Brown Datathon 2018 - RI Electricity

Thoa Ta

March 3, 2018

```
# include the necessary libraries
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages ----------------------------------------------
-
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
library(dplyr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
# set all the plots in this document to white background
theme_set(theme_bw())
```

```
# load & check data
mydata <- read_csv("my_copy_of_2017_smd_hourly.csv")
```

```
## Parsed with column specification:
## cols(
##   Date = col_character(),
##   Hr_End = col_character(),
##   DA_Demand = col_number(),
##   RT_Demand = col_number(),
##   DA_LMP = col_double(),
##   DA_EC = col_double(),
##   DA_CC = col_double(),
##   DA_MLC = col_double(),
##   RT_LMP = col_double(),
##   RT_EC = col_double(),
##   RT_CC = col_double(),
##   RT_MLC = col_double(),
##   Dry_Bulb = col_character(),
##   Dew_Point = col_character()
## )
```

```
mydata <- mydata %>% select(-13,-14)
head(mydata)
```

```
## # A tibble: 6 x 12
##       Date Hr_End DA_Demand RT_Demand DA_LMP DA_EC DA_CC DA_MLC RT_LMP
##      <chr>  <chr>     <dbl>     <dbl>  <dbl> <dbl> <dbl>  <dbl>  <dbl>
## 1 1-Jan-17     01     721.1    752.17  35.27 34.86     0   0.41  36.91
## 2 1-Jan-17     02     688.3    716.89  34.09 33.72     0   0.37  37.47
## 3 1-Jan-17     03     667.8    692.00  32.74 32.40     0   0.34  36.19
## 4 1-Jan-17     04     659.1    677.66  26.15 25.88     0   0.27  34.43
## 5 1-Jan-17     05     662.9    673.28  29.95 29.65     0   0.30  35.76
## 6 1-Jan-17     06     681.3    680.18  32.89 32.54     0   0.35  34.71
## # ... with 3 more variables: RT_EC <dbl>, RT_CC <dbl>, RT_MLC <dbl>
```

```
summary(mydata)
```

```
##      Date              Hr_End            DA_Demand        RT_Demand     
##  Length:8760        Length:8760        Min.   : 527.5   Min.   : 458.8  
##  Class :character   Class :character   1st Qu.: 758.0   1st Qu.: 764.3  
##  Mode  :character   Mode  :character   Median : 893.1   Median : 891.1  
##                                        Mean   : 902.4   Mean   : 896.9  
##                                        3rd Qu.:1008.5   3rd Qu.: 993.3  
##                                        Max.   :1915.1   Max.   :1703.5  
##      DA_LMP           DA_EC            DA_CC               DA_MLC       
##  Min.   :  1.03   Min.   :  1.02   Min.   :-14.09000   Min.   :-1.9200  
##  1st Qu.: 21.49   1st Qu.: 21.52   1st Qu.:  0.00000   1st Qu.:-0.3600  
##  Median : 27.74   Median : 27.93   Median :  0.00000   Median :-0.1200  
##  Mean   : 33.14   Mean   : 33.25   Mean   :  0.01076   Mean   :-0.1205  
##  3rd Qu.: 36.71   3rd Qu.: 37.16   3rd Qu.:  0.02000   3rd Qu.: 0.1000  
##  Max.   :235.12   Max.   :230.34   Max.   : 40.80000   Max.   : 4.7800  
##      RT_LMP            RT_EC             RT_CC              RT_MLC       
##  Min.   :-126.54   Min.   :-126.56   Min.   :-50.0800   Min.   :-6.9200  
##  1st Qu.:  19.65   1st Qu.:  19.55   1st Qu.:  0.0000   1st Qu.:-0.3000
```

```
##  Median :  26.00   Median :  26.07   Median :  0.0000   Median :-0.1000
##  Mean   :  33.79   Mean   :  33.73   Mean   :  0.1668   Mean   :-0.1117
##  3rd Qu.:  38.16   3rd Qu.:  38.26   3rd Qu.:  0.0300   3rd Qu.: 0.0800
##  Max.   : 690.08   Max.   : 697.00   Max.   : 33.4900   Max.   : 4.1600
```

```
# check missing data: all zero sums means no missing data
colSums(is.na(mydata))
```

```
##      Date    Hr_End DA_Demand RT_Demand    DA_LMP     DA_EC     DA_CC
##         0         0         0         0         0         0         0
##    DA_MLC    RT_LMP     RT_EC     RT_CC    RT_MLC
##         0         0         0         0         0
```

```
# check correlation
cor(mydata[,3:12])
```

```
##            DA_Demand  RT_Demand      DA_LMP       DA_EC        DA_CC
## DA_Demand  1.0000000  0.9707430 0.393505736  0.40532618 -0.109613760
## RT_Demand  0.9707430  1.0000000 0.406719411  0.41856035 -0.122855435
## DA_LMP     0.3935057  0.4067194 1.000000000  0.99910104  0.007544756
## DA_EC      0.4053262  0.4185603 0.999101037  1.00000000 -0.030517143
## DA_CC     -0.1096138 -0.1228554 0.007544756 -0.03051714  1.000000000
## DA_MLC    -0.2855598 -0.2588358 0.316826810  0.29481915  0.110744742
## RT_LMP     0.2769773  0.3188725 0.763637290  0.76537470 -0.030445394
## RT_EC      0.2912600  0.3339789 0.759860053  0.76315870 -0.062249238
## RT_CC     -0.1221051 -0.1360975 0.002446036 -0.01319854  0.372729344
## RT_MLC    -0.2586963 -0.2349475 0.304172731  0.28510927  0.107583275
##               DA_MLC      RT_LMP       RT_EC        RT_CC      RT_MLC
## DA_Demand -0.2855598  0.27697728  0.29126002 -0.122105143 -0.25869626
## RT_Demand -0.2588358  0.31887248  0.33397889 -0.136097533 -0.23494747
## DA_LMP     0.3168268  0.76363729  0.75986005  0.002446036  0.30417273
## DA_EC      0.2948191  0.76537470  0.76315870 -0.013198540  0.28510927
## DA_CC      0.1107447 -0.03044539 -0.06224924  0.372729344  0.10758327
## DA_MLC     1.0000000  0.18524925  0.16302446  0.114718783  0.85358550
## RT_LMP     0.1852493  1.00000000  0.99648145  0.046127631  0.08286199
## RT_EC      0.1630245  0.99648145  1.00000000 -0.036341486  0.05805579
## RT_CC      0.1147188  0.04612763 -0.03634149  1.000000000  0.11667351
## RT_MLC     0.8535855  0.08286199  0.05805579  0.116673506  1.00000000
```

