DATA 607 Extra Credit Week 11-Conrardy

Anthony Conrardy

2024-03-11

Extra Credit Assignment Week 11

1) Find a dataset that includes time series for two or more separate items. For example, you could use end of day stock or cryptocurrency prices since Jan 1, 2022 for several instruments.

2) Use window functions (in SQL or dplyr) to calculate the year-to-date average and the six-day moving averages for each item.

3) Present your code in a three to five minute presentation (or you may make a recording using screencastomatic or another tool).

Dataset Selected

The dataset I selected is an Hourly Energy Demand Generation file located on the Kaggle Platform at https://www.kaggle.com/datasets/nicholasjhana/energy-consumption-generation-prices-and-weather?select=energy_dataset.csv

I downloaded the dataset an placed within the GitHub repository for easier access. It is located at https://github.com/Aconrard/DATA607/tree/main/Extra%20Credit%20Week%2011

```
# Let us read in the data file.
energy_df <- read.csv("https://raw.githubusercontent.com/Aconrard/DATA607/main/Extra%20Credit%20Week%20
head(energy_df, 5)
```

```
##                         time generation.biomass
## 1 2015-01-01 00:00:00+01:00                447
## 2 2015-01-01 01:00:00+01:00                449
## 3 2015-01-01 02:00:00+01:00                448
## 4 2015-01-01 03:00:00+01:00                438
## 5 2015-01-01 04:00:00+01:00                428
##   generation.fossil.brown.coal.lignite generation.fossil.coal.derived.gas
## 1                                  329                                  0
## 2                                  328                                  0
## 3                                  323                                  0
## 4                                  254                                  0
## 5                                  187                                  0
##   generation.fossil.gas generation.fossil.hard.coal generation.fossil.oil
## 1                  4844                        4821                   162
## 2                  5196                        4755                   158
## 3                  4857                        4581                   157
## 4                  4314                        4131                   160
## 5                  4130                        3840                   156
```

```
##   generation.fossil.oil.shale generation.fossil.peat generation.geothermal
## 1                           0                     0                      0
## 2                           0                     0                      0
## 3                           0                     0                      0
## 4                           0                     0                      0
## 5                           0                     0                      0
##   generation.hydro.pumped.storage.aggregated
## 1                                         NA
## 2                                         NA
## 3                                         NA
## 4                                         NA
## 5                                         NA
##   generation.hydro.pumped.storage.consumption
## 1                                         863
## 2                                         920
## 3                                        1164
## 4                                        1503
## 5                                        1826
##   generation.hydro.run.of.river.and.poundage generation.hydro.water.reservoir
## 1                                       1051                             1899
## 2                                       1009                             1658
## 3                                        973                             1371
## 4                                        949                              779
## 5                                        953                              720
##   generation.marine generation.nuclear generation.other
## 1                 0                7096                43
## 2                 0                7096                43
## 3                 0                7099                43
## 4                 0                7098                43
## 5                 0                7097                43
##   generation.other.renewable generation.solar generation.waste
## 1                         73                49              196
## 2                         71                50              195
## 3                         73                50              196
## 4                         75                50              191
## 5                         74                42              189
##   generation.wind.offshore generation.wind.onshore forecast.solar.day.ahead
## 1                        0                    6378                       17
## 2                        0                    5890                       16
## 3                        0                    5461                        8
## 4                        0                    5238                        2
## 5                        0                    4935                        9
##   forecast.wind.offshore.eday.ahead forecast.wind.onshore.day.ahead
## 1                                NA                            6436
## 2                                NA                            5856
## 3                                NA                            5454
## 4                                NA                            5151
## 5                                NA                            4861
##   total.load.forecast total.load.actual price.day.ahead price.actual
## 1               26118              25385           50.10        65.41
## 2               24934              24382           48.10        64.92
## 3               23515              22734           47.33        64.48
## 4               22642              21286           42.27        59.32
## 5               21785              20264           38.41        56.04
```

Tidy and Transform Data

We can see that there are some structural changes that need to be done to this dataset before we are able to start answering the assignment. The time variable is tracked in hours each day. For hour purposes, we will only need the date portion of the time variable and then we can aggregate the hourly demand to be a daily demand. We will drop the time variable once the date has been extracted.

