Data Summary Quiz

Set your root directory for the notebook

```
require("knitr")
```

```
## Loading required package: knitr
```

```
opts_knit$set(root.dir = "~/Data-Analysis-Machine-Learning/")
```

Read in the file using readr package

```
library(readr)
```

```
daily_spec <- read_csv("data-files/daily_SPEC_2014.csv.bz2")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   `Parameter Code` = col_integer(),
##   POC = col_integer(),
##   Latitude = col_double(),
##   Longitude = col_double(),
##   `Date Local` = col_date(format = ""),
##   `Observation Count` = col_integer(),
##   `Observation Percent` = col_double(),
##   `Arithmetic Mean` = col_double(),
##   `1st Max Value` = col_double(),
##   `1st Max Hour` = col_integer(),
##   `Method Code` = col_integer(),
##   `Date of Last Change` = col_date(format = "")
## )
```

```
## See spec(...) for full column specifications.
```

Get all column names

```
colnames(daily_spec)
```

```
##  [1] "State Code"          "County Code"         "Site Num"
##  [4] "Parameter Code"      "POC"                 "Latitude"
##  [7] "Longitude"           "Datum"               "Parameter Name"
## [10] "Sample Duration"     "Pollutant Standard"  "Date Local"
## [13] "Units of Measure"    "Event Type"          "Observation Count"
## [16] "Observation Percent" "Arithmetic Mean"     "1st Max Value"
## [19] "1st Max Hour"        "AQI"                 "Method Code"
## [22] "Method Name"         "Local Site Name"     "Address"
## [25] "State Name"          "County Name"         "City Name"
## [28] "CBSA Name"           "Date of Last Change"
```

What is the average Arithmetic mean for "Bromine PM2.5 LC" in the state of Wisconsin in the dataset?

```
library(tidyverse)
```

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

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

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

```
wisc_bpm <- daily_spec %>%
  filter(`State Name` == "Wisconsin",
         `Parameter Name` == "Bromine PM2.5 LC") %>%
  summarize(mean = mean(`Arithmetic Mean`))

wisc_bpm
```

```
## # A tibble: 1 x 1
##          mean
##         <dbl>
## 1 0.003960482
```

Calculate the average of each chemical constituent across all states, monitoring sites and all time.

Which constituent has the highest average level?

```
highest_level <- daily_spec %>%
  group_by(`Parameter Name`) %>%
  summarize(mean = mean(`Arithmetic Mean`, na.rm = TRUE) ) %>%
  arrange(desc(mean))

head(highest_level,5)
```

```
## # A tibble: 5 x 2
##                  `Parameter Name`      mean
##                             <chr>     <dbl>
## 1       Sample Max Baro Pressure 744.63264
## 2           Sample Baro Pressure 739.37011
## 3       Sample Min Baro Pressure 738.36388
## 4 OC CSN Unadjusted PM2.5 LC TOT  67.78383
## 5        Ambient Max Temperature  20.02881
```

Which monitoring site has the highest average level of "Sulfate PM2.5 LC" across all time

```
monitoring_site <- daily_spec %>%
  filter(`Parameter Name` == "Sulfate PM2.5 LC") %>%
  group_by(`State Code`, `County Code`, `Site Num`) %>%
  summarize(mean = mean(na.rm = TRUE, `Arithmetic Mean`)) %>%
  arrange(desc(mean))

head(monitoring_site, 5)
```

```
## # A tibble: 5 x 4
## # Groups:   State Code, County Code [5]
##   `State Code` `County Code` `Site Num`     mean
##          <chr>         <chr>      <chr>    <dbl>
## 1           39           081       0017 3.182189
## 2           42           003       0064 3.055483
## 3           54           039       1005 2.938800
## 4           18           019       0006 2.738700
## 5           39           153       0023 2.706449
```

What is the absolute difference in the aveage levels of "EC PM2.5 LC TOR" between the states California and Arizona, across all time and all monitoring sites

```
states <- c("California", "Arizona")
param <- "EC PM2.5 LC TOR"
abs_diff <- daily_spec %>%
  filter(`Parameter Name` == param, `State Name` %in% states) %>%
  group_by(`State Name`) %>%
  summarize(mean = mean(na.rm = TRUE, `Arithmetic Mean`))

diff <- abs(abs_diff$mean[1] - abs_diff$mean[2])
diff
```

```
## [1] 0.01856696
```

What is the median level of "OC PM2.5 LC TOR" in the Western United States, across all time? Define Western as any monitoring location that has a Longitude less that -100.

```
param <- "OC PM2.5 LC TOR"
med_level <- daily_spec %>%
  filter(`Parameter Name` == param, `Longitude` < -100) %>%
  summarize(median = median(na.rm = TRUE, `Arithmetic Mean`))
med_level
```

```
## # A tibble: 1 x 1
##   median
##    <dbl>
## 1   0.43
```

```
library(readxl)
```

```
aqs_sites <- read_excel("data-files/aqs_sites.xlsx")
```

```
## Warning in read_fun(path = path, sheet = sheet, limits = limits, shim =
## shim, : Expecting numeric in A20237 / R20237C1: got 'CC'
```

```
## Warning in read_fun(path = path, sheet = sheet, limits = limits, shim =
## shim, : Expecting numeric in A20238 / R20238C1: got 'CC'
```

```
## Warning in read_fun(path = path, sheet = sheet, limits = limits, shim =
## shim, : Expecting numeric in A20239 / R20239C1: got 'CC'
```

```
## Warning in read_fun(path = path, sheet = sheet, limits = limits, shim =
## shim, : Expecting numeric in A20240 / R20240C1: got 'CC'
```

```
colnames(aqs_sites)
```

```
##  [1] "State Code"            "County Code"
##  [3] "Site Number"           "Latitude"
##  [5] "Longitude"             "Datum"
##  [7] "Elevation"             "Land Use"
##  [9] "Location Setting"      "Site Established Date"
## [11] "Site Closed Date"      "Met Site State Code"
## [13] "Met Site County Code"  "Met Site Site Number"
## [15] "Met Site Type"         "Met Site Distance"
## [17] "Met Site Direction"    "GMT Offset"
## [19] "Owning Agency"         "Local Site Name"
## [21] "Address"               "Zip Code"
## [23] "State Name"            "County Name"
## [25] "City Name"             "CBSA Name"
## [27] "Tribe Name"            "Extraction Date"
```

```
str(aqs_sites$`Land Use`)
```

```
##  chr [1:20239] "RESIDENTIAL" "AGRICULTURAL" "FOREST" "UNKNOWN" ...
```

```
str(aqs_sites$`Location Setting`)
```

```
##  chr [1:20239] "SUBURBAN" "RURAL" "RURAL" "RURAL" "RURAL" ...
```