We see that the following pairs have the highest correlations:

Variable A	Variable B	Correlation
RT_Demand	DA_Demand	.97
DA_EC	DA_LMP	.99
RT_EC	RT_LMP	.99

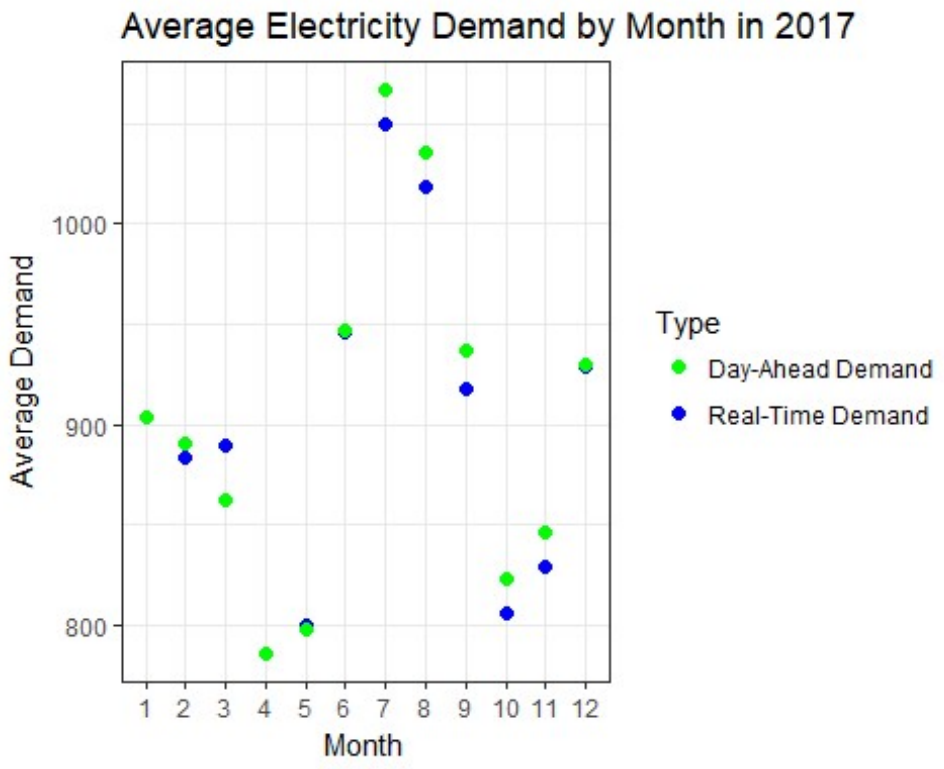
Since I don't have background in the electricity market, variables other than the Real-Time Demand and the Day-Ahead Demand make little sense to me. Hence, I chose to work on the Demand variables only.

