Homework 2

Your Name

Spring 2022

```
knitr::opts_chunk$set(echo = TRUE,
                      fig.width=10,
                      fig.height=6,
                      fig.align = "center")


# Load the needed package(s) below:
library(readr)
library(ggplot2)
library(dplyr)
library(gridExtra)
library(tidyr)
library("RColorBrewer")


# Change the default theme below:
theme_set(theme_bw())
```

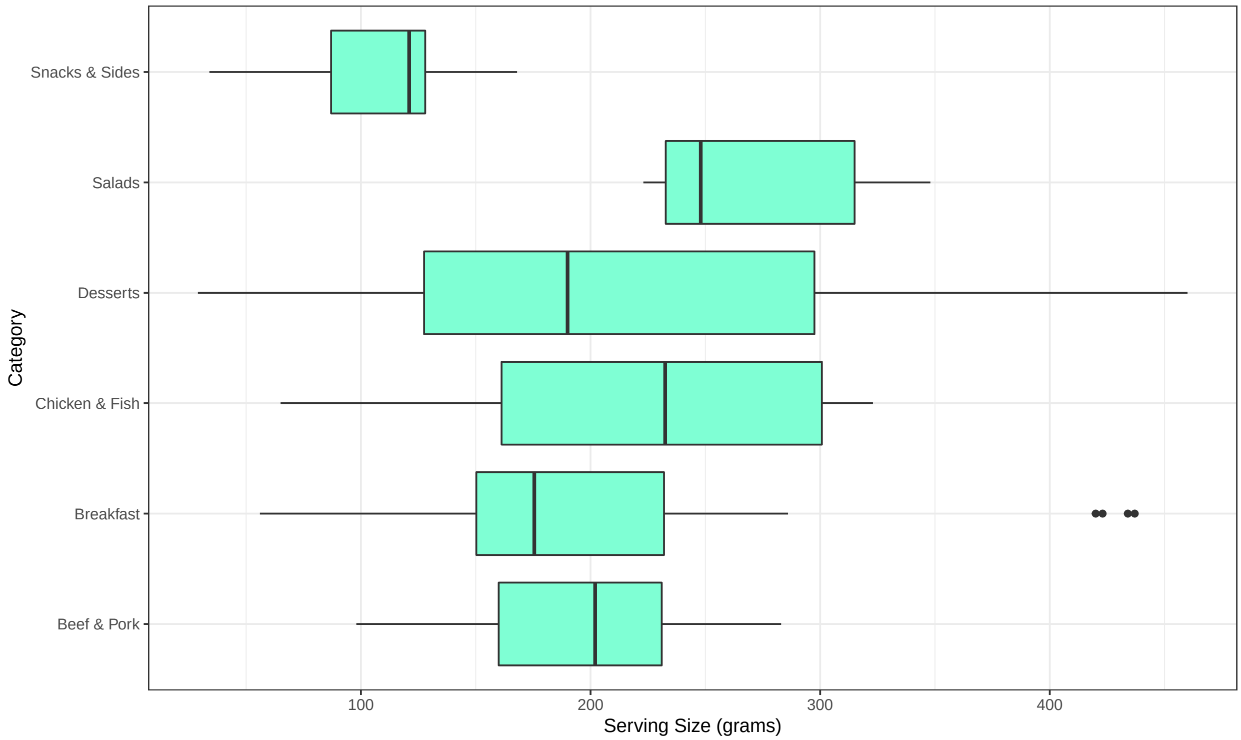
Part 1: McDonald's Nutrition

The "fast food menu.csv" data set contains the nutritional information on many different non-drink options offered at McDonald's. Use an appropriate theme for the graphs.

1a) Calories by serving size

Create horizontal box plots of serving_size_grams by Category. Fill the box plots with all the same color, but use a different color than the default. Change the label of the x-axis to "Serving Size (grams)". Which Category has the largest and smallest serving sizes overall?

```
# Start by reading in the data here.
# Name the data set McD
McD <- read.csv("C:/Users/huaye/Desktop/CS 187A/HW/HW_2/fast food menu.csv")
View(McD)

# Create the boxplot below
ggplot(data=McD,mapping=aes(x=serving_size_grams,y=Category))+
  labs(x="Serving Size (grams)")+
  geom_boxplot(fill="aquamarine")
```



```
## Salads have largest serving sizes and snacks&sides have the smallest serving size overall.
```

1b) Calories by Category, Serving Size, and Nutritional Info

Create and SAVE create 6 similar scatterplots with:

