STATS 506 HW 4

Calder Moore

STATS 506 HW 4

Problem 1 - Tidyverse: New Zealand

a)

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.1     v stringr   1.5.1
v ggplot2   4.0.0     v tibble    3.3.0
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.1.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(nzelect)

nztib <- tibble(vote = nzge$votes, year = nzge$election_year, type = nzge$voting_type) %>%
          group_by(type, year) %>%
          summarise(vote_total = sum(vote)) %>%
          arrange(desc(vote_total))
```

```
`summarise()` has grouped output by 'type'. You can override using the
`.groups` argument.
```

```r
nztib
```

```
# A tibble: 10 x 3
# Groups:   type [2]
   type       year vote_total
   <chr>     <dbl>      <dbl>
 1 Party      2014    2416479
 2 Candidate  2014    2375493
 3 Party      2008    2356536
 4 Candidate  2008    2325598
 5 Party      2005    2286190
 6 Candidate  2005    2260670
 7 Party      2011    2257336
 8 Candidate  2011    2225766
 9 Party      2002    2040248
10 Candidate  2002    2022115
```

b)

```r
vote2014 <- tibble(vote = nzge$votes, year = nzge$election_year, party = nzge$party) %>%
              group_by(year, party) %>%
              filter(year == 2014) %>%
              summarise(vote_total = sum(vote)) %>%
              mutate(vote_percent = 100*vote_total/sum(vote_total))
```

```
`summarise()` has grouped output by 'year'. You can override using the
`.groups` argument.
```

```r
vote2014
```

```
# A tibble: 28 x 4
# Groups:   year [1]
    year party                            vote_total vote_percent
   <dbl> <chr>                                 <dbl>        <dbl>
 1  2014 ACT New Zealand                       44467      0.928
 2  2014 Alliance                                 59      0.00123
 3  2014 Aotearoa Legalise Cannabis Party      15897      0.332
 4  2014 Ban1080                                9561      0.200
 5  2014 Climate Party                           116      0.00242
```

```
 6  2014 Communist League                        135      0.00282
 7  2014 Conservative Party                   176673      3.69
 8  2014 Democrats for Social Credit            6377      0.133
 9  2014 Focus New Zealand                      2436      0.0508
10  2014 Green Party                          423077      8.83
# i 18 more rows
```

c)

```r
win <- tibble(vote = nzge$votes, year = nzge$election_year, party = nzge$party, type = nzge$v
              group_by(year, party, type) %>%
              summarise(vote_total = sum(vote)) %>%
              group_by(year) %>%
              mutate(vote_percent = 100*vote_total/sum(vote_total)) %>%
              group_by(year, type) %>%
              mutate(winner = party[which.max(vote_percent)]) %>%
              slice_max(vote_percent) %>%
              select(year, type, winner)
```

```
`summarise()` has grouped output by 'year', 'party'. You can override using the
`.groups` argument.
```

```r
win
```

```
# A tibble: 10 x 3
# Groups:   year, type [10]
    year type      winner
   <dbl> <chr>     <chr>
 1  2002 Candidate Labour Party
 2  2002 Party     Labour Party
 3  2005 Candidate National Party
 4  2005 Party     Labour Party
 5  2008 Candidate National Party
 6  2008 Party     National Party
 7  2011 Candidate National Party
 8  2011 Party     National Party
 9  2014 Candidate National Party
10  2014 Party     National Party
```

Problem 2 - Tidyverse: Tennis

a)

```r
tennis <- read.csv("atp_matches_2019.txt")

tourney_count <- tennis %>%
  mutate(tourney_date = ymd(tourney_date)) %>%
  filter(year(tourney_date) == 2019) %>%
  distinct(tourney_date)

nrow(tourney_count)
```

```
[1] 48
```

There were 48 tournaments in 2019.

b)

```r
winners <- tennis %>%
  group_by(tourney_id) %>%
  slice_head(n = 1) %>%
  ungroup() %>%
  count(winner_name, sort = TRUE) %>%
  filter(n > 1)

nrow(winners)
```

```
[1] 25
```

```r
max(winners$n)
```

```
[1] 7
```

25 players have won more than one tournament, and the most winning player has won 7 tournaments.

c)

```r
tennis %>%
  summarise(
    w_ace_mean = mean(w_ace, na.rm = TRUE),
    l_ace_mean  = mean(l_ace, na.rm = TRUE),
    w_ace_sd   = sd(w_ace, na.rm = TRUE),
    l_ace_sd    = sd(l_ace, na.rm = TRUE)
  )
```