Electricity demand by month

```
# parse the Date column
mydata$Date <- dmy(mydata$Date)
head(mydata)
```

```
## # A tibble: 6 x 12
##         Date Hr_End DA_Demand RT_Demand DA_LMP DA_EC DA_CC DA_MLC RT_LMP
##       <date>  <chr>     <dbl>     <dbl>  <dbl> <dbl> <dbl>  <dbl>  <dbl>
## 1 2017-01-01     01     721.1    752.17  35.27 34.86     0   0.41  36.91
## 2 2017-01-01     02     688.3    716.89  34.09 33.72     0   0.37  37.47
## 3 2017-01-01     03     667.8    692.00  32.74 32.40     0   0.34  36.19
## 4 2017-01-01     04     659.1    677.66  26.15 25.88     0   0.27  34.43
## 5 2017-01-01     05     662.9    673.28  29.95 29.65     0   0.30  35.76
## 6 2017-01-01     06     681.3    680.18  32.89 32.54     0   0.35  34.71
## # ... with 3 more variables: RT_EC <dbl>, RT_CC <dbl>, RT_MLC <dbl>
```

```
mydata %>%
  mutate(Month = as.factor(month(Date))) %>%
  group_by(Month) %>%
  summarize("Real-Time Demand" = mean(RT_Demand),
            "Day-Ahead Demand" = mean(DA_Demand)) %>%
  gather(2:3, key = Type, value = Mean_Demand) %>%
  ggplot(aes(x = Month, y = Mean_Demand, color = Type)) +
  geom_point(size = 2) +
  scale_color_manual(values = c("green", "blue")) +
  labs(title = "Average Electricity Demand by Month in 2017", y = "Average
Demand")
```



This plot shows us two things:

1. How close the average day-ahead and real-time demand are in some months, and
2. The demands in different months throughout a year.

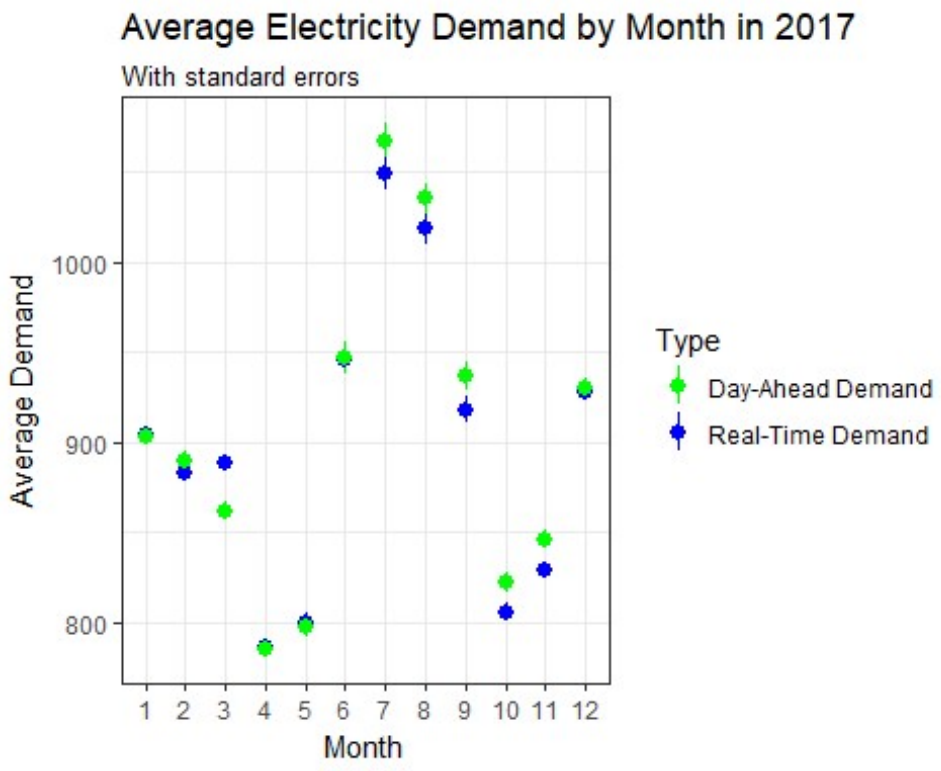
We will further examine these two observations below.