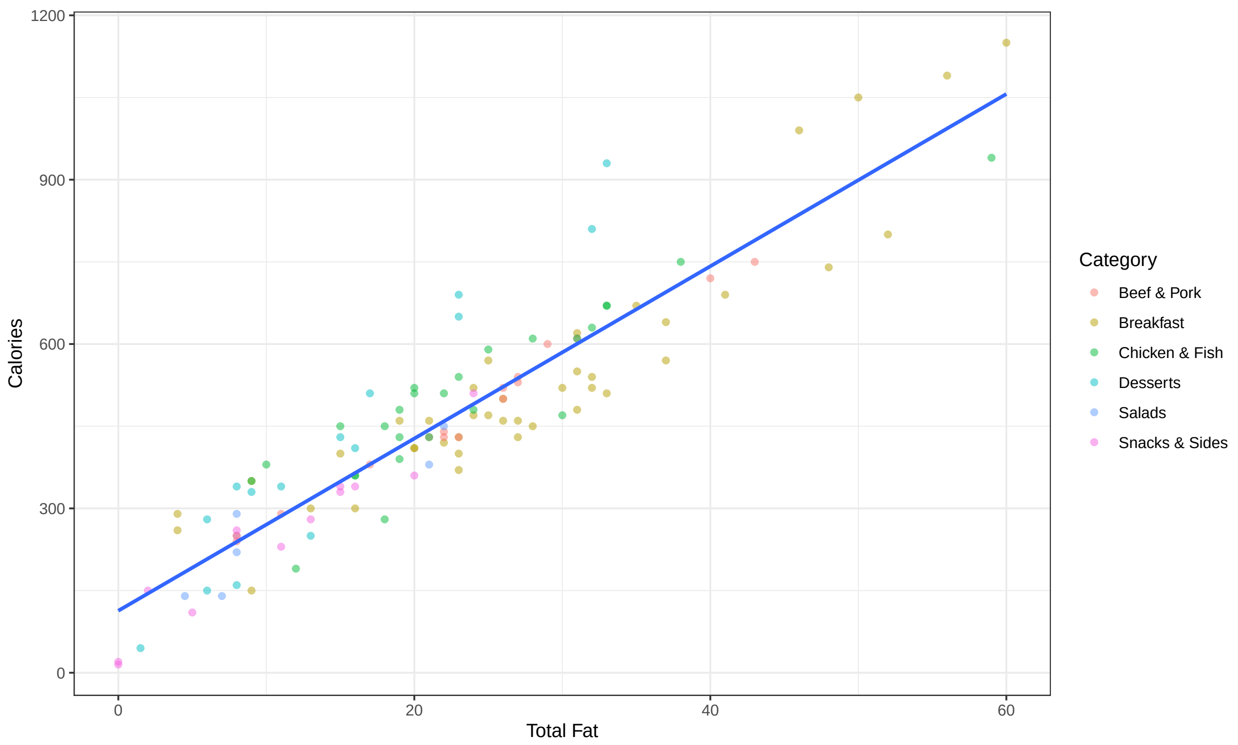
- Calories on the y-axis
- Category indicated by color
- serving_size_grams represented by size.
- Each graph should have points and a single (straight) regression line, without the shaded region.
- In geom_point, include alpha = 0.50 to help combat overplotting
- add + guides(size = "none") to each of the six plots to hide the legend for the size aesthetic
- Change the labels on the axes & legend to be more readable, if necessary

The x-axis for each of the 6 scatterplots is:

i. Total Fat
ii. Saturated Fat
iii. Cholesterol
iv. Sodium
v. Protein
vi. Sugars

```
# Create, save, and display the scatterplot with x = Total Fat below

p1<-ggplot(data=McD,mapping=aes(x=Total.Fat,y=Calories))+
  labs(x="Total Fat",y="Calories")+
  geom_point(aes(color=Category),alpha=0.50)+
  geom_smooth(method=lm,formula=y~x,se=0)+
  guides(size = "none")

p1
```

