747_HW2_bcm

Bruce Mallory

3/8/2021

Part1: Warmup

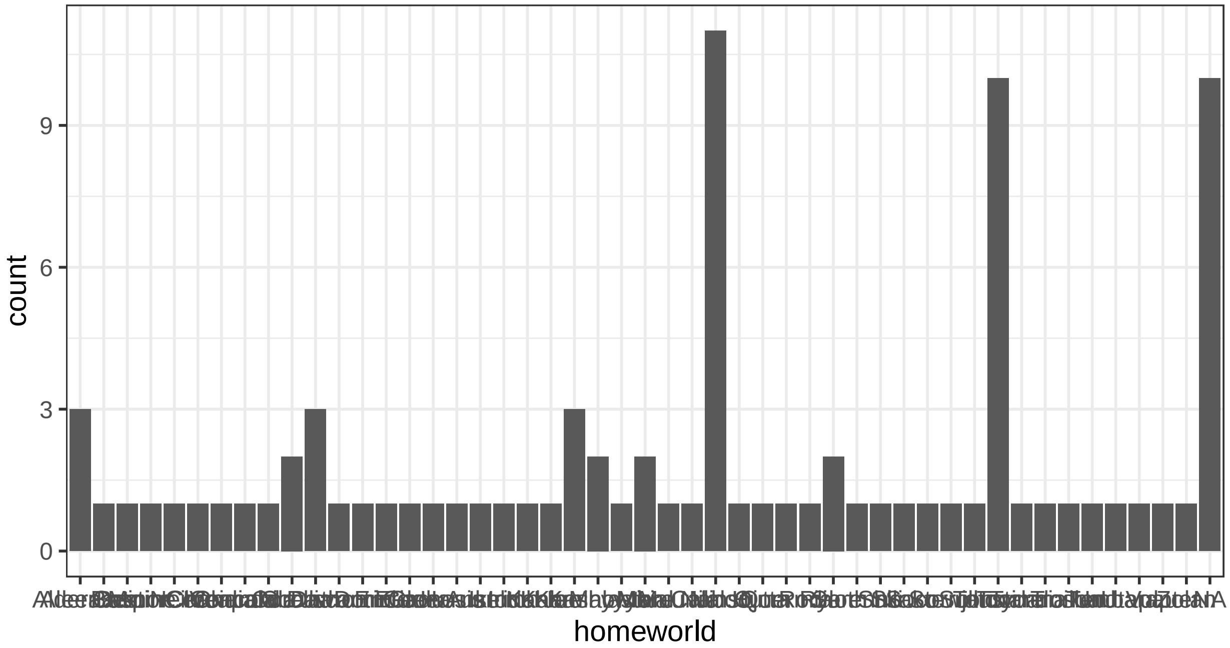
1. Take a look at the starwars dataset. (With dplyr loaded, you can just type "starwars".) Does this dataset follow the guidelines of "tidy" data? Why or why not?

```
# A tibble: 6 x 14
  name  height  mass hair_color skin_color eye_color birth_year sex   gender
  <chr>  <int> <dbl> <chr>      <chr>      <chr>          <dbl> <chr> <chr>
1 Luke~    172    77 blond      fair       blue            19   male  mascu~
2 C-3PO    167    75 <NA>       gold       yellow         112   none  mascu~
3 R2-D2     96    32 <NA>       white, bl~ red             33   none  mascu~
4 Dart~    202   136 none       white      yellow          41.9 male  mascu~
5 Leia~    150    49 brown      light      brown           19   fema~ femin~
6 Owen~    178   120 brown, gr~ light      blue            52   male  mascu~
# ... with 5 more variables: homeworld <chr>, species <chr>, films <list>,
#   vehicles <list>, starships <list>
```

Each character has it's own row. Each measurement has it's own column. So, yes, this dataset follows the 'tidy' guidelines.")

2. Using ggplot, make a bar chart showing how many characters are from each homeworld in the dataset. What is the code, and what does your chart look like? (You can take a screenshot, or use the "Export" drop down menu to save as an image or copy to clipboard.) (Hint: to achieve this, you only need to use ggplot() with geom_bar())

```
attach(starwars)
ggplot(aes(homeworld), data=starwars) +
  geom_bar() +
  theme_bw()
```



```
There are 49 unique homeworlds, including NA.

A table displays this information in a more readable manner.
```

