Project 1

Jason Braverman

2/21/25

This is the dataset you will be working with:

```
olympics <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2



olympics_alpine <- olympics %>%
  filter(!is.na(weight)) %>%             # only keep athletes with known weight
  filter(sport == "Alpine Skiing") %>%   # keep only alpine skiers
  mutate(
    medalist = case_when(                # add column to
      is.na(medal) ~ FALSE,              # NA values go to FALSE
      !is.na(medal) ~ TRUE               # non-NA values (Gold, Silver, Bronze) go to TRUE
    )
  )
```

`olympics_alpine` is a subset of `olympics` and contains only the data for alpine skiers. More information about the original `olympics` dataset can be found at https://github.com/rfordatascience/tidytuesday/tree/master/data/2021/2021-07-27/readme.md and https://www.sports-reference.com/olympics.html.

For this project, use `olympics_alpine` to answer the questions 1, 2 and 3, about the weights of alpine skiers. For Question 4, you should use the FULL dataset, `olympics`

1. Are there weight differences for male and female Olympic skiers who were successful or not in earning a medal?
2. Are there weight differences for skiers who competed in different alpine skiing events?
3. How has the weight distribution of alpine skiers changed over the years?
4. Create ANY interesting animation from the full `olympics` dataset.

You should make one plot per question.

Hints:

- We recommend you use a violin plot for question 1 and boxplots for questions 2 and 3. However, you are free to use any of the plots we have discussed in class.
- For question 3, it may be helpful to consider only a subset of alpine skiers, such as those who competed in a specific event.
- To make a series of boxplots over time, you will have to add the following to your `aes()` statement: `group = year`.
- It can be a bit tricky to re-label facets generated with `facet_wrap()`. The trick is to add a `labeller` argument, for example:

```
+ facet_wrap(
   # your other arguments to facet_wrap() go here
   ...,
   # this replaces "TRUE" with "medaled" and "FALSE" with "did not medal"
   labeller = as_labeller(c(`TRUE` = "medaled", `FALSE` = "did not medal"))
 )
```

Introduction: *Your introduction here.*

The dataset that is being used contains data from the Olympics starting in Athens in 1896 to Rio in 2016. Each row contains information about the athlete competing (name, sex, age, weight, team, and nationality), the year, season, and city of which game they competed in, the sport competed, if they got a medal or not, and what that medal was (Gold, Silver, or Bronze) if true. The dataset then was filtered to have a focus on just competitors in alpine skiing events. The questions being answered are:

1. What is the distribution of weight between male and female alpine skiers if they won a medal or not?
2. What is the average weight for each skiing event between males and females?
3. How has the average female skier weight changed over the course of Olympic games?
4. Where around the world have the Olympics been held?

An additional dataset was needed to answer the last question of how were the games distributed across the globe because the original dataset did not include latitude and longitude of the host cities.

Approach: *Your approach here.*

The approach that was used for question 1 was to separate the graphs per gender, then further divide the data by if the competitor won a medal or not. This was put into a violin plot to be able to view the distribution of competitor weights. A regression analysis was also done to determine if there was any statistical significance between sex and weight to earning a medal.