Part 1bi. Total Fat



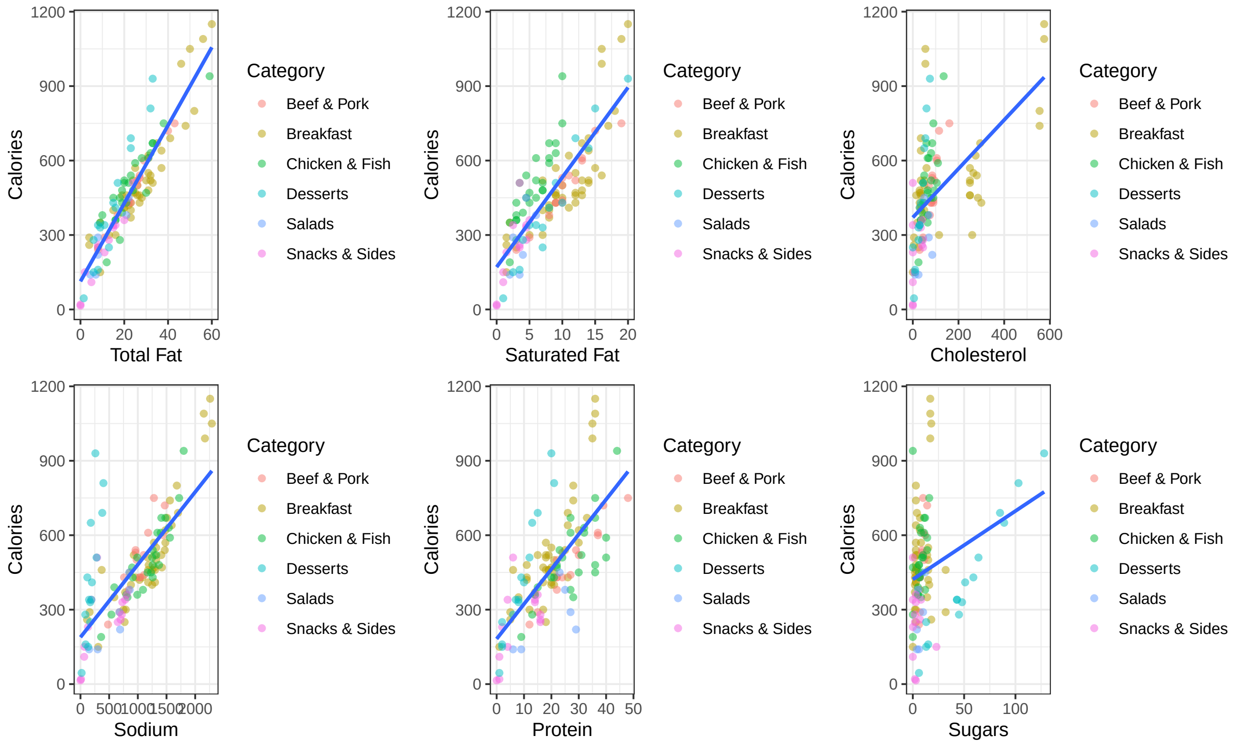
```
## Create and save but don't display the scatterplot with

# ii) x = Saturated Fat below

p2<-ggplot(data=McD,mapping=aes(x=Saturated.Fat,y=Calories))+
  labs(x="Saturated Fat",y="Calories")+
  geom_point(aes(color=Category),alpha=0.50)+
  geom_smooth(method=lm,formula=y~x,se=0)+
  guides(size = "none")



# iii) x = Cholesterol below

p3<-ggplot(data=McD,mapping=aes(x=Cholesterol,y=Calories))+
  labs(x="Cholesterol",y="Calories")+
  geom_point(aes(color=Category),alpha=0.50)+
  geom_smooth(method=lm,formula=y~x,se=0)+
  guides(size = "none")


# iv) x = Sodium

p4<-ggplot(data=McD,mapping=aes(x=Sodium,y=Calories))+
  labs(x="Sodium",y="Calories")+
  geom_point(aes(color=Category),alpha=0.50)+
  geom_smooth(method=lm,formula=y~x,se=0)+
  guides(size = "none")


# v) x = Protein below

p5<-ggplot(data=McD,mapping=aes(x=Protein,y=Calories))+
  labs(x="Protein",y="Calories")+
  geom_point(aes(color=Category),alpha=0.50)+
  geom_smooth(method=lm,formula=y~x,se=0)+
  guides(size = "none")


# vi) x = Sugars below

p6<-ggplot(data=McD,mapping=aes(x=Sugars,y=Calories))+
  labs(x="Sugars",y="Calories")+
  geom_point(aes(color=Category),alpha=0.50)+
  geom_smooth(method=lm,formula=y~x,se=0)+
  guides(size = "none")
```

Part 1bii. - vi. Other 5 scatterplots

Part 1c)

Present the 6 scatterplots using grid.arrange(), as described on page 1. Put all six together with 3 rows.

```
# Use grid.arrange below to place all 6 scatter plots together

grid.arrange(p1,p2,p3,p4,p5,p6,nrow=2)
```



Part 1d) Long Format

Run the code in the chunk below to transform the data into a "long" format.