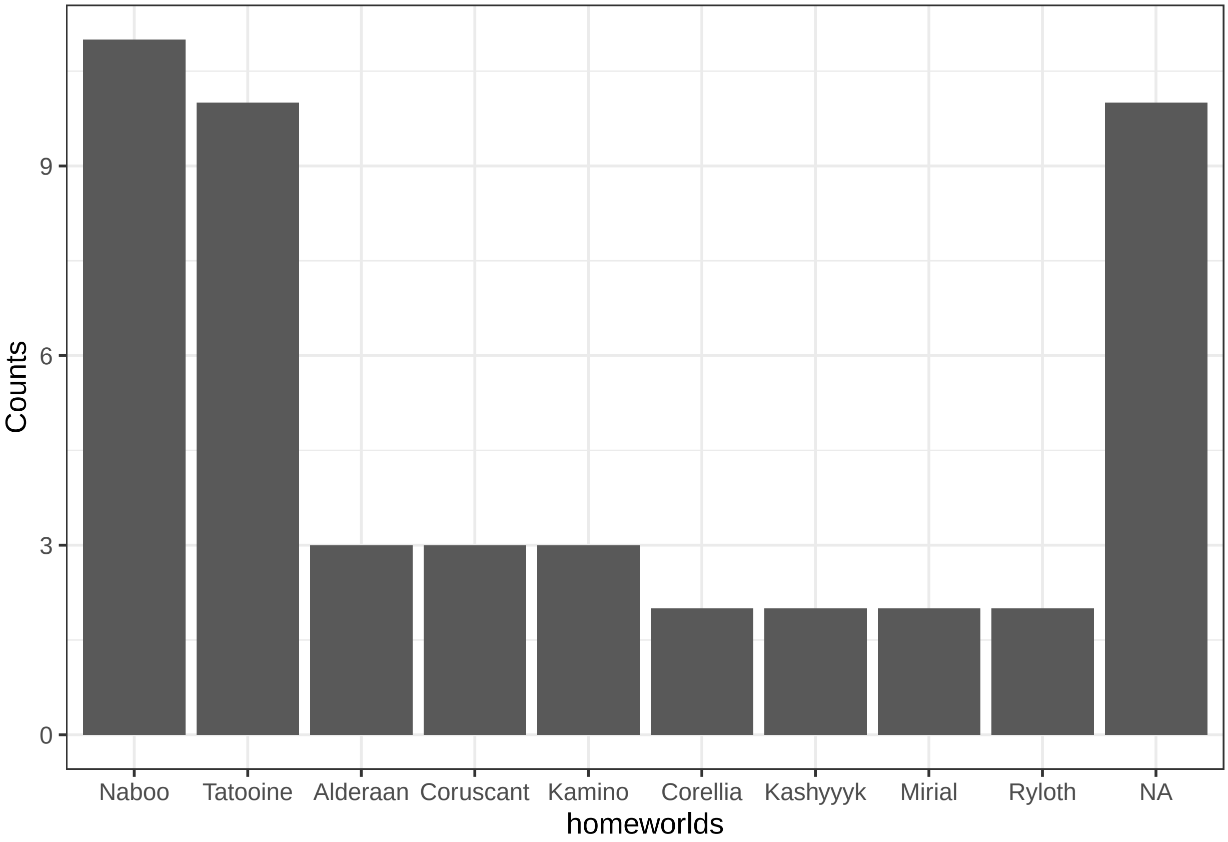
```r
t1 <- data.frame(table(homeworld))
NAs <- data.frame(homeworld="NA", Freq=nrow(starwars[is.na(starwars$homeworld),]))
t1 <- rbind(t1, NAs)
par(mfrow=c(1,3))
k1 <- kable(t1[1:16,], row.names = FALSE)
k2 <- kable(t1[17:33,], row.names = FALSE)
k3 <- kable(t1[34:49,], row.names = FALSE)

k1 <- t1[1:16,]
k2 <- t1[17:32,]
k3 <- t1[33:48,]
k4 <- t1[49,]
kable(list(k1, k2, k3,k4), row.names=FALSE)
```

homeworld	Freq
Alderaan	3
Aleen Minor	1
Bespin	1
Bestine IV	1
Cato Neimoidia	1
Cerea	1
Champala	1
Chandrila	1
Concord Dawn	1
Corellia	2
Coruscant	3
Dathomir	1
Dorin	1
Endor	1
Eriadu	1
Geonosis	1
Glee Anselm	1
Haruun Kal	1
Iktotch	1
Iridonia	1
Kalee	1
Kamino	3
Kashyyyk	2
Malastare	1
Mirial	2
Mon Cala	1
Muunilinst	1
Naboo	11
Nal Hutta	1
Ojom	1
Quermia	1
Rodia	1
Ryloth	2
Serenno	1
Shili	1
Skako	1
Socorro	1
Stewjon	1
Sullust	1
Tatooine	10
Toydaria	1
Trandosha	1
Troiken	1
Tund	1
Umbara	1
Utapau	1
Vulpter	1
Zolan	1
NA	10

3. The chart looks very cluttered, as it is now. If our goal here is to look at homeworlds that lots of Star Wars characters have come from, we might want to filter out planets that are the homeworld of only one character. Reorder the code block to create a bar chart that only includes homeworlds that occur more than once in the dataset.

```
starwars %>%
  group_by(homeworld) %>%
  summarise(count=n()) %>%
  filter(count>=2) %>%
  arrange(desc(count)) %>%
  ggplot(aes(x=reorder(homeworld, -count), y=count)) +
    labs(x="homeworlds", y="Counts") +
    geom_bar(stat="identity") +
    theme_bw()
```



Part2: NHL Data

1. A consequence of the data structure is that if a player changed teams in mid-season, he will appear in two separate rows–one for each team. What step would be necessary to make observations based on player performance over the full season as opposed to player performance on a specific team?

Need to group that player's statistic into one row.

1b. What dplyr function would accomplish this step? Provide a code snippet that would result in a tibble where each row is a player's performance over the whole season.

```
setwd("~/MSSP/747 Social Data Analysis/747_HW2")
hockey <- read_csv("hockey.csv")
hockey <- arrange(hockey, desc(Player))

hockey2char <- hockey %>%
  group_by(Player) %>%
  summarise_if(is.character, funs(paste(., collapse=" & ")))

#unique(hockey2char$General.Position)
hockey2char[hockey2char$General.Position=="D & D",]$General.Position <- "D"
hockey2char[hockey2char$General.Position=="D & D & D",]$General.Position <- "D"
hockey2char[hockey2char$General.Position=="F & F",]$General.Position <- "F"
```