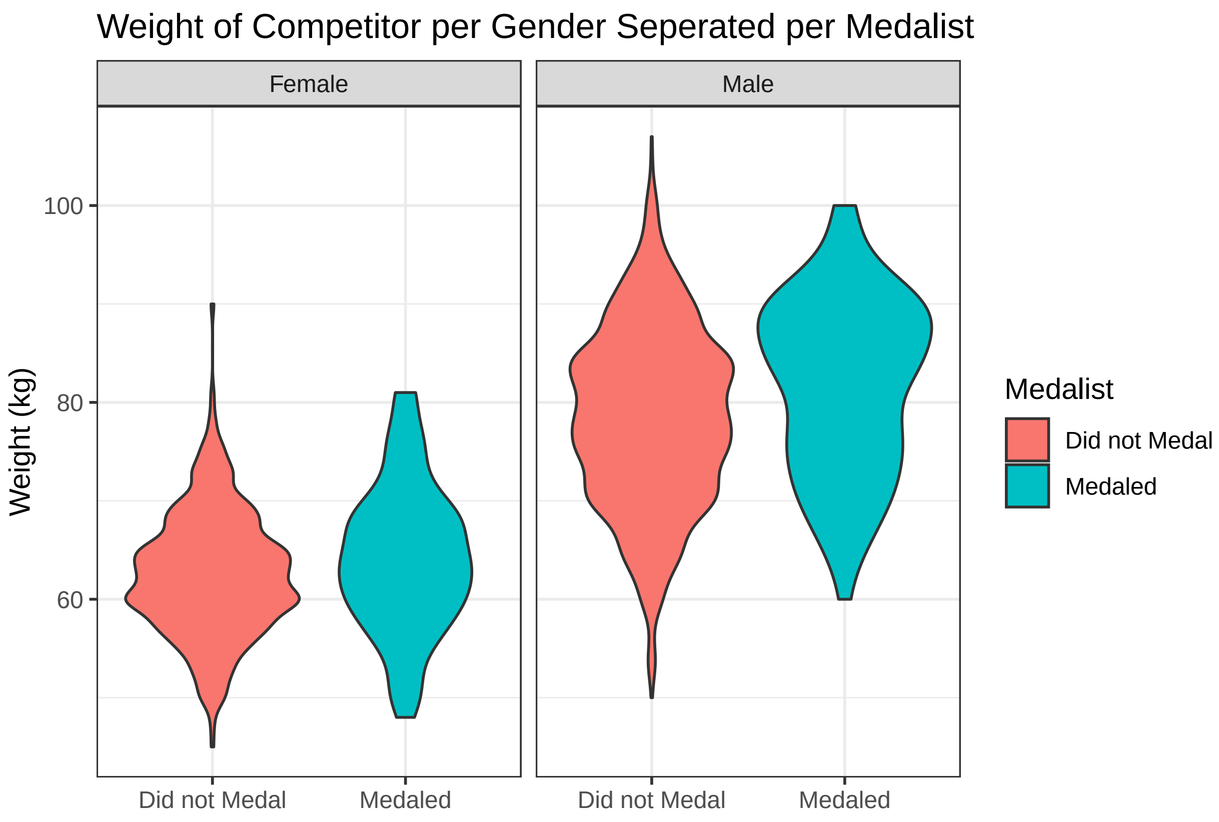
For questions 2 and 3, the approach was similar in that both worked with boxplots as they deal with averages. To answer the question of what is the average weight for each skiing event, the events were separated and then colored by gender, allowing the trends to emerge. Question 3 for the average weight across the games had a similar process, replacing the separation by event and gender to year. However, the data itself was chosen to be just the womens events to minimize variability. The female skier subset was chosen because there were more competitors over a shorter time span to better understand the moving averages. Trend lines for both the moving average and overall average was added for effect as well to show the overall trends.

The last question is made to be a map animation to show the inter connectivity of the world as technology improved. In order to build this map, longitude and latitude is needed which required a supplemental dataset from kaggle listed here: https://www.kaggle.com/datasets/jonscheaffer/olympic-host-cities.

Analysis:

```
# Your R code here
#make new column to have correct axis name
olympics_alpine$Medalist <- gsub("TRUE","Medaled",olympics_alpine$medalist)
olympics_alpine$Medalist <- gsub("FALSE","Did not Medal",olympics_alpine$Medalist)

ggplot(olympics_alpine,aes(x=Medalist,y=weight,fill=Medalist))+
  geom_violin()+
  facet_wrap(vars(sex),
             labeller = as_labeller(c(`M`='Male',`F`='Female')))+
  labs(title = "Weight of Competitor per Gender Seperated per Medalist",
       x = "", y = "Weight (kg)")+
  scale_fill_discrete(name = "Medalist", labels = c("Did not Medal","Medaled"))+
  theme_bw()
```