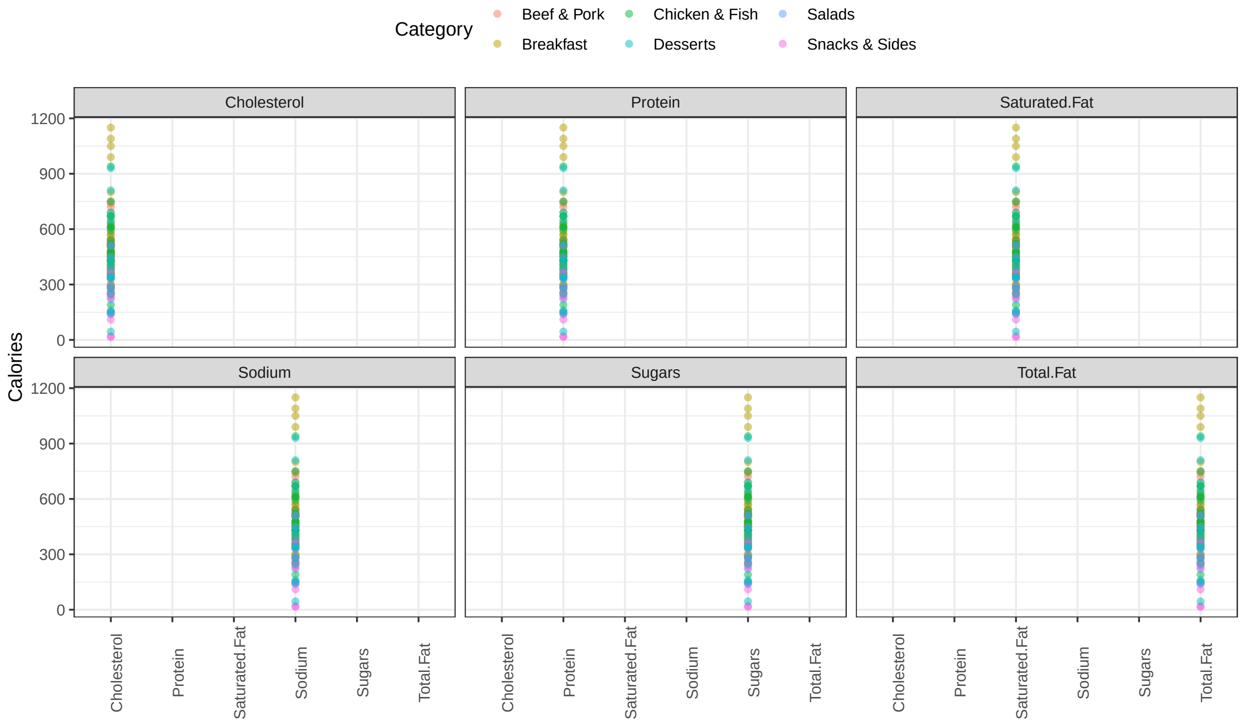
```
# Using pivot_longer to create a new data set in the "long" format
McD_long <-
  McD %>%

  dplyr::select(Category, serving_size_grams, Calories, Total.Fat,
                Saturated.Fat, Cholesterol, Sodium, Protein, Sugars) %>%

  pivot_longer(cols = Total.Fat:Sugars,
               names_to = "nutrition_type",
               values_to = "amount") %>%

  mutate(nutrition_type = as.factor(nutrition_type))

McD_long
```

```
## # A tibble: 702 x 5
##    Category  serving_size_grams Calories nutrition_type amount
##    <chr>                  <int>    <int> <fct>           <dbl>
##  1 Breakfast                136      300 Total.Fat          13
##  2 Breakfast                136      300 Saturated.Fat       5
##  3 Breakfast                136      300 Cholesterol       260
##  4 Breakfast                136      300 Sodium            750
##  5 Breakfast                136      300 Protein            17
##   6 Breakfast                 136       300 Sugars           3
##   7 Breakfast                 135       250 Total.Fat        8
##   8 Breakfast                 135       250 Saturated.Fat    3
##   9 Breakfast                 135       250 Cholesterol     25
##  10 Breakfast                 135       250 Sodium         770
## # ... with 692 more rows
```

Using McD_long, create the same six scatterplots from part b), but using facet_wrap instead of grid.arrange.