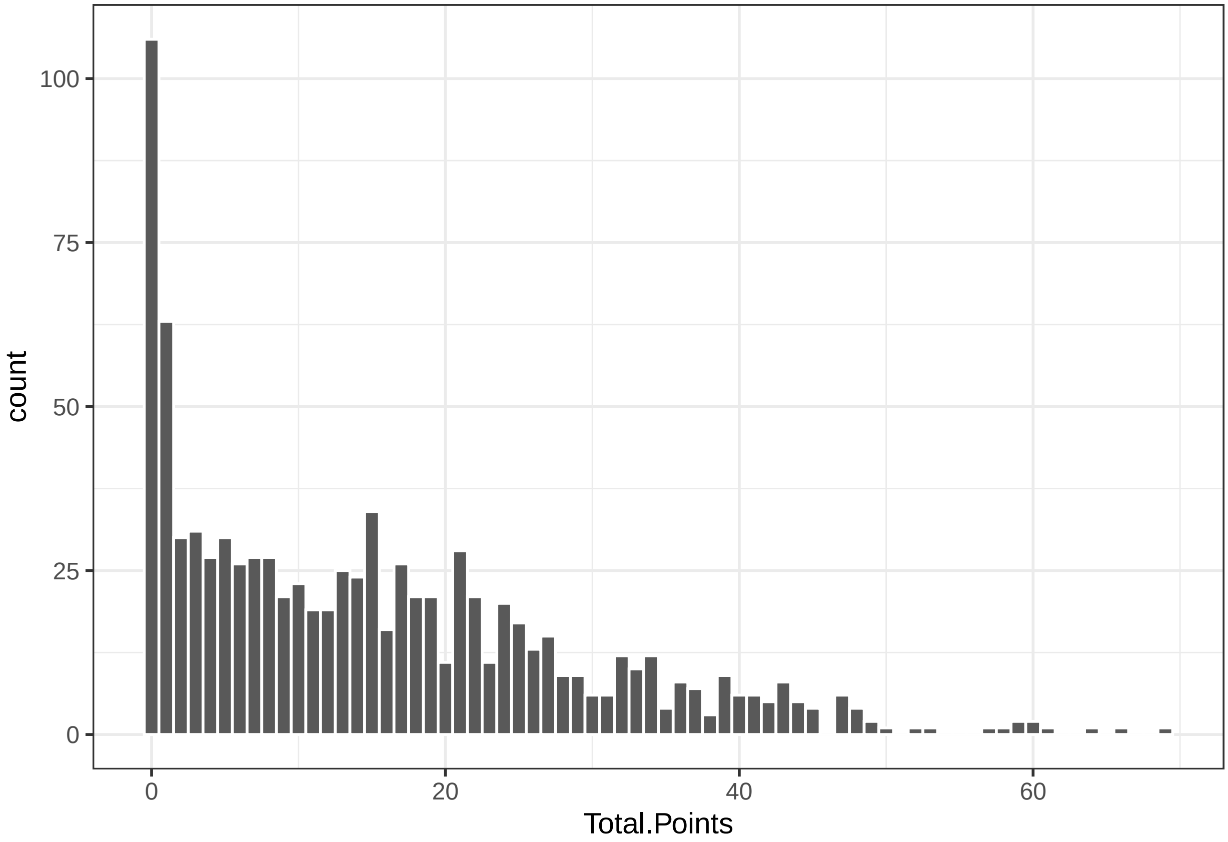
```
hockey2char[hockey2char$General.Position=="F & F & F",]$General.Position <- "F"

hockey2num <- hockey %>%
  group_by(Player) %>%
  summarise_if(is.numeric, funs(sum))

hockey2 <- inner_join(hockey2char, hockey2num, by="Player")
#Is there a more elegant way to do this than what I just did?
```

2a. Keeping in mind that we want to look at each player's season-level performance (i.e., keep using your code from 1b), create a histogram of the distribution of total points for all players (i.e. bars should represent the number of players achieving a certain number of points). Report the code used to create the histogram. What do you notice about the distribution of the histogram?

```
ggplot(hockey2, aes(Total.Points)) +
  geom_histogram(binwidth = 1, color="white") +
  theme_bw()
```



```
no_points <- round(100*sum(hockey2$Total.Points==0)/nrow(hockey2), 2)
high_pointers <- round(100*sum(hockey2$Total.Points>50)/nrow(hockey2), 2)
high_points <- round(100*sum(hockey2[hockey2$Total.Points>50, ]$Total.Points)/sum(hockey2$Total.Points
less_mean <- round(100*sum(hockey2$Total.Points<mean(hockey2$Total.Points))/nrow(hockey2), 2)
skw <- skewness(hockey2$Total.Points)
```

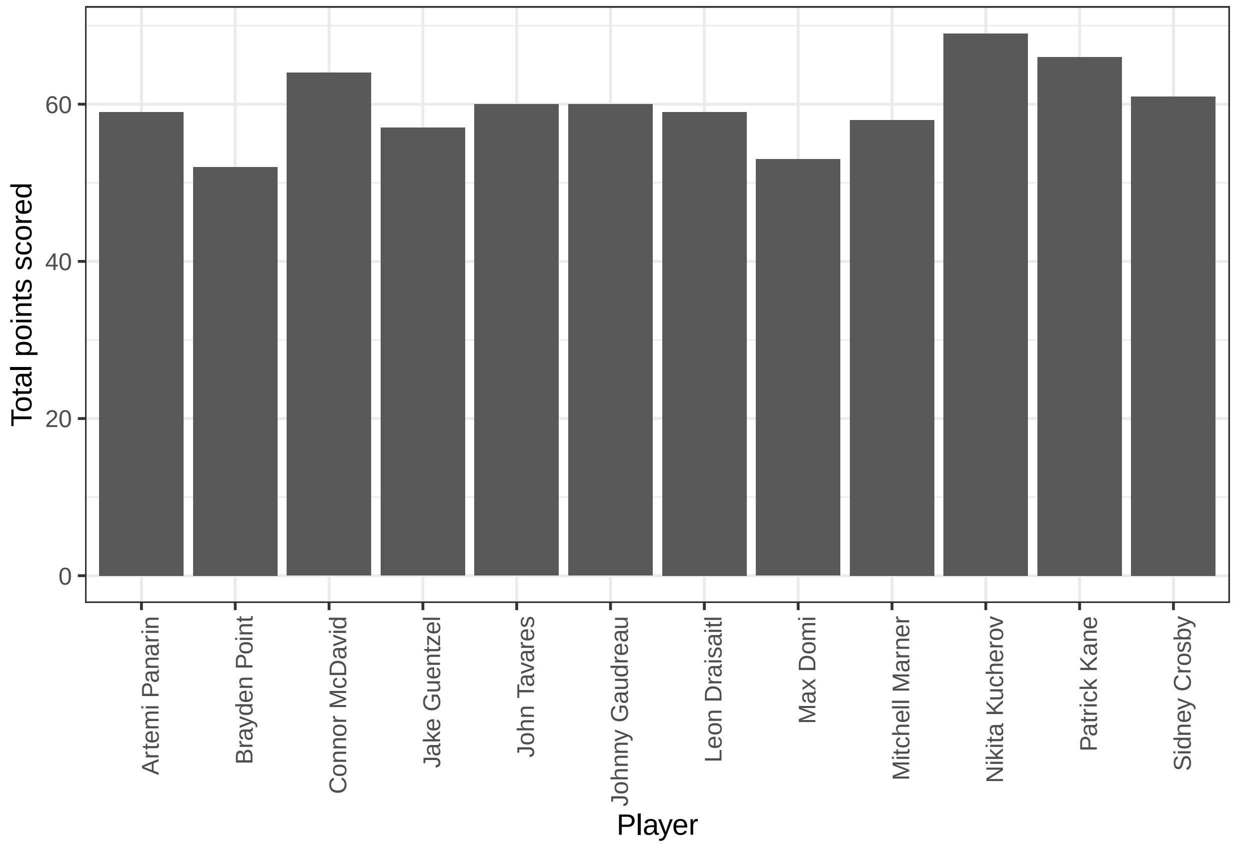
The distribution is skewed right. With 11.7% of the players scoring no points at all during the season, and 58.7%% of the players with points totals that were less than the mean. The distribution is moderately skewed with *skewness* = 0.98.