1. Variance in demand and day-ahead / real-time correlation

I wonder if the variance in monthly demand could be indicative of the day-ahead / real-time correlation. In other words, can we say that day-ahead and real-time values are less correlated in months with higer variance?

```
mydata %>%
  mutate(Month = as.factor(month(Date))) %>%
  group_by(Month) %>%
  summarize("Real-Time Demand" = mean(RT_Demand),
            "Day-Ahead Demand" = mean(DA_Demand),
            "Real-Time Std.Err" = sd(RT_Demand) / sqrt(n()),
            "Day-Ahead Std.Err" = sd(DA_Demand) / sqrt(n())) %>%
  gather(2:3, key = Demand_Type, value = Mean_Value) %>%
  gather(2:3, key = Std.Err_Type, value = Std.Err_Value) %>%
  arrange(Month) %>%
  filter((Demand_Type == "Real-Time Demand" & Std.Err_Type == "Real-Time
Std.Err") |
           (Demand_Type == "Day-Ahead Demand" & Std.Err_Type == "Day-Ahead
Std.Err")) %>%
```

```
  ggplot(aes(x = Month, y = Mean_Value, color = Demand_Type)) +
  geom_pointrange(aes(ymin = Mean_Value - Std.Err_Value, ymax = Mean_Value +
Std.Err_Value)) +
  scale_color_manual(values = c("green", "blue")) +
  labs(title = "Average Electricity Demand by Month in 2017",
       subtitle = "With standard errors",
       y = "Average Demand",
       color = "Type")
```



It seems like the relationship is not that strong.

2. Demand trend throughout the year

We see that electricity demand is the highest in July and August; relatively high in December, January, February, March, June, and September; and lowest in April, May, October, and November. Possible explanations are as follows:

- July and August are the peak of summer, so electricity for running the air-conditioner is high.
- April-May and October-November are the transition months from a cold season to a hot one (and vice versa), so users might simply turn off the heater and not (yet to / need to) turn on the air-conditioner. Therefore, electricity demand is low.
- June and September are the pre and post of summer time, so electricity demand is slightly, but not drastically, higher than usual.

- The other months show the average electricity need for heaters during winter time.

One interesting point for future research would be: consider what types of appliances are used by the majority of users *(probably both industrial and residential users?)* in the summer and the winter seasons, to verify the demand for electricity.

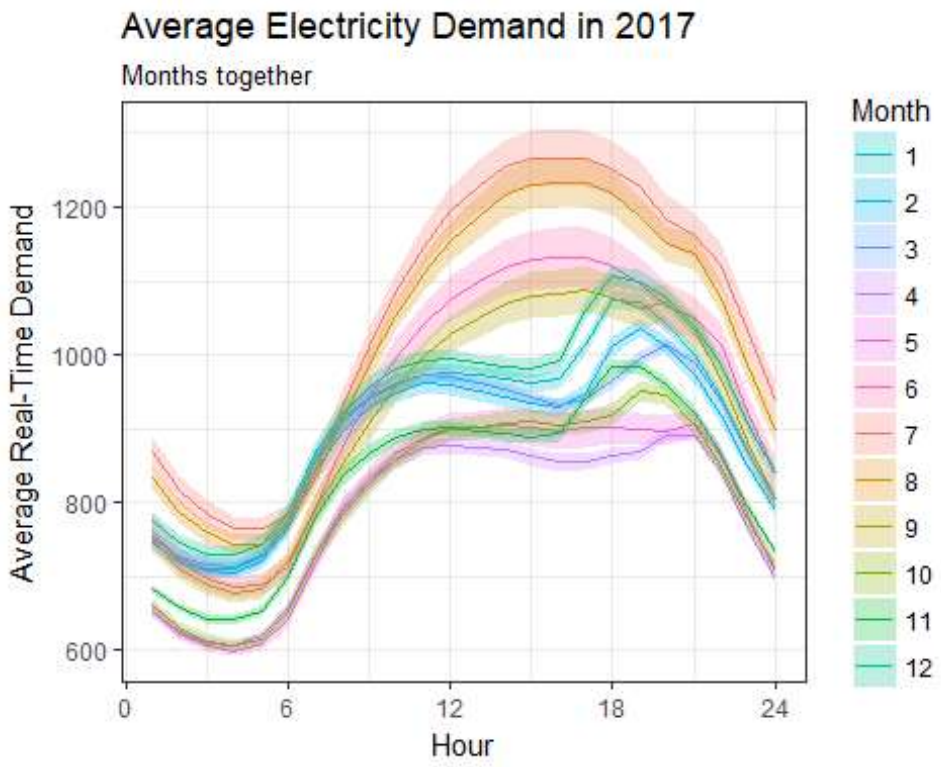
Electricity demand by time of day

Next up, I want to examine the demand trend throughout the course of a day, faceted by month.

Note Since there is not a lot of difference between the Real-Time and Day-Ahead Demand, I will just use the Real-Time Demand for the following analysis.

