Lab 5

Jazmin Hernandez

Reading in Data

```
library(data.table)
library(dtplyr)
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:data.table':

    between, first, last

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
library(ggplot2)
met <- read.csv(file.path("~", "Github", "met_all.gz"))
head(met)
```

```
  USAFID  WBAN year month day hour min  lat      lon elev wind.dir wind.dir.qc
1 690150 93121 2019     8   1    0  56 34.3 -116.166  696      220           5
2 690150 93121 2019     8   1    1  56 34.3 -116.166  696      230           5
```

```
3 690150 93121 2019     8    1     2  56 34.3 -116.166  696       230             5
4 690150 93121 2019     8    1     3  56 34.3 -116.166  696       210             5
5 690150 93121 2019     8    1     4  56 34.3 -116.166  696       120             5
6 690150 93121 2019     8    1     5  56 34.3 -116.166  696        NA             9
  wind.type.code wind.sp wind.sp.qc ceiling.ht ceiling.ht.qc ceiling.ht.method
1              N     5.7          5      22000             5                 9
2              N     8.2          5      22000             5                 9
3              N     6.7          5      22000             5                 9
4              N     5.1          5      22000             5                 9
5              N     2.1          5      22000             5                 9
6              C     0.0          5      22000             5                 9
  sky.cond vis.dist vis.dist.qc vis.var vis.var.qc temp temp.qc dew.point
1        N    16093           5       N          5 37.2       5      10.6
2        N    16093           5       N          5 35.6       5      10.6
3        N    16093           5       N          5 34.4       5       7.2
4        N    16093           5       N          5 33.3       5       5.0
5        N    16093           5       N          5 32.8       5       5.0
6        N    16093           5       N          5 31.1       5       5.6
  dew.point.qc atm.press atm.press.qc       rh
1            5    1009.9            5 19.88127
2            5    1010.3            5 21.76098
3            5    1010.6            5 18.48212
4            5    1011.6            5 16.88862
5            5    1012.7            5 17.38410
6            5    1012.7            5 20.01540
```

```r
stations <- fread("https://noaa-isd-pds.s3.amazonaws.com/isd-history.csv")
stations <- as.data.frame(stations)
stations$USAF <- as.integer(stations$USAF)
```

```
Warning: NAs introduced by coercion
```

```r
stations$USAF[stations$USAF == 999999] <- NA
stations$CTRY[stations$CTRY == ""] <- NA
stations$STATE[stations$STATE == ""] <- NA
```

```r
stations <- unique(stations[, c('USAF', 'CTRY', 'STATE')])
stations <- stations[!is.na(stations$USAF), ]
head(stations, n = 4)
```

```
  USAF CTRY STATE
```

```
1 7018 <NA>  <NA>
2 7026   AF  <NA>
3 7070   AF  <NA>
4 8260 <NA>  <NA>
```

```
# Merging data
merge(
  # Data
  x      = met,
  y      = stations,
  # List of variables to match
  by.x   = "USAFID",
  by.y   = "USAF",
  # Which obs to keep?
  all.x = TRUE,
  all.y = FALSE
  ) |> nrow()
```

```
[1] 2385443
```

```
stations <- stations[!duplicated(stations$USAF), ]
```

```
# Fixed data dropping duplicate IDs from stations
met <- merge(
  x      = met,
  y      = stations,
  by.x   = "USAFID",
  by.y   = "USAF",
  all.x = TRUE,
  all.y = FALSE
  )
head(met[, c('USAFID', 'WBAN', 'STATE')], n = 4)
```

```
  USAFID  WBAN STATE
1 690150 93121    CA
2 690150 93121    CA
3 690150 93121    CA
4 690150 93121    CA
```