```
  w_ace_mean l_ace_mean w_ace_sd l_ace_sd
1   7.497402   5.792502 6.065966 5.631426
```

They have similar standard deviations and the means are around 2 aces apart, so there does seem to be evidence for a difference in means in the number of aces hit by winners vs losers.

d)

```r
players <- tennis %>%
  select(tourney_id, winner_name, loser_name) %>%
  pivot_longer(
    cols = c(winner_name, loser_name),
    names_to = "outcome",
    values_to = "player") %>%
  mutate(
    win = if_else(outcome == "winner_name", 1, 0)
  )

winrate <- players %>%
  group_by(player) %>%
  summarise(
    games = n(),
    wins = sum(win),
    win_rate = wins/games) %>%
  filter(games > 5) %>%
  arrange(desc(win_rate))

winrate
```

```
# A tibble: 161 x 4
   player             games  wins win_rate
   <chr>              <int> <dbl>    <dbl>
 1 Rafael Nadal          69    60    0.870
 2 Novak Djokovic        69    58    0.841
 3 Roger Federer         66    55    0.833
 4 Daniil Medvedev       80    59    0.738
 5 Kevin Anderson        15    11    0.733
 6 Dominic Thiem         69    50    0.725
 7 Attila Balazs         10     7    0.7
 8 Stefanos Tsitsipas    80    55    0.688
 9 Alex De Minaur        62    42    0.677
10 Kei Nishikori         43    29    0.674
# i 151 more rows
```

Nadal has the highest win rate at 86.96% of games won.

Problem 3 - Visualization

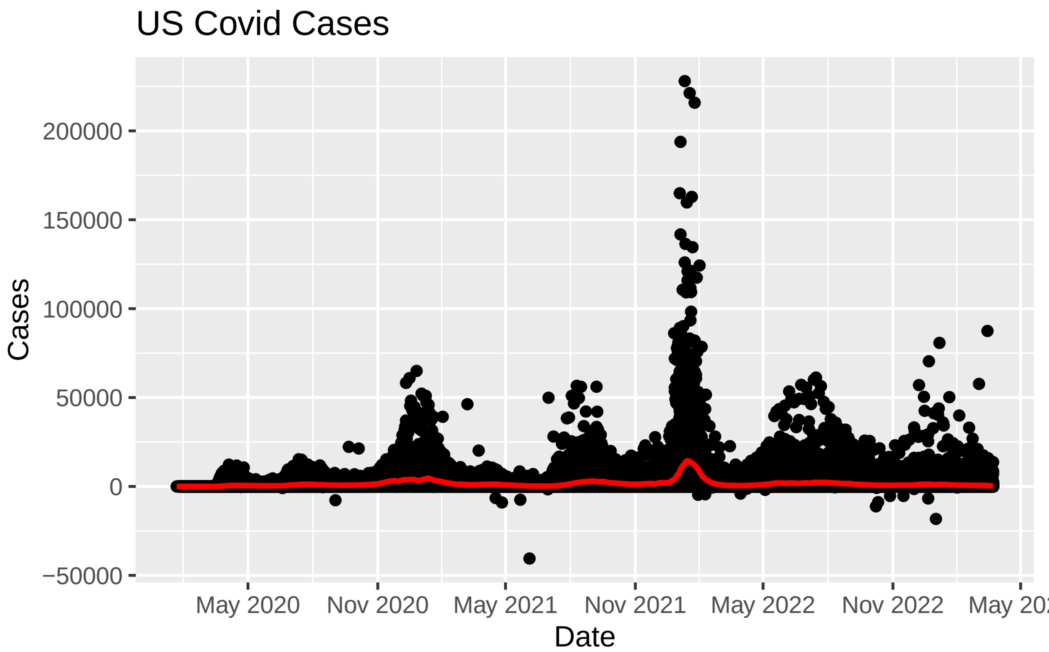
a)

```r
library(ggplot2)
covid <- read.csv("us-states.txt")

covid$date <- as.Date(covid$date)

case_mean <- covid %>%
  group_by(date) %>%
  summarise(cases_avg = mean(cases_avg, na.rm = TRUE))

ggplot(covid, aes(x = date, y = cases)) +
  geom_point() +
  geom_line(data = case_mean, aes(y = cases_avg), color = "red", linewidth = 1) +
  scale_x_date(date_breaks = "6 months", date_labels = "%b %Y") +
  labs(title = "US Covid Cases", x = "Date", y = "Cases")
```



There seem to be two major spikes in December 2020 - January 2021 and again from December 2021 - January 2022, and five smaller spikes in April 2020, July 2020, September 2021, July 2022, and December 2022.

b)