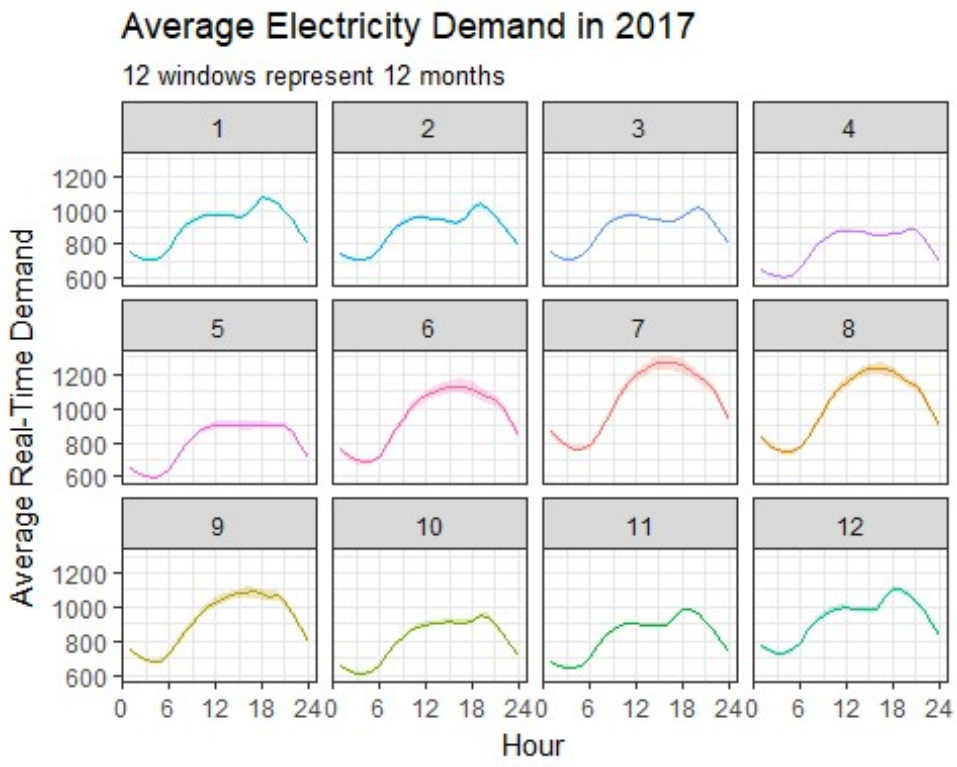
```
# by default, the color wheel starts with hot colors at 2 ends. We want to
shift that by 180 degrees
# so that cold colors can align with cold months and vice versa.
mycolorshift <- hue_pal(h.start = 180)(12) # stores an array of 12 hue
shades, to be used for 12 months

mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  #filter(Month %in% c("12", "1", "2", "3")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
                  fill = Month),
              alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_manual(values = mycolorshift) +
  scale_fill_manual(values = mycolorshift) +
  labs(title = "Average Electricity Demand in 2017",
       subtitle = "Months together",
       y = "Average Real-Time Demand")
```



To see the trend in each month more clearly, I am going to facet the data into month windows.

```
mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  #filter(Month %in% c("12", "1", "2", "3")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
                  fill = Month),
              alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_discrete(h.start = 180) +
  scale_fill_discrete(h.start = 180) +
  facet_wrap( ~ Month) +
  theme(legend.position = 'none') +
  labs(title = "Average Electricity Demand in 2017",
       subtitle = "12 windows represent 12 months",
       y = "Average Real-Time Demand")
```



From the two plots above, we observe that :