There are also a number of categories in this dataset, but we are only going to perform function on these five (5), which include: Biomass Fossil Brown Coal Fossil Gas Fossil Hard Coal Fossil Oil

It should also be noted that a "0", actually means not production and that "NA" means an absent value.

```
# Strip just the Date and Relocate
energy_df <- energy_df |> mutate(date_only = as.Date(time)) |> relocate(date_only)

# Select the Variables for
energy_df_select <- energy_df |> select(date_only,generation.biomass, generation.fossil.brown.coal.ligni


head(energy_df_select, 5)
```

```
##    date_only generation.biomass generation.fossil.brown.coal.lignite
## 1 2015-01-01                447                                  329
## 2 2015-01-01                449                                  328
## 3 2015-01-01                448                                  323
## 4 2015-01-01                438                                  254
## 5 2015-01-01                428                                  187
##   generation.fossil.hard.coal generation.fossil.oil generation.fossil.gas
## 1                        4821                   162                  4844
## 2                        4755                   158                  5196
## 3                        4581                   157                  4857
## 4                        4131                   160                  4314
## 5                        3840                   156                  4130
```

Extracting Daily Average Electricity Generation for Each Type of Fuel Source

```
# Biomass
daily_average_biomass <- energy_df_select |> group_by(date_only) |> summarize(daily_avg_biomass = round
# Brown Coal
daily_average_brown_coal <- energy_df_select |> group_by(date_only) |> summarize(daily_avg_brown_coal =
# Hard Coal
daily_average_hard_coal <- energy_df_select |> group_by(date_only) |> summarize(daily_avg_hard_coal = r
# Oil
daily_average_oil <- energy_df_select |> group_by(date_only) |> summarize(daily_avg_oil = round(mean(ge
# Gas
daily_average_gas <- energy_df_select |> group_by(date_only) |> summarize(daily_avg_gas = round(mean(ge

# Combine all the columns into one frame
daily_avgs <- cbind(daily_average_biomass, daily_average_brown_coal[,-1], daily_average_gas[,-1], daily
```

Year-to-Date (YTD) Averages

This particular data set runs from 2015 through 2018. For our purposes we will run from YTD averages for 2017. We will have to extract the year and day of year from the datasets, and then calculate the YTD averages for the different sources of electricity generation.

```
# Strip Year and day of year (doy) from date
daily_avgs <- daily_avgs|> mutate(year = lubridate::year(date_only),
                    doy = lubridate::yday(date_only)) |> relocate(doy) |> relocate(year)

ytd_avg_2017 <- daily_avgs |>
  filter(year==2017) |>
  group_by(year,doy) |>
  mutate(ytd_avg_biomass = round(cummean(daily_avg_biomass)),
         ytd_avg_brown_coal = round(cummean(daily_avg_brown_coal)),
         ytd_avg_gas = round(cummean(daily_avg_gas)),
         ytd_avg_hard_coal = round(cummean(daily_avg_hard_coal)),
         ytd_avg_oil = round(cummean(daily_avg_oil))) |>
  filter(doy == max(71)) |>
  select(year,doy,ytd_avg_biomass, ytd_avg_brown_coal, ytd_avg_gas, ytd_avg_hard_coal, ytd_avg_oil) |>
  unique()

knitr::kable(ytd_avg_2017)
```

year	doy	ytd_avg_biomass	ytd_avg_brown_coal	ytd_avg_gas	ytd_avg_hard_coal	ytd_avg_oil
2017	71	351	746	7142	6204	263

Six Day Moving Average

In this section we will calculate the six(6) moving average for the previously identified fuel sources. Since we already calculated the daily average for each of the fuel sources, we will apply the moving average to the year of 2017 and report the results in a plot. We will calculate the six_day averages for each of the fuel sources, but we are going to plot only one example, Biomass. However, it is similarly done for the other sources.

```
# Biomass Six Day
daily_average_biomass <-  daily_average_biomass |> filter(format(date_only, "%Y") == "2017") |>
  mutate(biomass_six_day = round(zoo::rollmean(daily_avg_biomass, k = 6, fill = NA)))

## Brown Coal Six Day
daily_average_brown_coal <- daily_average_brown_coal |> filter(format(date_only, "%Y") == "2017") |>
  mutate(brown_coal_six_day =round(zoo::rollmean(daily_avg_brown_coal, k = 6, fill = NA)))

## Gas Six Day
daily_average_gas <- daily_average_gas |> filter(format(date_only, "%Y") == "2017") |>
  mutate(gas_six_day =round(zoo::rollmean(daily_avg_gas, k = 6, fill = NA)))

## Hard Coal Six Day
daily_average_hard_coal <- daily_average_hard_coal |> filter(format(date_only, "%Y") == "2017") |>
  mutate(hard_cola_six_day =round(zoo::rollmean(daily_avg_hard_coal, k = 6, fill = NA)))
```