```
states <- covid %>%
  group_by(state) %>%
  summarise(rate_avg = mean(cases_avg_per_100k, na.rm = TRUE)) %>%
  arrange(desc(rate_avg))

states$state[1:3]
```

```
[1] "American Samoa" "Rhode Island"   "Alaska"
```

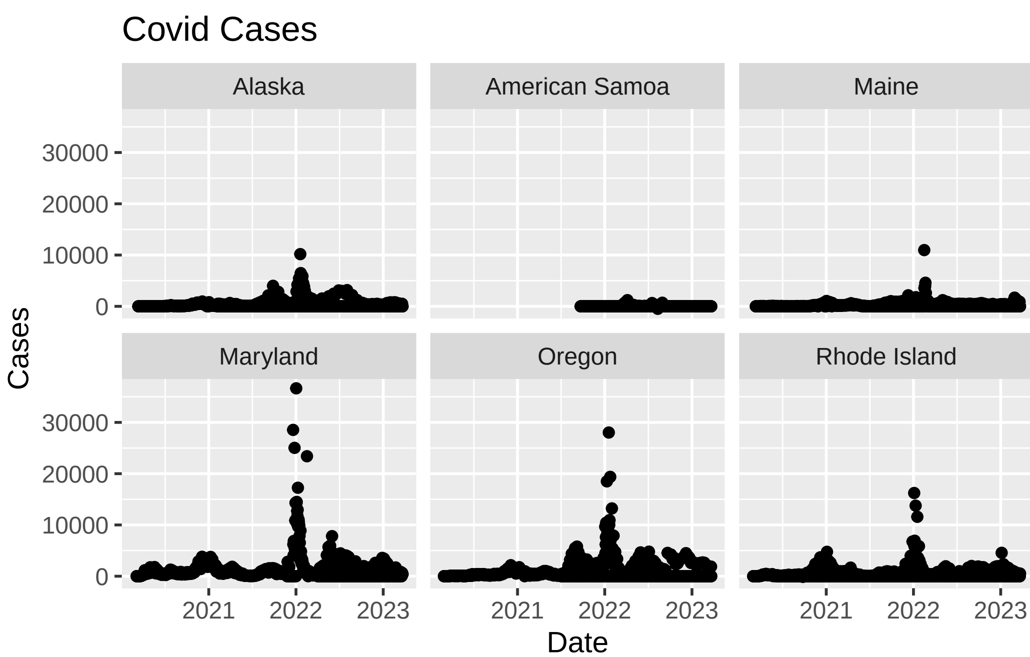
```
states$state[(nrow(states)-2):nrow(states)]
```

```
[1] "Oregon"   "Maine"    "Maryland"
```

American Samoa, Rhode Island, and Alaska have the highest rates and Oregon, Maine, and Maryland have the lowest rates.

```r
case_states <- covid %>%
  filter(state %in% c(states$state[1:3], states$state[(nrow(states)-2):nrow(states)]))

ggplot(data = case_states, aes(x = date, y = cases)) +
  geom_point() +
  labs(title = "Covid Cases", x = "Date", y = "Cases") +
  facet_wrap(vars(state))
```



Interestingly it appears that the states with the lowest average were actually hit harder by the big spike in early 2022. Despite having the highest running average, American Samoa appears to have a relatively flat line, possibly with their average just being higher in general even though they didn't experience an extreme spike, or because there isn't any data from there prior to late 2021.

c)

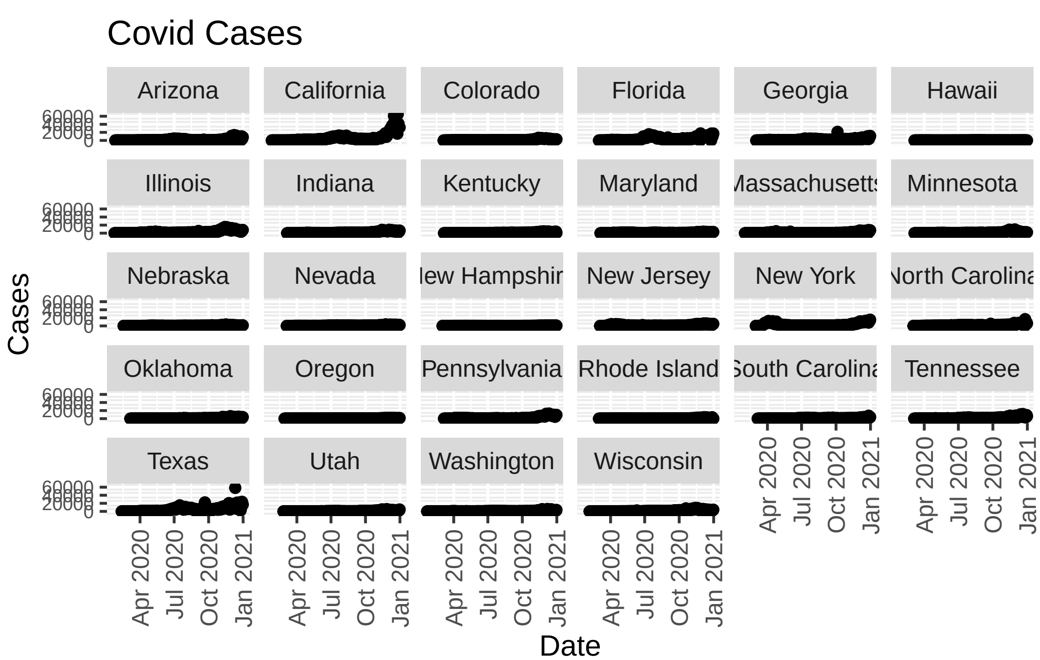
```r
state_list <- unique(covid$state)