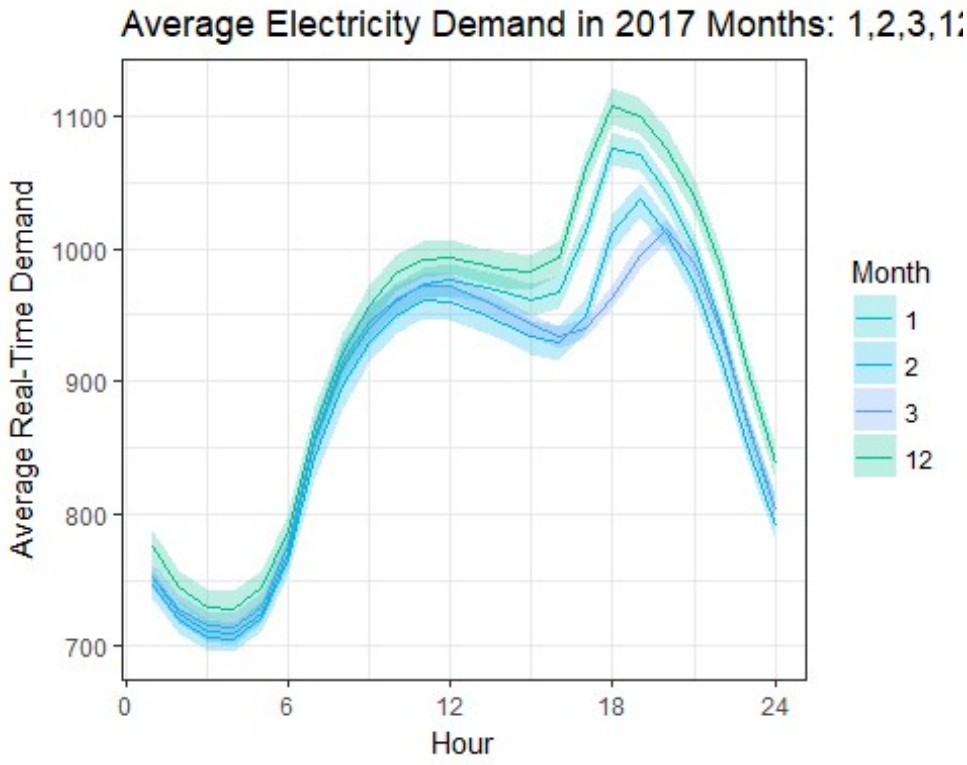
1. The timewise trends group themselves into five (5) groups of pattern:
    a. December to March
    b. April and May
    c. June and September
    d. July and August
    e. October and November
2. All 5 groups have similar lowest points of demand (the 3-5am range), but their peak patterns differ.

Below, we will look more closely at each group to examine their peak pattern.

December to March

```
mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  filter(Month %in% c("12", "1", "2", "3")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
```

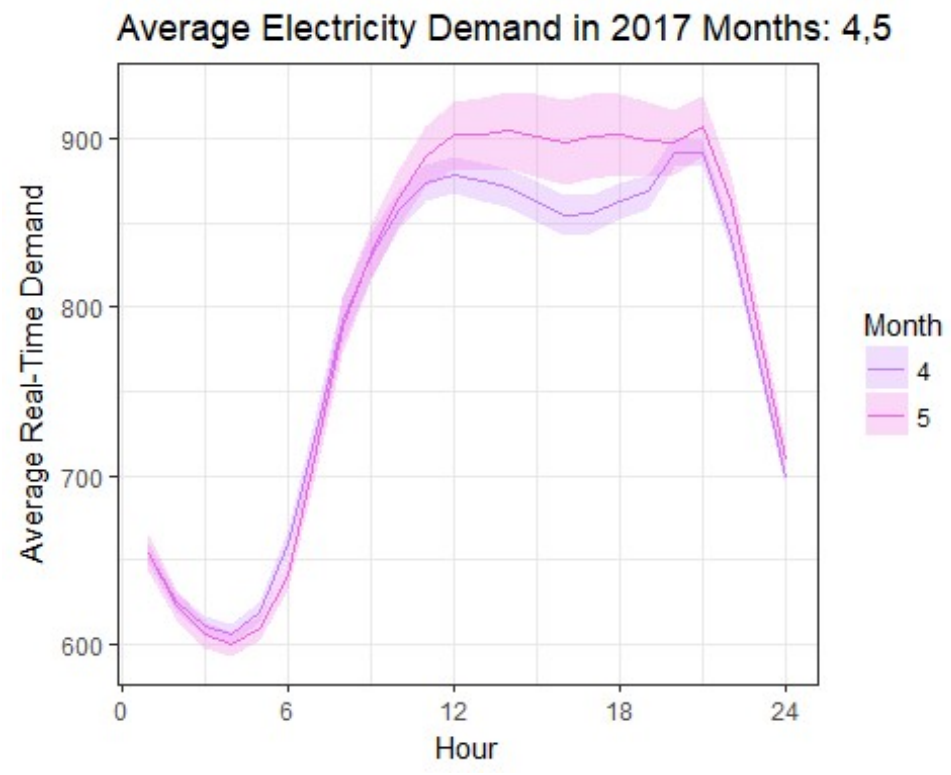
```
                fill = Month),
            alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_manual(values = c(mycolorshift[1], mycolorshift[2],
mycolorshift[3], mycolorshift[12])) +
  scale_fill_manual(values = c(mycolorshift[1], mycolorshift[2],
mycolorshift[3], mycolorshift[12])) +
  labs(title = "Average Electricity Demand in 2017 Months: 1,2,3,12",
       y = "Average Real-Time Demand")
```