2b. Choose a reasonable cut-off point that will help you identify who the very highest-scoring players are. What code would return the list/table of these players?

```
high_scorers <-  hockey2[hockey2$Total.Points>50,]
```

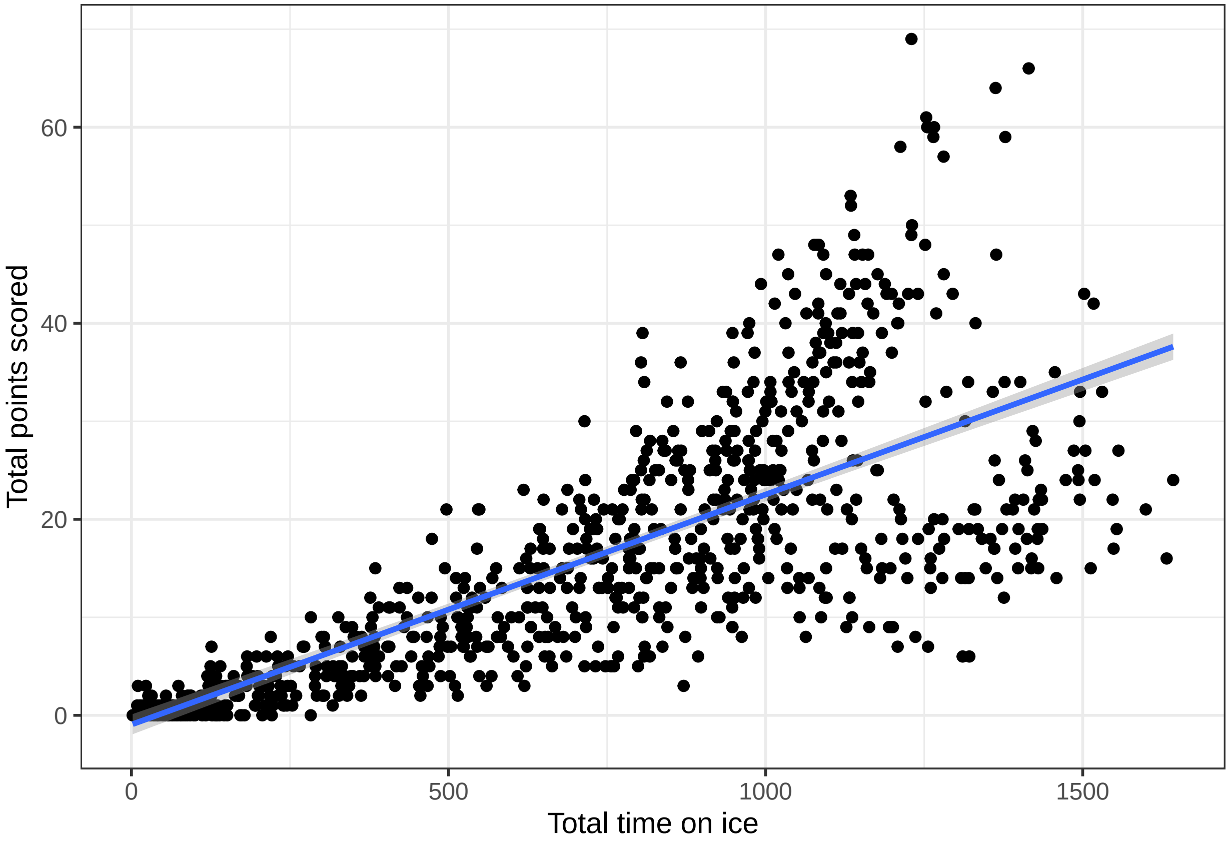
2c. Create a bar chart with only these players: the players' names should be on the x-axis, and the height of hte bars should indicate the total points they scored.

```
ggplot(high_scorers, aes(Player)) +
  geom_bar(aes(weight=Total.Points)) +
  labs(y="Total points scored") +
  theme_bw() +
  theme(axis.text.x=element_text(angle=90,hjust=1))
```



2d. There are 82 games in a hockey season, and you might notice that many of the players with high total points have played almost all of them and have over 1000 minutes of time on ice. Maybe you are interested in the relationship between points and time on ice. Make a scatterplot comparing total points to total time on ice. Report your code. What basic relationship do you observe? What are 2 different causal explanations that could explain this relationship?