Question 1: Representative station for the US

The three weather stations that best represent continental US are located in California, Arkansas, and Michigan. This makes sense as these states are located at different extremes of the US and would therefore better be representative of weather in the US.

```
# Finding median values
library(dplyr)
library(data.table)
median_weather <- met |>
group_by(USAFID, STATE, CTRY, lat, lon, temp, wind.sp, atm.press) |>
  summarise(
    median_temp = median(temp, na.rm = TRUE),
    median_wind.sp = median(wind.sp, na.rm = TRUE),
    median_atm.press = median(atm.press, na.rm = TRUE)
  )
```

```
`summarise()` has grouped output by 'USAFID', 'STATE', 'CTRY', 'lat', 'lon',
'temp', 'wind.sp'. You can override using the `.groups` argument.
```

```
head(median_weather, 4)
```

```
# A tibble: 4 x 11
# Groups:   USAFID, STATE, CTRY, lat, lon, temp, wind.sp [3]
  USAFID STATE CTRY    lat   lon  temp wind.sp atm.press median_temp
   <int> <chr> <chr> <dbl> <dbl> <dbl>   <dbl>     <dbl>       <dbl>
1 690150 CA    US     34.3 -116.  22.8     0       1013.        22.8
2 690150 CA    US     34.3 -116.  23.3     2.1     1014.        23.3
3 690150 CA    US     34.3 -116.  23.9     4.6     1010         23.9
4 690150 CA    US     34.3 -116.  23.9     4.6     1013.        23.9
# i 2 more variables: median_wind.sp <dbl>, median_atm.press <dbl>
```

```
# Using quantile function
temp_quantiles <- quantile(median_weather$median_temp, probs = c(0.25, 0.5, 0.75), na.rm = T
wind.sp_quantiles <- quantile(median_weather$median_wind.sp, probs = c(0.25, 0.5, 0.75), na.
atm.press_quantiles <- quantile(median_weather$median_atm.press,probs = c(0.25, 0.5, 0.75), 
print(temp_quantiles)
```

```
 25%  50%  75%
20.1 24.4 28.3
```

```r
print(wind.sp_quantiles)
```

```
25% 50% 75% 
1.5 2.6 4.1 
```

```r
print(atm.press_quantiles)
```

```
   25%    50%    75% 
1011.7 1014.1 1016.5 
```

```r
# Three weather stations that best represent continental US
rep_stations_temp <- median_weather |>
filter(median_temp <= temp_quantiles[3])
rep_stations_wind.sp <- median_weather |>
filter(median_wind.sp <= wind.sp_quantiles[3])
rep_stations_atm.press <- median_weather |>
filter(median_atm.press <= atm.press_quantiles[3])
```

```r
print(head(rep_stations_temp, 3))
```

```
# A tibble: 3 x 11
# Groups:   USAFID, STATE, CTRY, lat, lon, temp, wind.sp [3]
  USAFID STATE CTRY    lat   lon  temp wind.sp atm.press median_temp
   <int> <chr> <chr> <dbl> <dbl> <dbl>   <dbl>     <dbl>       <dbl>
1 690150 CA    US     34.3 -116.  22.8     0       1013.         22.8
2 690150 CA    US     34.3 -116.  23.3     2.1     1014.         23.3
3 690150 CA    US     34.3 -116.  23.9     4.6     1010          23.9
# i 2 more variables: median_wind.sp <dbl>, median_atm.press <dbl>
```

```r
print(head(rep_stations_wind.sp, 3))
```

```
# A tibble: 3 x 11
# Groups:   USAFID, STATE, CTRY, lat, lon, temp, wind.sp [3]
  USAFID STATE CTRY    lat   lon  temp wind.sp atm.press median_temp
   <int> <chr> <chr> <dbl> <dbl> <dbl>   <dbl>     <dbl>       <dbl>
1 690150 CA    US     34.3 -116.  22.8     0       1013.         22.8
2 690150 CA    US     34.3 -116.  23.3     2.1     1014.         23.3
3 690150 CA    US     34.3 -116.  25.6     1.5     1013.         25.6
# i 2 more variables: median_wind.sp <dbl>, median_atm.press <dbl>
```

```r
print(head(rep_stations_atm.press, 3))
```

```
# A tibble: 3 x 11
# Groups:   USAFID, STATE, CTRY, lat, lon, temp, wind.sp [3]
  USAFID STATE CTRY    lat   lon  temp wind.sp atm.press median_temp
   <int> <chr> <chr> <dbl> <dbl> <dbl>   <dbl>     <dbl>       <dbl>
1 690150 CA    US     34.3 -116.  22.8     0        1013.        22.8
2 690150 CA    US     34.3 -116.  23.3     2.1      1014.        23.3
3 690150 CA    US     34.3 -116.  23.9     4.6      1010         23.9
# i 2 more variables: median_wind.sp <dbl>, median_atm.press <dbl>
```