April and May

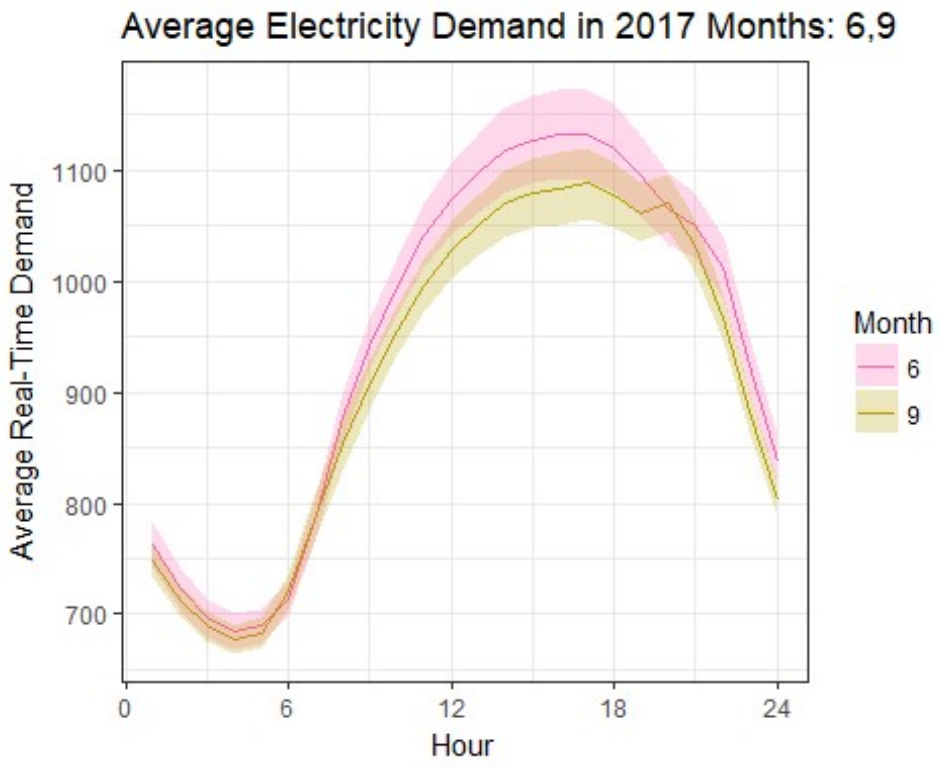
```
mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  filter(Month %in% c("4", "5")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
                  fill = Month),
              alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_manual(values = mycolorshift[4:5]) +
```

```
  scale_fill_manual(values = mycolorshift[4:5]) +
  labs(title = "Average Electricity Demand in 2017 Months: 4,5",
       y = "Average Real-Time Demand")
```



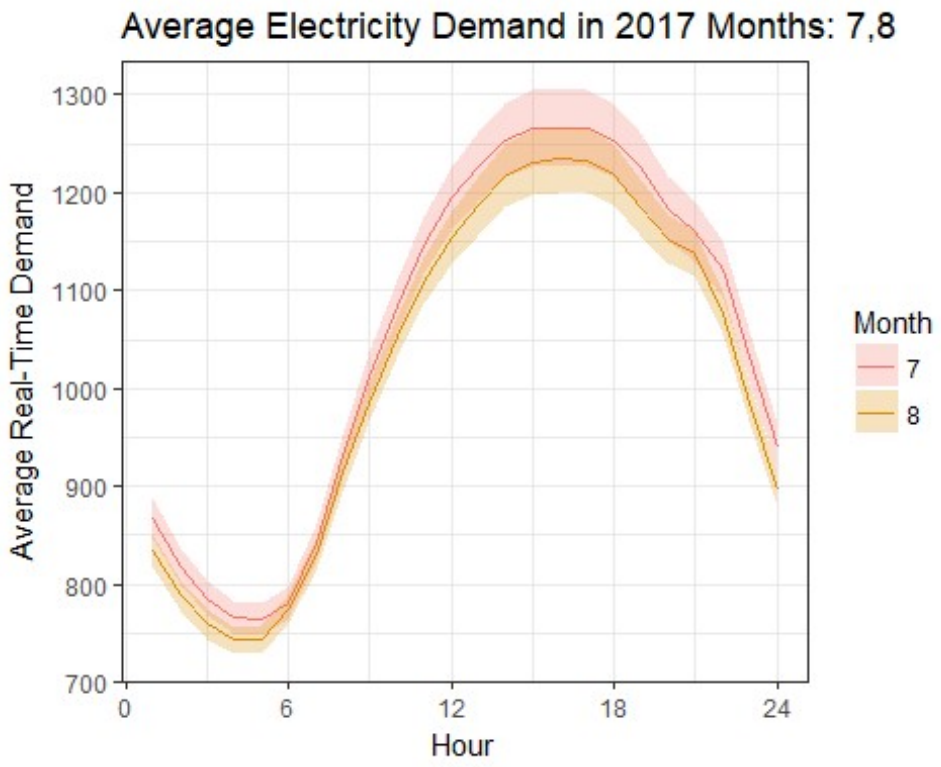
June and September

```
mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  filter(Month %in% c("6", "9")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
                  fill = Month),
              alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_manual(values = c(mycolorshift[6], mycolorshift[9])) +
  scale_fill_manual(values = c(mycolorshift[6], mycolorshift[9])) +
#"#00C08B", "#619CFF")) +
  labs(title = "Average Electricity Demand in 2017 Months: 6,9",
       y = "Average Real-Time Demand")
```