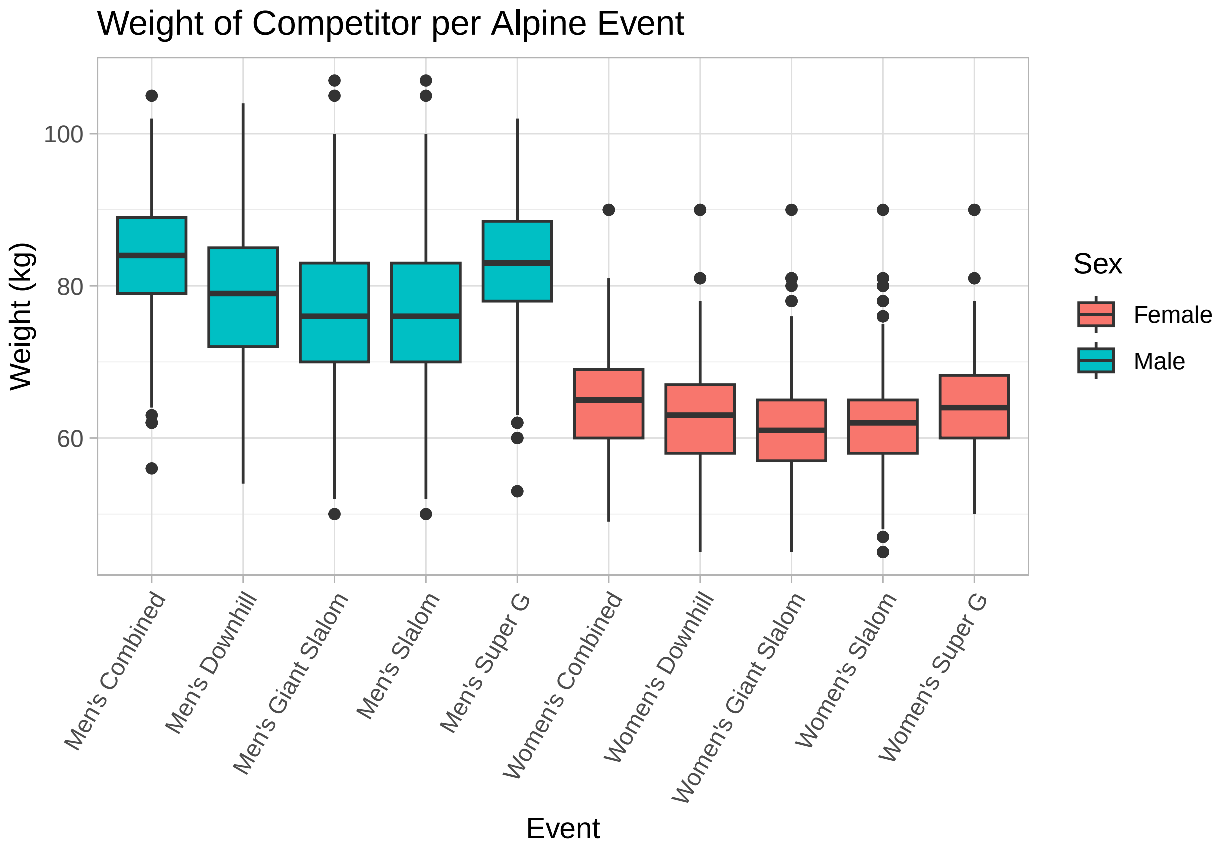


```
model1 <- lm(medalist ~ weight+sex+weight*sex, data=olympics_alpine)
summary(model1)
```

```
##
## Call:
## lm(formula = medalist ~ weight + sex + weight * sex, data = olympics_alpine)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.13908 -0.06746 -0.05406 -0.03933  0.98591
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0911155  0.0444170  -2.051 0.040272 *
## weight       0.0025577  0.0007055   3.626 0.000291 ***
## sexM         0.0009154  0.0550205   0.017 0.986727
## weight:sexM -0.0008195  0.0008161  -1.004 0.315295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2287 on 6346 degrees of freedom
## Multiple R-squared:  0.00722,    Adjusted R-squared:  0.00675
## F-statistic: 15.38 on 3 and 6346 DF,  p-value: 5.712e-10
```

```r
# Your R code here
#making a new dataset to remove apostrophe from event names
olympics_alpine_names <- olympics_alpine
olympics_alpine_names$Event <- gsub('Alpine Skiing ','',olympics_alpine_names$event)


ggplot(olympics_alpine_names,aes(Event,weight,fill=sex))+
  geom_boxplot()+
  labs(title = "Weight of Competitor per Alpine Event",
       x = "Event", y = "Weight (kg)")+
  theme_light()+
  theme(axis.text.x = element_text(angle = 60, vjust = 1, hjust=1))+
  scale_fill_discrete(name = "Sex", labels = c("Female","Male"))
```



```r
# Your R code here
#determine which events have the most competitors
which(table(olympics_alpine$event)>1)
```