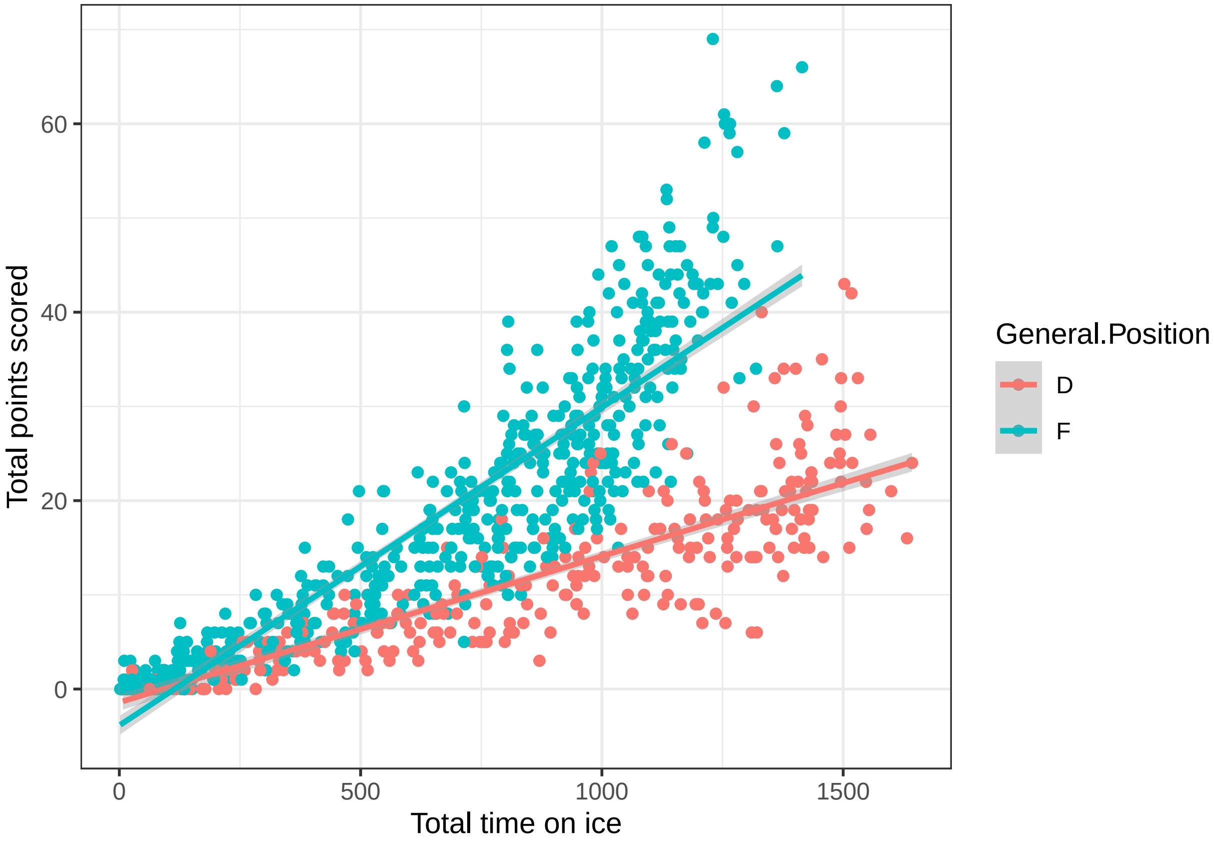
```
ggplot(hockey2, aes(TOI, Total.Points)) +
  geom_point() +
  geom_smooth(method='lm', se=TRUE) +
  labs(x="Total time on ice", y="Total points scored") +
  theme_bw()
```



There are two general patterns here:
First, there is a positive association between the total time on the ice and the number of points scored. Makees sense, the more time on ice, the more opportunities. And the better scorers will be given more time to be on the ice.
The second pattern is the heteroscedasticity of the data. You see this in the plot as the variation in the number of points scored increases as the player spends more time on the ice. This also makes sense, because for the not-on-ice-much-group there isn't a big range of points possibilities. The minimum is close to the average, and there is only so many point you can score in a limited time on ice.
The third interesting pattern is that there appear to be two groups of scorers on the ice for more than 1,000 minutes. There are high scorers (the forwards), and the low scorer's (the defense).

2e. Interesting… but hockey is a team sport, which means different players have different roles. Now build on your scatterplot: using the same x and y, now use color to indicate what position each player plays on the ice (use the variable General.Position, not Position). Report your code and briefly explain how this change helps you understand the relationships in the graph. (Hint: in the variable General.Position, D stands for Defense and F stands for Forward.)

```
ggplot(hockey2, aes(TOI, Total.Points, color=General.Position)) +
  geom_point() +
  geom_smooth(method='lm', se=TRUE) +
  labs(x="Total time on ice", y="Total points scored") +
  theme_bw()
```



2f. Now for a quick sub-set. When exploring data, we often want to be able to zoom in on datapoints that look especially different or interesting. Here, you might be interested in those 3 defensemen who scored quite a few points. If you wanted to zoom in on these individuals, and created a table that showed those three players (in rows) and their team, total points and time on ice, what dplyr commands would you use?

```
high_score_defense <- hockey2 %>%
  filter(General.Position=="D", Total.Points > 38) %>%
  select(Player, Team, Total.Points, TOI)
kable(high_score_defense, align="llcc")
```

Player	Team	Total.Points	TOI
Brent Burns	S.J	42	1517.250
Mark Giordano	CGY	40	1331.017
Morgan Rielly	TOR	43	1502.333

3. Maybe you don't really care about particular NHL players, you want to compare the performance of teams. Say we want to create a bar chart indicating the total goals scored by each team, another indicating the average goals scored by each player on each team, and a box plot so we can understand the distribution of goals on a team level.

3a. First, we want to think about what we want our graph to look like. If we want to compare teams with each other based on the all of the goals they have scored, what will be measured on the x-axis and y-axis of the chart?