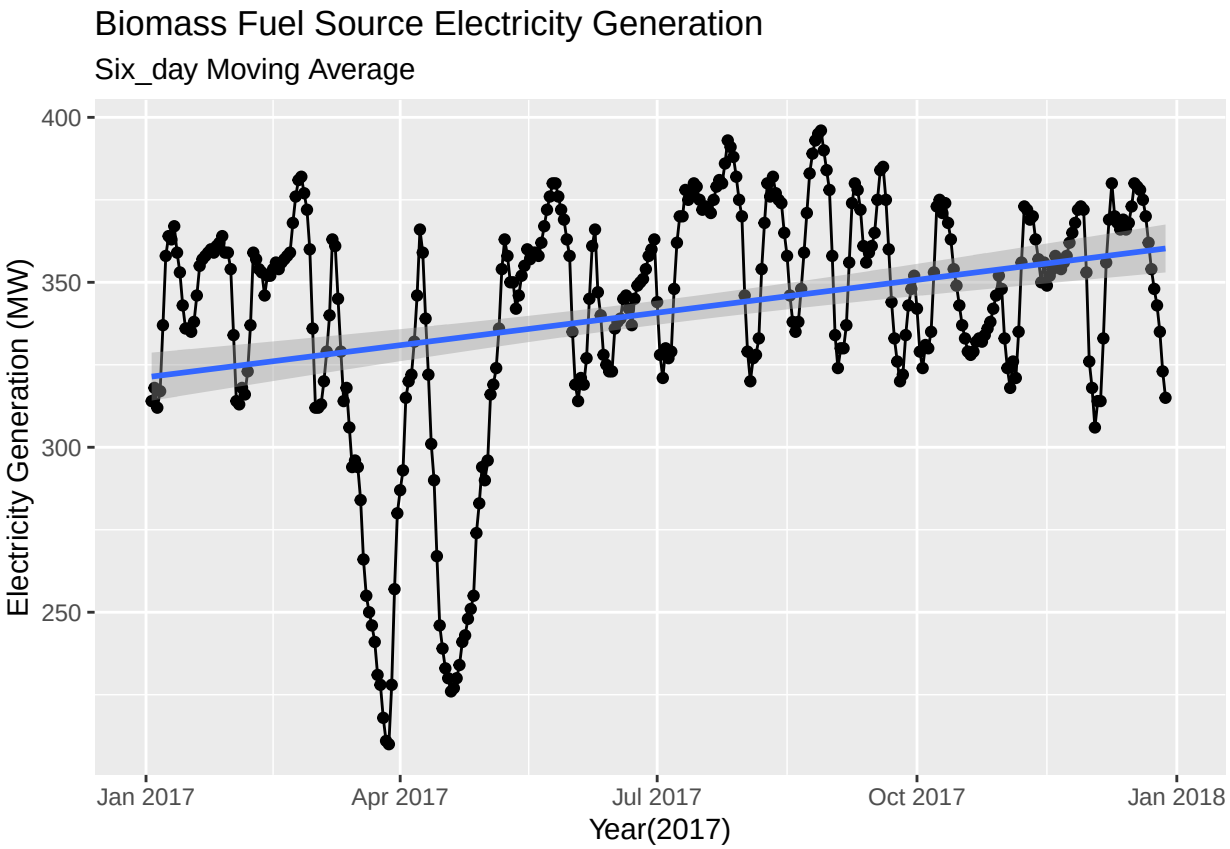
```r
## Oil Six Day
daily_average_oil <- daily_average_oil |> filter(format(date_only, "%Y") == "2017") |>
  mutate(oil_six_day =round(zoo::rollmean(daily_avg_oil, k = 6, fill = NA)))

# Combine all the columns into one frame
six_day_avgs <- cbind(daily_average_biomass, daily_average_brown_coal[,-1], daily_average_gas[,-1], dai

head(six_day_avgs,5)
```

```
##    date_only daily_avg_biomass biomass_six_day daily_avg_brown_coal
## 1 2017-01-01               336              NA                   913
## 2 2017-01-02               365              NA                   849
## 3 2017-01-03               361             314                   803
## 4 2017-01-04               242             318                     0
## 5 2017-01-05               229             312                    32
##   brown_coal_six_day daily_avg_gas gas_six_day daily_avg_hard_coal
## 1                 NA          4587          NA                5533
## 2                 NA          5438          NA                6428
## 3                534          4781        5031                3164
## 4                486          3715        5087                1340
## 5                488          4027        5706                1568
##   hard_cola_six_day daily_avg_oil oil_six_day
## 1                NA           173          NA
## 2                NA           312          NA
## 3              4080           313         266
## 4              3689           262         285
## 5              3552           217         285
```

ggplot of Biomass Electricity Generation

```r
ggplot(six_day_avgs, aes(x=date_only , y=biomass_six_day)) +
  geom_point(na.rm = TRUE) +
  geom_line(na.rm = TRUE) +
  geom_smooth(method = "lm", na.rm = TRUE) +
labs(
  title = "Biomass Fuel Source Electricity Generation",
  subtitle = "Six_day Moving Average",
  x = "Year(2017)",
  y = "Electricity Generation (MW)"
)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Conclusion

I am sure that there are definitely shorter routes to complete what I have done here. Given enough time I might have found them, but the exercise provided insight into how to frame pipes and functions to get to then end point of what was needed.