Question 2: Representative station per state

The station shown at the lowest latitude is located in Montana, CA.

```r
# Calculating euclidean distance
overall_median <- colMeans(median_weather[, c("median_temp", "median_wind.sp", "median_atm.p
met <- median_weather
```

```r
median_weather <- median_weather |>
  mutate(
    euclidean_distance = sqrt(
      (median_temp - overall_median[1])^2 +
      (median_wind.sp - overall_median[2])^2 +
      (median_atm.press - overall_median[3])^2
    )
  )
```

```r
representative_stations_state <- data.frame()


# Find the representative station
for (state in unique(median_weather$STATE))
  state_data <- median_weather |>
    filter(STATE == state)

  # Get the station with the minimum distance, with a tie-breaker on latitude
  selected_station <- state_data |>
    arrange(euclidean_distance, lat) |>
    slice(1)
```

```r
representative_stations_state <- rbind(representative_stations_state, selected_station)
print(representative_stations_state)
```

```
# A tibble: 3,384 x 12
# Groups:   USAFID, STATE, CTRY, lat, lon, temp, wind.sp [3,384]
   USAFID STATE CTRY    lat   lon  temp wind.sp atm.press median_temp
    <int> <chr> <chr> <dbl> <dbl> <dbl>   <dbl>     <dbl>       <dbl>
 1 726676 MT    US     47.1 -105.   5       3.6     1017.         5
 2 726676 MT    US     47.1 -105.   6.1     2.1     1018          6.1
 3 726676 MT    US     47.1 -105.   6.7     2.6     1018.         6.7
 4 726676 MT    US     47.1 -105.   6.7     3.6     1016.         6.7
 5 726676 MT    US     47.1 -105.   7       3.1       NA          7
 6 726676 MT    US     47.1 -105.   7       3.6       NA          7
 7 726676 MT    US     47.1 -105.   7.2     3.6     1013.         7.2
 8 726676 MT    US     47.1 -105.   7.8     1.5     1017          7.8
 9 726676 MT    US     47.1 -105.   7.8     2.1     1018.         7.8
10 726676 MT    US     47.1 -105.   7.8     2.6     1016.         7.8
# i 3,374 more rows
# i 3 more variables: median_wind.sp <dbl>, median_atm.press <dbl>,
#   euclidean_distance <dbl>
```

Question 3: In the middle?

```r
library(data.table)
library(dplyr)
library(leaflet)
# Find mid-point for each state
state_midpoints <- met |>
  group_by(STATE) |>
  summarise(
    mid_lat = mean(lat, na.rm = TRUE),
    mid_long = mean(lon, na.rm = TRUE),
    .groups = 'drop'
  )
print(head(state_midpoints, 5))
```

```
# A tibble: 5 x 3
  STATE mid_lat mid_long
  <chr>   <dbl>    <dbl>
```

```
1 AL       32.6    -86.6
2 AR       35.3    -92.6
3 AZ       33.7   -111.
4 CA       36.2   -120.
5 CO       39.1   -106.
```

```
distances <- met |>
inner_join(state_midpoints, by = "STATE") |>
mutate(
distance = sqrt((lat - mid_lat)^2 + (lon - mid_long)^2)  # Calculate Euclidean distance
  ) |>
  select(STATE, USAFID, lat, lon, distance)  # Select relevant columns
```

```
Adding missing grouping variables: `CTRY`, `temp`, `wind.sp`
```