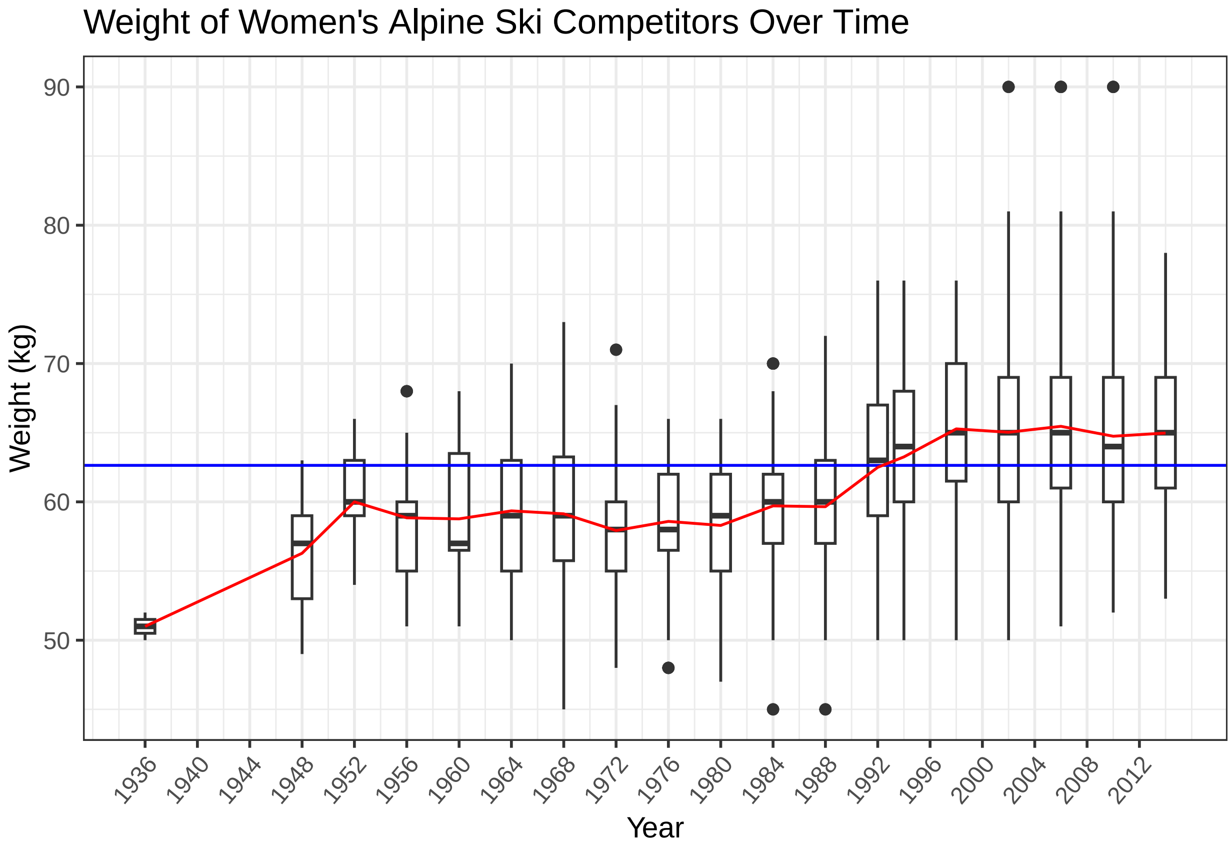
```
##       Alpine Skiing Men's Combined       Alpine Skiing Men's Downhill
##                                  1                                  2
##    Alpine Skiing Men's Giant Slalom         Alpine Skiing Men's Slalom
##                                  3                                  4
##        Alpine Skiing Men's Super G     Alpine Skiing Women's Combined
##                                  5                                  6
##     Alpine Skiing Women's Downhill Alpine Skiing Women's Giant Slalom
##                                   7                                  8
##       Alpine Skiing Women’s Slalom      Alpine Skiing Women’s Super G
##                                   9                                 10
```

```
#new dataset to filter womens events
olympics_alpine_womens <- olympics_alpine
olympics_alpine_womens$event <- gsub("'","",olympics_alpine_womens$event)
olympics_alpine_womens <- olympics_alpine_womens |>
  filter(grepl("Womens", event))

#getting mean weight for each year
mean <- olympics_alpine_womens |>
  group_by(year)|>
  summarize(average = mean(weight)) %>%
  ungroup()


m <- ggplot(olympics_alpine_womens,aes(x=year,y=weight,group=year))+
  geom_boxplot()+
  scale_x_continuous(breaks = round(seq(min(olympics_alpine_womens$year), max(olympics_alpine_womens$yea
  geom_hline(yintercept = mean(olympics_alpine_womens$weight),color='blue')+
  geom_line(data = mean, mapping = aes(x = year, y = average, group=2),color="red")+
  labs(title = "Weight of Women's Alpine Ski Competitors Over Time",
       x = "Year", y = "Weight (kg)")+
  theme_bw()

m+theme(axis.text.x = element_text(angle = 50, vjust = 1, hjust=1))
```



```
# Your R code here
#map of game locations; animate over time adding points
#getting longitude and latitude
longlat <- read_csv('olym.csv')
```

```
## Rows: 51 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): City, Country, NOC, Summer, Winter
## dbl (3): Year, Latitude, Longitude
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
longlat1 <- longlat |> select(6:8)
longlat2 <- data.frame(Year="1906",
                       Latitude="37.98333",
                       Longitude="23.7333336")
longlat3 <- rbind(longlat1,longlat2) |>
  arrange(Year)
longlat3$year <- longlat3$Year

#getting city and country and combining the two dataframes
places <- olympics |> select(7,10:12) |>
  distinct(year,season,city)
```

```r
places_lat <- merge(places,longlat3, by='year') |>
  select(1:3,5,6)
places_lat$Longitude <- as.integer(places_lat$Longitude)
places_lat$Latitude <- as.integer(places_lat$Latitude)

places_lat1 <- places_lat