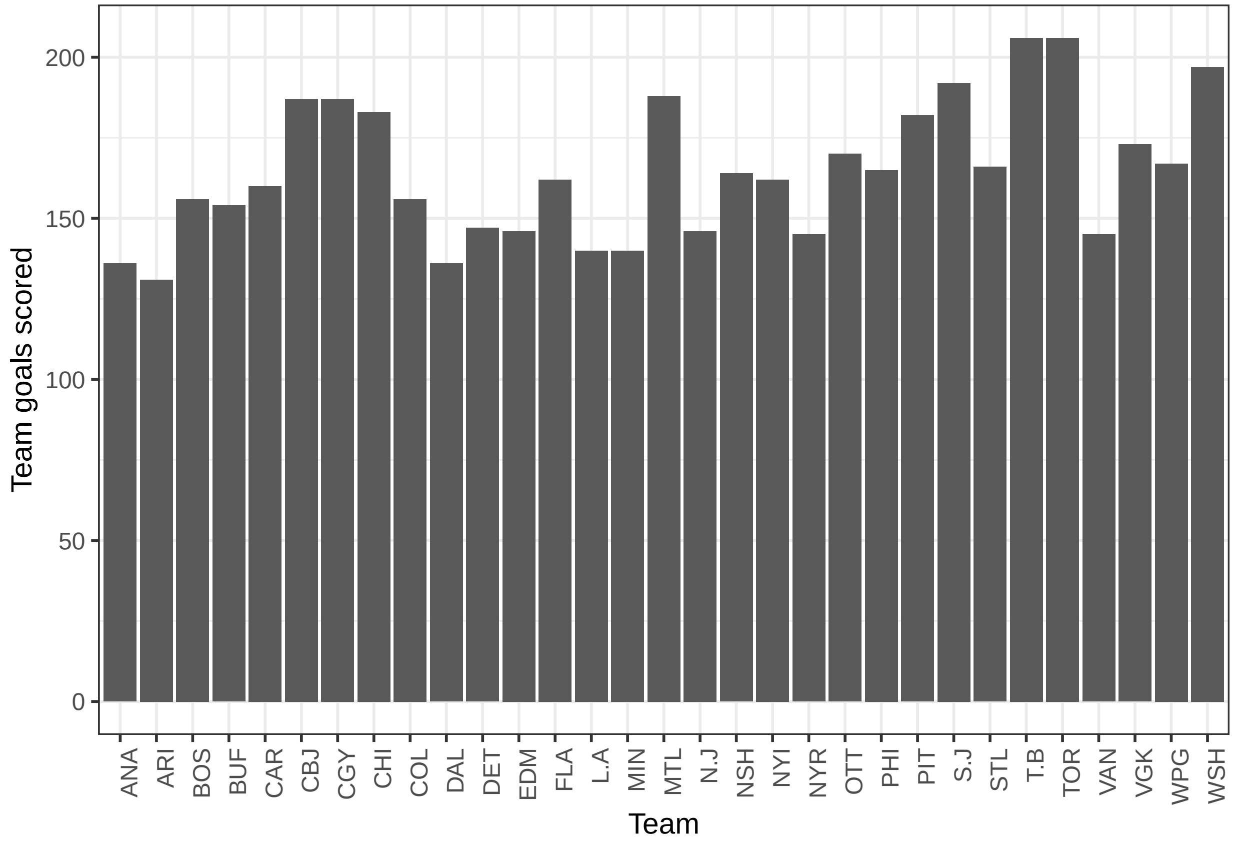
`Team` on the x-axis, and `Goals` on the y-axis.

3b. To begin setting up your data, use group_by and summarise to create a tibble with teams as rows and a value for each row showing how many goals the team scored.

```
teams <- hockey %>%
  group_by(Team) %>%
  summarise_if(is.numeric, funs(sum))
```

3c. Now, pipe this expression into ggplot and create a bar chart with the correct x and y axes, keeping in mind the default statistic used in geom_bar, and how to change it. (Accuracy check: There should be 32 bars on the graph, one for each team.)

```
ggplot(teams, aes(Team)) +
  geom_bar(aes(weight=Goals)) +
  labs(y="Team goals scored") +
  theme_bw() +
  theme(axis.text.x=element_text(angle=90,hjust=1))
```