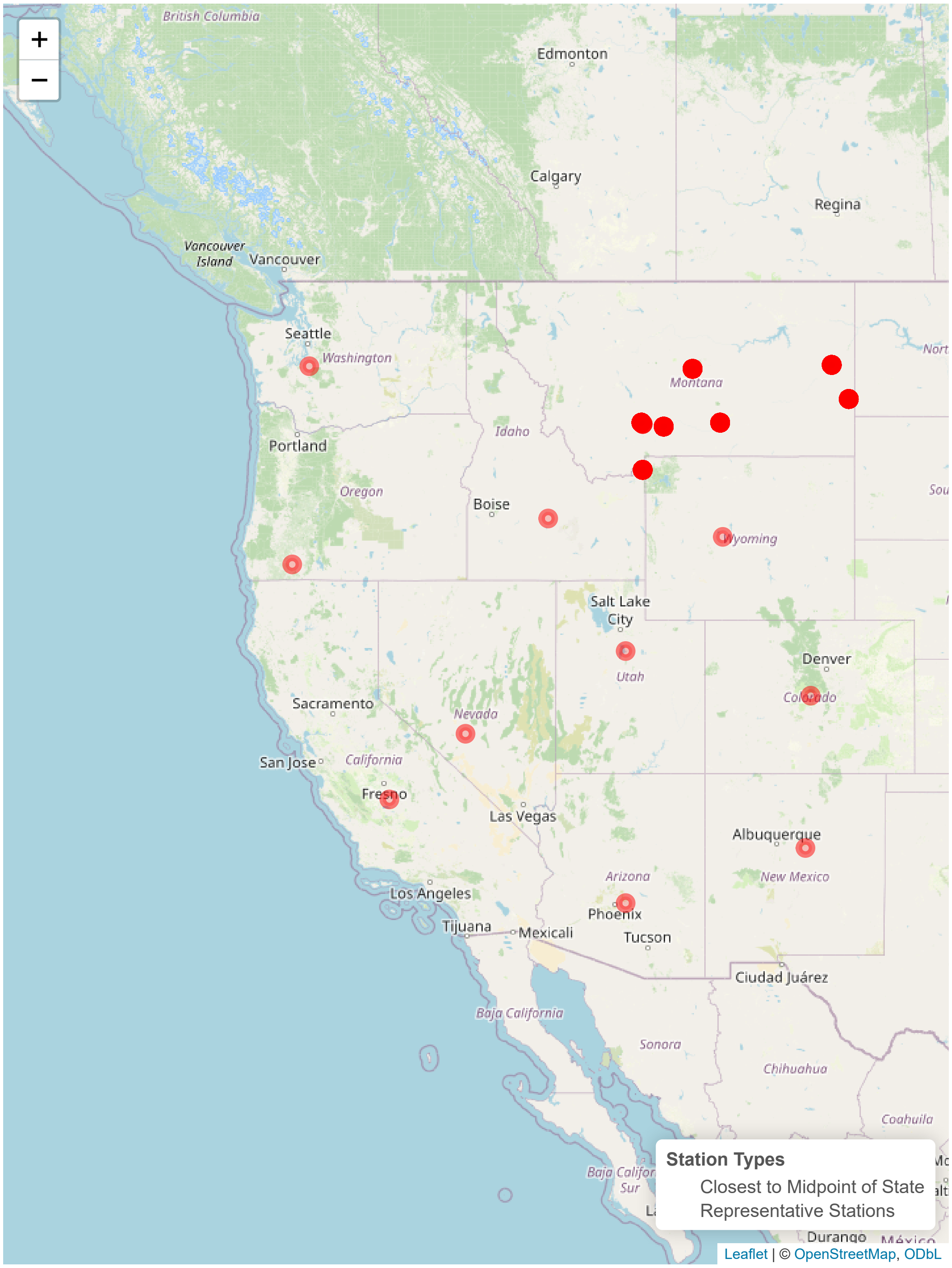
```
# Closest station to mid-point
library(tidyr)
closest_stations <- distances |>
group_by(STATE) |>
slice(which.min(distance)) |>
ungroup()
print(head(closest_stations, 5))
```

```
# A tibble: 5 x 8
  CTRY   temp wind.sp STATE USAFID   lat    lon distance
  <chr> <dbl>   <dbl> <chr>  <int> <dbl>  <dbl>    <dbl>
1 US     22.9     1.5 AL    722265  32.4  -86.4   0.300
2 US     15       0   AR    720401  35.6  -92.4   0.349
3 US     31.7     0   AZ    722783  33.5 -112.    0.477
4 US     25.6     0   CA    723898  36.3 -120.    0.179
5 US      9       3.1 CO    726396  39.0 -106.    0.0901
```

```
all_stations <- bind_rows(
representative_stations_state,
closest_stations
) |>
distinct()
```

```
library(leaflet)
leaflet(all_stations) |>
```

```
addTiles() |>
addCircleMarkers(
    lng = ~lon,
    lat = ~lat,
    color = ifelse(all_stations$STATE %in% unique(closest_stations$STATE), "red", "purple"),
    radius = 5,
    label = ~USAFID,
    group = "Stations"
  ) |>
addLegend("bottomright",
colors = c("black", "blue"),
labels = c("Closest to Midpoint of State", "Representative Stations"),
title = "Station Types")
```