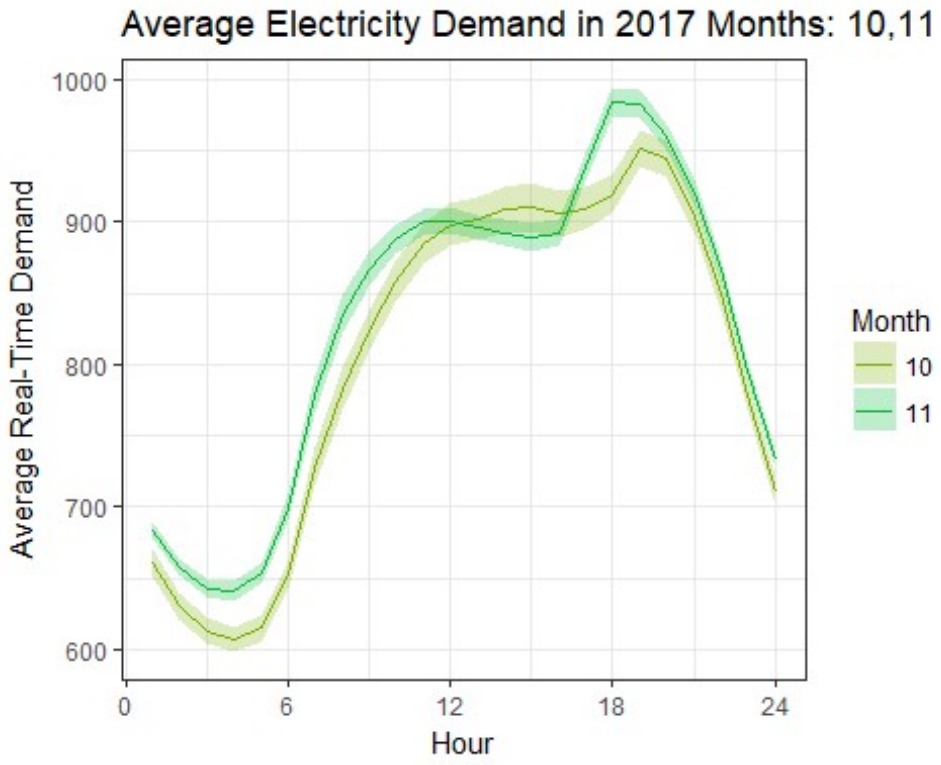
July and August

```
mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  filter(Month %in% c("7", "8")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
                  fill = Month),
              alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_manual(values = mycolorshift[7:8]) +
  scale_fill_manual(values = mycolorshift[7:8]) +
  labs(title = "Average Electricity Demand in 2017 Months: 7,8",
       y = "Average Real-Time Demand")
```



October and Novemeber

```
mydata %>%
  mutate(Month = as.factor(month(Date)), Hour = as.numeric(Hr_End)) %>%
  select(-Hr_End) %>%
  filter(Month %in% c("10", "11")) %>%
  group_by(Month, Hour) %>%
  summarize(Mean_RT_Demand = mean(RT_Demand),
            Se_RT_Demand = sd(RT_Demand) / sqrt(n())) %>%
  ggplot(aes(x = Hour)) +
  geom_line(aes(y = Mean_RT_Demand, color = Month)) +
  geom_ribbon(aes(ymin = Mean_RT_Demand - Se_RT_Demand,
                  ymax = Mean_RT_Demand + Se_RT_Demand,
                  fill = Month),
              alpha = 0.25) +
  scale_x_continuous(breaks = seq(0, 24, by = 6)) +
  scale_color_manual(values = mycolorshift[10:11]) +
  scale_fill_manual(values = mycolorshift[10:11]) +
  labs(title = "Average Electricity Demand in 2017 Months: 10,11",
       y = "Average Real-Time Demand")
```



Next steps:

1. **Seasonal use of appliances** consider what types of appliances are used by the majority of users *(probably both industrial and residential users?)* in the summer and the winter seasons, to verify (and explain) the demand pattern for electricity.

2. **Integrate weather data** from some other engines or database to learn more about the daily, hourly, and monthly electricity use.