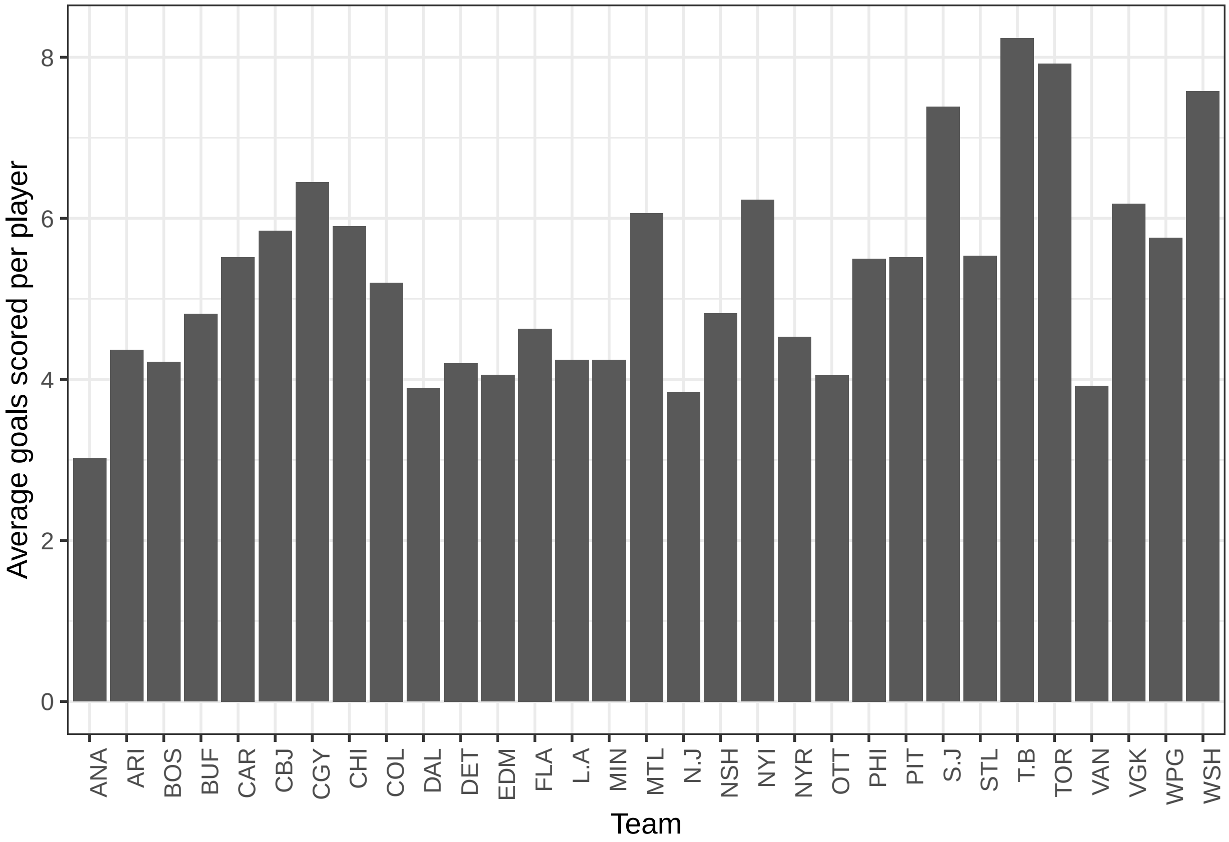


```
cat("Accuracy check: length(unique(hockey$Team)) = ", length(unique(hockey$Team)), "What team did I mis
```

```
Accuracy check: length(unique(hockey$Team)) =  31 What team did I miss?
```

3d. Now let's take a look at average goals scored per player. You can use the code you produced for 3c, and make a minor alteration to display the mean number of goals scored by members of each team.

```
hockey %>%
  group_by(Team) %>%
  summarise_if(is.numeric, funs(mean)) %>%
  ggplot(aes(Team)) +
  geom_bar(aes(weight=Goals)) +
  labs(y="Average goals scored per player") +
  theme_bw() +
  theme(axis.text.x=element_text(angle=90,hjust=1))
```



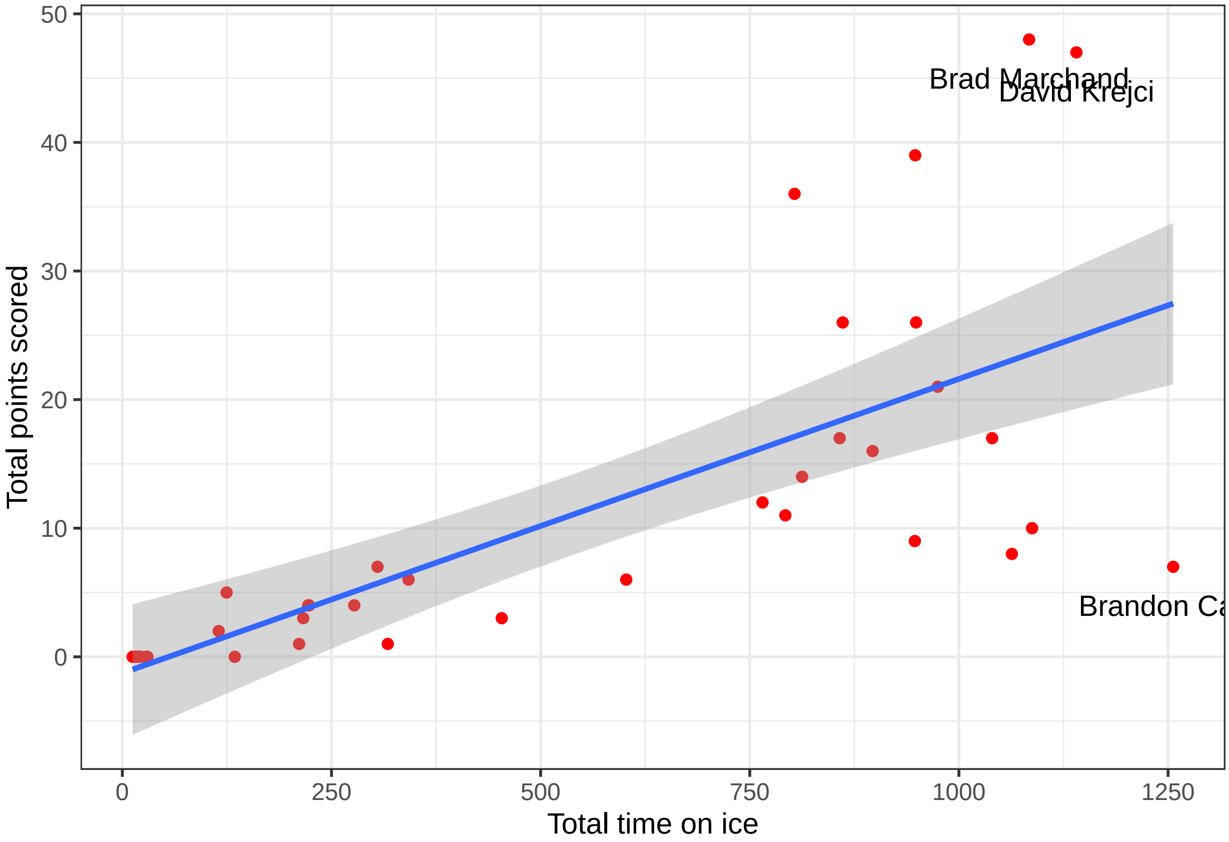
4a. Suppose you don't care about the rest of the NHL, the only thing that really matters to you is the Boston Bruins. Report the code used to make a tibble with only players from the Boston Bruins (BOS) included.

```
BOS <- hockey %>%
  filter(Team=="BOS")
```

4b. Using pipes (%>%), report the code necessary to go from the original dataset to a scatterplot of Boston Bruins players' total points versus time on ice.