places_lat1$group <- ifelse(places_lat1$year>1999,'1',ifelse(places_lat1$year>1899 ,'2','3'))


# create data for world coordinates using
# map_data() function
world_coordinates <- map_data("world")
```

```r
# create world map using ggplot() function
# geom_map() function takes world coordinates
# as input to plot world map
p <- ggplot() + geom_map(data=world_coordinates,map=world_coordinates,
                    aes(long,lat,map_id=region))+
  geom_point(data=places_lat1,aes(x=Longitude,y=Latitude,color=city),show.legend = F) +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.title.y=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank())+
  theme_map()+
  guides(color = "none") +
  coord_cartesian(clip = "off") +
  transition_reveal(year)
```

```
## Warning in geom_map(data = world_coordinates, map = world_coordinates,
## aes(long, : Ignoring unknown aesthetics: x and y
```

```r
#the following is for the animation and is commented out to be able to knit
#to pdf

# animate(
#   p,
#   width = 6*params$res,
#   height = 0.618*6*params$res,
#   res = params$res,
#   nframes = params$nframes,
#   fps = params$fps)

#anim_save("JasonBravermanProject1.gif")
```

Discussion: *Your discussion of results here.*

The plots yielded results that were mostly in line with expectations. For the first violin plot, there is a clear trend of men being heavier on average than their female competitors; yet no significant difference in medal earning could be seen visually. A regression analysis corroborated this and showed that there is statistical significance that weight predicts a competitor receiving a medal, but no significance that sex can predict if a competitor will receive a medal. However, there may be some bias in this data since the total number of competitors for each gender is skewed more in mens favor since the first female datapoints occurred in 1936, whereas men's data goes back to 1896. Also, on average men tend to be heavier than their female counterparts; even for the same event as shown in the second plot.

The second plot exposed a similar trend that men, on average, were heavier. The third plot however, where trend lines was added to show averages over time, showed that for the winter Olympic games the mean weight had two consistent time periods: 1948-1988, and 1992-2014. The overall average trend line showed the mean for all competitors to be between those two periods. It showed that as the games progressed, the average weight increased; reaching a maximum in 2006.

The last map plot, did show what was expected, that as humanity became more connected, the games expanded beyond the western world of Europe and America into Australia in the 1950s, East Asia in the 1960s, and South America in 2016.