frequency = 24)

fcast_demand <- forecast(ts_Demand, h = 744, robust = TRUE, find.frequency = TRUE)

## obtaining quantiles with standard deviation
results <- matrix(0, nrow=81, ncol=720)
m <- fcast_demand$mean
s <- (fcast_demand$upper-fcast_demand$lower)/1.96/2

for(h in 1:720)
  results[,h] <- qnorm((10:90)/100, m[h], s[h])

for(h in 1:720)
  if (results[1,h] <= 0 ) results[1,h] <- results[10,h]

## pulling only needed quantiles
March10 <- results[1, 1:720]
March20 <- results[11,1:720]
March30 <- results[21,1:720]
March40 <- results[31,1:720]
March50 <- results[41,1:720]
March60 <- results[51,1:720]
March70 <- results[61,1:720]
March80 <- results[71,1:720]
March90 <- results[81,1:720]

## write to workbook

writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March10, sheet = "CT", startRow = 2, startCol = 3)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March20, sheet = "CT", startRow = 2, startCol = 4)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March30, sheet = "CT", startRow = 2, startCol = 5)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March40, sheet = "CT", startRow = 2, startCol = 6)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March50, sheet = "CT", startRow = 2, startCol = 7)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March60, sheet = "CT", startRow = 2, startCol = 8)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March70, sheet = "CT", startRow = 2, startCol = 9)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March80, sheet = "CT", startRow = 2, startCol = 10)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March90, sheet = "CT", startRow = 2, startCol = 11)
```

Appendix C – Round 6 Code

Using April from 2015 and 2016

```
## Load packages
library(XLConnect)
library(dplyr)
library(tibble)
library(forecast)

## Load workbook and create numeric vector
wk1 = loadWorkbook("Misc/510/smd_hourly_2016.xls")
wk2 = loadWorkbook("Misc/510/smd_hourly_2015.xls")
data_2016 <- select(as_tibble(readWorksheet(wk1,sheet="VT")),Date, Hr_End, RT_Demand, Dry_Bulb)
data_2015 <- select(as_tibble(readWorksheet(wk2,sheet="VT")),Date, Hour, DEMAND, DryBulb)

##read in variables
Demand_2016 <- as.numeric(unlist(data_2016[3]))
Demand_2015 <- as.numeric(unlist(data_2015[3]))

## remove workbook once it is read in
rm(wk1)
rm(wk2)
rm(data_2015)
rm(data_2016)

## Using two years of April to predict
Total_Demand <- c(Demand_2015[2185:2904], Demand_2016[2161:2880])

## make sure there are no zeros (month)
for (i in 1:1440)
  if (Total_Demand[i] <= 0 ) Total_Demand[i] <- Total_Demand[i-1]

##using two years of April to predict
demand.multi <- msts(Total_Demand, c(24, 168))
double_season <- dshw(demand.multi, h=720)

## obtaining quantiles with standard deviation
results <- matrix(0, nrow=81, ncol=720)
m <- double_season$mean
stand_dev <- sd(Total_Demand)
for(h in 1:720)
  results[,h] <- qnorm((10:90)/100, m[h], sd = stand_dev)

for(h in 1:720)
  if (results[1,h] <= 0 ) results[1,h] <- results[10,h]

## pulling only needed quantiles
March10 <- results[1, 1:720]
March20 <- results[11,1:720]
March30 <- results[21,1:720]
March40 <- results[31,1:720]
March50 <- results[41,1:720]
March60 <- results[51,1:720]
March70 <- results[61,1:720]
March80 <- results[71,1:720]
March90 <- results[81,1:720]

## Write to workbook

writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March10, sheet = "VT", startRow = 2, startCol = 3)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March20, sheet = "VT", startRow = 2, startCol = 4)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March30, sheet = "VT", startRow = 2, startCol = 5)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March40, sheet = "VT", startRow = 2, startCol = 6)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March50, sheet = "VT", startRow = 2, startCol = 7)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March60, sheet = "VT", startRow = 2, startCol = 8)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March70, sheet = "VT", startRow = 2, startCol = 9)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March80, sheet = "VT", startRow = 2, startCol = 10)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March90, sheet = "VT", startRow = 2, startCol = 11)
```

Using 2016 Data

```r
## Load packages
library(XLConnect)
library(dplyr)
library(tibble)
library(forecast)

## Load workbook and create numeric vector
wk1 = loadWorkbook("Misc/510/smd_hourly_2016_season.xls")
data_2016 <- select(as_tibble(readWorksheet(wk1,sheet="NEMA")),Date, Hr_End, RT_Demand, Dry_Bulb)

##read in variables
Demand_2016 <- as.numeric(unlist(data_2016[3]))

## remove workbook once it is read in
rm(wk1)
rm(data_2016)

## remove zeros (if using year data)
for (i in 1:8784)
 if (Demand_2016[i] <= 0) Demand_2016[i] <- Demand_2016[i-1]

##using 2016 to predict
multi_2016 <- msts(Demand_2016, c(24,2160))
dseason_2016 <-dshw(multi_2016, h = 2880)

## obtaining quantiles with standard deviation
results <- matrix(0, nrow=81, ncol=720)
m <- dseason_2016$mean[2161:2880]
stand_dev <- sd(Demand_2016)

for(h in 1:720)
  results[,h] <- qnorm((10:90)/100, m[h], sd = stand_dev)

for(h in 1:720)
  if (results[1,h] <= 0 ) results[1,h] <- results[10,h]

## pulling only needed quantiles
March10 <- results[1, 1:720]
March20 <- results[11,1:720]
March30 <- results[21,1:720]
March40 <- results[31,1:720]
March50 <- results[41,1:720]
March60 <- results[51,1:720]
March70 <- results[61,1:720]
March80 <- results[71,1:720]
March90 <- results[81,1:720]

## write to workbook

writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March10, sheet = "NEMASSBOST", startRow = 2, startCol = 3)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March20, sheet = "NEMASSBOST", startRow = 2, startCol = 4)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March30, sheet = "NEMASSBOST", startRow = 2, startCol = 5)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March40, sheet = "NEMASSBOST", startRow = 2, startCol = 6)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March50, sheet = "NEMASSBOST", startRow = 2, startCol = 7)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March60, sheet = "NEMASSBOST", startRow = 2, startCol = 8)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March70, sheet = "NEMASSBOST", startRow = 2, startCol = 9)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March80, sheet = "NEMASSBOST", startRow = 2, startCol = 10)
writeWorksheetToFile("Misc/510/D6-PSU ETM 510.xls", March90, sheet = "NEMASSBOST", startRow = 2, startCol = 11)
```