4c. Building on Healy's suggestions for datapoint labeling, choose three of the more notable datapoints from 4b and label them with the player's names.

```
hockey %>%
  filter(Team=="BOS") %>%
  ggplot(aes(TOI, Total.Points)) +
  geom_point(col="red") +
  geom_smooth(method='lm', se=TRUE) +
  labs(x="Total time on ice", y="Total points scored") +
  theme_bw() +
  geom_text(data=subset(BOS, Total.Points>40 | TOI > 1250),
            aes(TOI,Total.Points,label=Player), nudge_y=-3)
```



```
#Could NOT get nudge_x to work - was trying to get Brandon's whole name
```

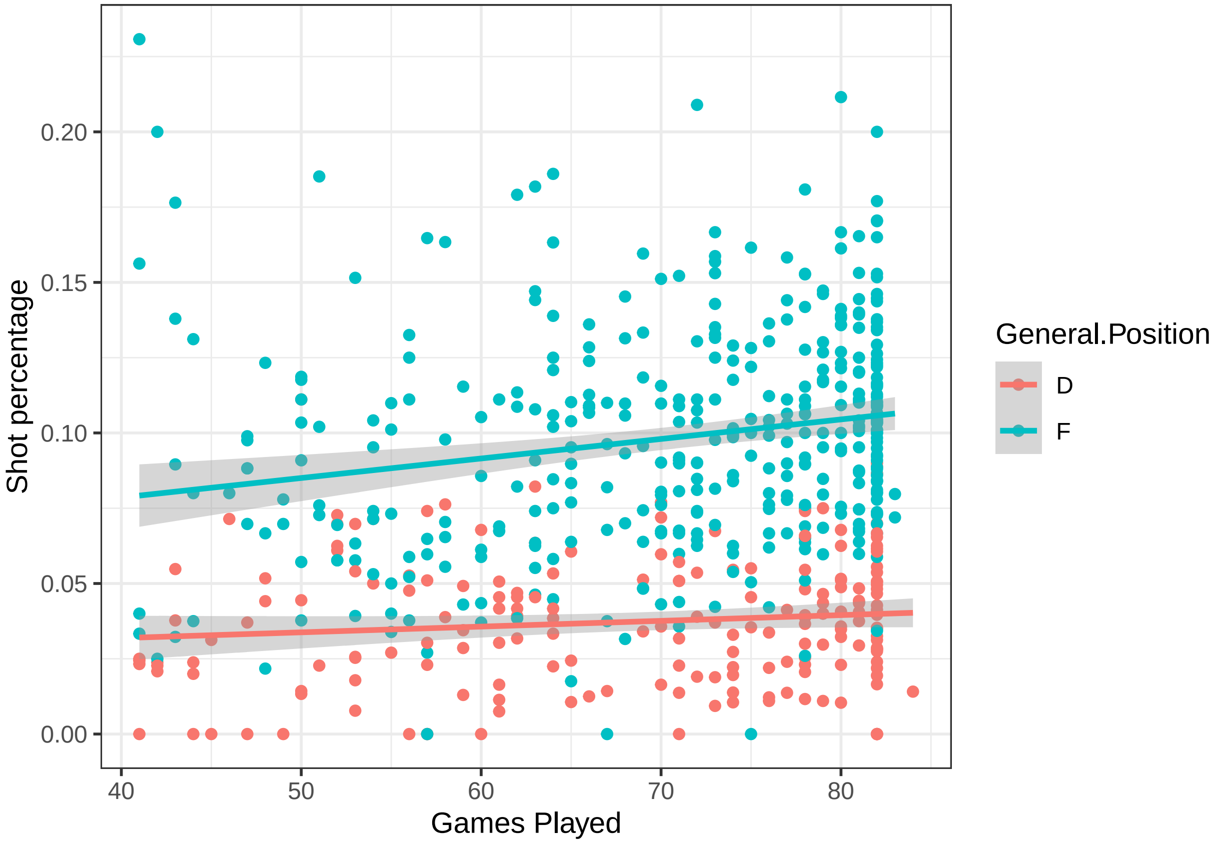
5. Total points are one way to think about player skill. Another might be the accuracy of player shots. Percentages are common measures of accuracy, usually calculated as the proportion of attempts that succeeded. In the case of hockey, shot percentage might be a useful measure. There is no measure of shot percentage in the current data, but you can create it:

5a. Use the variables Shots and Goals to create a new variable, shot percentage. What you want to calculate is the percentage of shots that scored goals. (Hint: mutate() could be useful).

```
#hockey2 %>% mutate(Shot.Prop=Goals/Shots)
```

5b. Let's graph that new variable somehow. Pipe your result from 5a into a ggplot() statement. You choose what you want to graph with it: you could graph it on its own as a histogram or bar plot, or you could use another variable to create a scatterplot. What have you found?

```
hockey2 %>%
  mutate(Shot.Prop=Goals/Shots) %>%
  filter(GP>40) %>%
  ggplot(aes(GP, Shot.Prop, color=General.Position)) +
  geom_point() +
  labs(x="Games Played", y="Shot percentage") +
  geom_smooth(method='lm', se=TRUE) +
  theme_bw()
```



For players who play in a non-trivial number of games, there is a slight improvement in the accuracy the more games they play in (which is probably causal the other direction, in that the good shooters get more game time). There is also a clear difference between the accuracy of the forward (higher) and the defense (lower). AND the variability of the linear regression estimate gets smaller the more games (the bigger n is) that the player plays.