How many monitoring sites are labelled as both RESIDENTIAL for "Land Use" and SUBURBAN for "Location Setting"

```
resi_suburban <- aqs_sites %>%
  filter(`Land Use` == "RESIDENTIAL",
         `Location Setting` == "SUBURBAN") %>%
  distinct(`Site Number`, `State Code`, `County Code`) %>%
  summarize(n = n())

resi_suburban
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1  3527
```

What is the median level of "EC PM2.5 LC TOR" amongst monitoring sites that are labelled as both "RESIDENTIAL" and "SUBURBAN" in the eastern US, where eastern US is defined as Longitude >= -100

```
# Join the two data sets
param <- "EC PM2.5 LC TOR"
land_use <- "RESIDENTIAL"
location <- "SUBURBAN"
long <- -100
join <- left_join(daily_spec, aqs_sites, by = c("Latitude", "Longitude"))
```

```
med_level_eastern <- join %>%
  filter(`Land Use` == land_use, `Location Setting` == location,
         `Parameter Name` == param, `Longitude` >= long) %>%
  summarize(median = median(`Arithmetic Mean`, na.rm = TRUE))

med_level_eastern
```

```
## # A tibble: 1 x 1
##   median
##    <dbl>
## 1   0.61
```

```
str(daily_spec$`Date Local`)
```

```
##  Date[1:2108467], format: "2014-01-02" "2014-01-05" "2014-01-08" "2014-01-11" "2014-01-14" ...
```

```
class(daily_spec$`Date Local`)
```

```
## [1] "Date"
```

Amongst monitoring sites that are labeled as COMMERCIAL for "Land Use", which month of the year has the highest average levels of "Sulfate PM2.5 LC?"

```
library(lubridate)
```

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

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

```
param <- "Sulfate PM2.5 LC"
land_use <- "COMMERCIAL"
comm_sulfate <- join %>%
  filter(`Parameter Name` == param, `Land Use` == land_use) %>%
  mutate(month = month(`Date Local`)) %>%
  group_by(month) %>%
  summarize(average_level = mean(`Arithmetic Mean`, na.rm = TRUE)) %>%
  arrange(desc(average_level))

head(comm_sulfate, 5)
```

```
## # A tibble: 5 x 2
##   month average_level
##   <dbl>         <dbl>
## 1     2      2.021325
## 2     3      1.805260
## 3     7      1.777605
## 4     8      1.761226
## 5     6      1.750571
```

Take a look at the monitoring site State Code = 6,Conty Code = 65, Site Number 8001. At this monitor, for how many days is the sum of "Sulfate PM2.5 LC" and "Total Nitrate PM2.5 LC" greater than 10.

```
california <- daily_spec %>%
  filter(`State Code` == "06", `County Code` == "065", `Site Num` == "8001",
         `Parameter Name` %in% c("Sulfate PM2.5 LC", "Total Nitrate PM2.5 LC")) %>%
  group_by(`Parameter Name`, `Date Local`) %>%
  summarize(level = mean(`Arithmetic Mean`, na.rm = TRUE)) %>%
  group_by(`Date Local`) %>%
  summarize(total = sum(`level`)) %>%
  filter(total > 10) %>%
  count()
```

Which monitoring site has the highest correlation between "Sulfate PM2.5 LC" and "Total Nitrate PM2.5 LC" across all dates? When multiple values are on a given date, take the average of the constituent for that date.

```
corr <- daily_spec %>%
  filter(`Parameter Name`  %in% c("Sulfate PM2.5 LC",
                                  "Total Nitrate PM2.5 LC")) %>%
  group_by(`Parameter Name`, `State Code`, `County Code`, `Site Num`,
           `Date Local`) %>%
  summarize(level = mean(`Arithmetic Mean`, na.rm = TRUE)) %>%
```

```
  spread(`Parameter Name`, level) %>%
  group_by(`State Code`, `County Code`, `Site Num`) %>%
  summarize(correlation = cor(`Sulfate PM2.5 LC`, `Total Nitrate PM2.5 LC`)) %>%
  arrange(desc(correlation))
```

```
head(corr, 5)
```

```
## # A tibble: 5 x 4
## # Groups:   State Code, County Code [4]
##   `State Code` `County Code` `Site Num` correlation
##          <chr>         <chr>      <chr>       <dbl>
## 1           02           090       0035   0.8978038
## 2           08           001       0006   0.8956944
## 3           34           001       0006   0.8812428
## 4           42           045       0002   0.8739804
## 5           02           090       0010   0.8637321
```