firsthalf <- state_list[1:(length(state_list)/2)]
secondhalf <- state_list[(length(state_list)/2 + 1):length(state_list)]
```

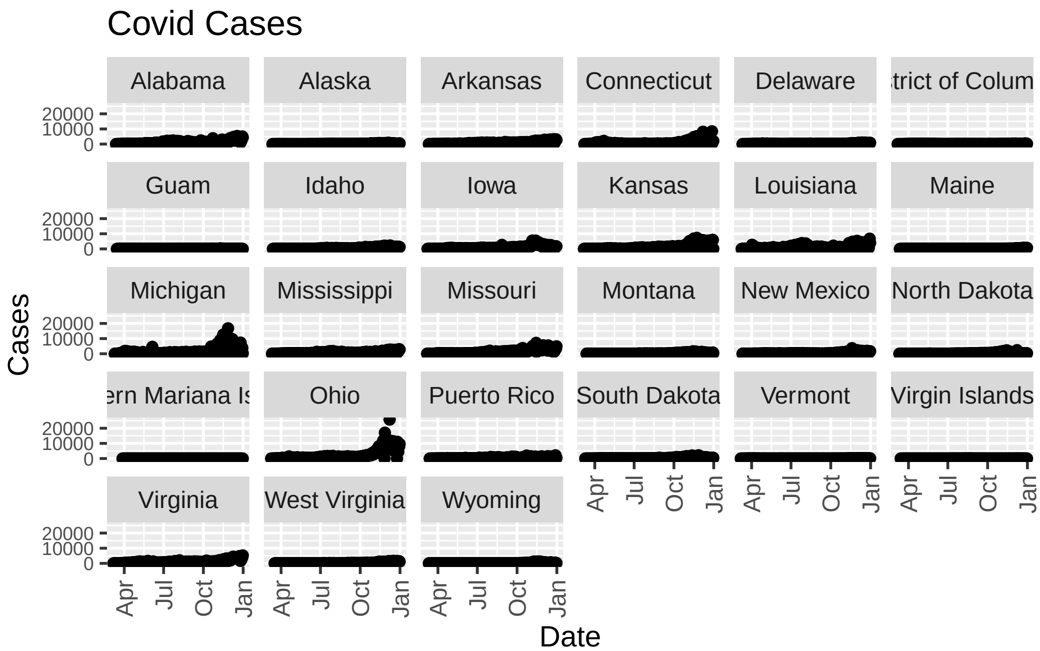
```
plot1 <- covid %>%
  filter(state %in% firsthalf) %>%
  filter(date < as.Date("2021-01-01"))

plot2 <- covid %>%
  filter(state %in% secondhalf) %>%
  filter(date < as.Date("2021-01-01"))

ggplot(data = plot1, aes(x = date, y = cases)) +
  geom_point() +
  labs(title = "Covid Cases", x = "Date", y = "Cases") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)) +
  theme(axis.text.y = element_text(size = 7)) +
  facet_wrap(vars(state))
```




```
ggplot(data = plot2, aes(x = date, y = cases)) +
  geom_point() +
  labs(title = "Covid Cases", x = "Date", y = "Cases") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)) +
  theme(axis.text.y = element_text(size = 7)) +
  facet_wrap(vars(state))
```



New York, Florida, Connecticut, Michigan, and Texas were among some of the states that were hit by covid cases earliest.