+
−
British Columbia
Edmonton
Calgary
Regina
Vancouver Island
Vancouver
Seattle
Washington
Portland
Oregon
Idaho
Montana
Nort
Boise
Wyoming
Sou
Salt Lake City
Utah
Denver
Colorado
Sacramento
Nevada
San Jose
California
Fresno
Las Vegas
Albuquerque
New Mexico
Arizona
Los Angeles
Phoenix
Tijuana
Mexicali
Tucson
Ciudad Juárez
Baja California
Sonora
Chihuahua
Coahuila
Baja California Sur
Durango
México
Station Types
Closest to Midpoint of State
Representative Stations
Leaflet | © OpenStreetMap, ODbL

Question 4: Means of means

```
state_summary <- met |>
  group_by(STATE) |>
  summarise(
    avg_temp = mean(temp, na.rm = TRUE),
    avg_wind.sp = mean(wind.sp, na.rm = TRUE),
    avg_atm.press = mean(atm.press, na.rm = TRUE),
    .groups = 'drop'
  )
```

```
state_summary <- state_summary |>
  mutate(
    temp_level = case_when(
      avg_temp < 20 ~ "Low",
      avg_temp >= 20 & avg_temp < 25 ~ "Mid",
      avg_temp >= 25 ~ "High",
      TRUE ~ NA_character_
    )
  )
```

```
#generating rest of summary table
summary_table <- state_summary |>
group_by(temp_level) |>
summarise(
    num_entries = n(),
    num_na_entries = sum(is.na(avg_temp)),
    num_stations = n_distinct(STATE),  # Assuming each State corresponds to one station
    num_states = n(),  # Number of unique states in each temperature level
    mean_temp = mean(avg_temp, na.rm = TRUE),
    mean_wind_speed = mean(avg_wind.sp, na.rm = TRUE),
    mean_atm_pressure = mean(avg_atm.press, na.rm = TRUE),
    .groups = 'drop'
  )
print(summary_table)
```

```
# A tibble: 3 x 8
  temp_level num_entries num_na_entries num_stations num_states mean_temp
  <chr>            <int>          <int>        <int>      <int>     <dbl>
1 High                14              0           14         14      27.2
2 Low                  8              0            8          8      19.4
```

```
3 Mid                 26              0            26         26      22.7
# i 2 more variables: mean_wind_speed <dbl>, mean_atm_pressure <dbl>
```