You should only need 1 ggplot function, and use 3 rows again. Make sure to use a good choice for the scales argument inside facet_wrap.

```
# Use the transformed data and facet_wrap to create a similar graph to 1c)

ggplot(data=McD_long,mapping=aes(x=nutrition_type,y=Calories))+
  geom_point(aes(color=Category),alpha=0.50)+
  facet_wrap(vars(nutrition_type))+
  geom_smooth(method=lm,formula=y~x,se=0)+
  theme(legend.position = "top",axis.text.x = element_text(angle = 90)) +
  labs(x = NULL)
```



Part 1e)

Which set of graphs do you prefer? Describe why

I prefer the first set of graphs since there are more details in it. It appears that although the codes are simple and convenient for facet_wrap, there are some details that can be hard for observation when we use it for plots which contains lots of details(for example, it is hard to see the regression line since the graph is small and too close to each other). But I do think it can be good for other types of plot when there are less details required.

Question 2: NFL Drive Data

Use the NFL Drive data set contains how an NFL drive begins and ends for 30374 drives for the 2016 – 2021 NFL seasons. A "drive" in football is a series of plays where the same team is on offense and ends when the teams switch being on offense or defense.

```
### Import the NFL Drive data set below:
nfl <-  read_csv("C:/Users/huaye/Desktop/CS 187A/HW/HW_2/NFL Drive(1).csv")
```

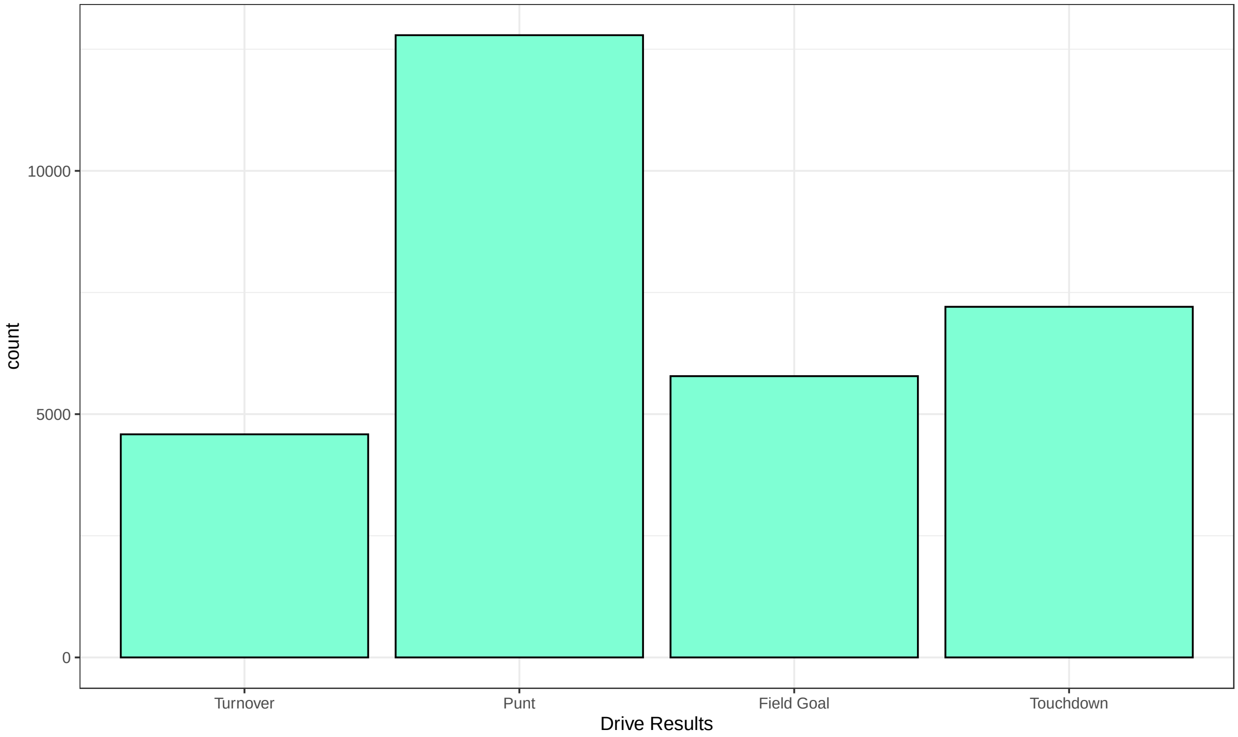
```
## Rows: 30374 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): drive_id, drive_start, drive_end
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
View(nfl)


# Run the code below after reading in your data to change the order of the groups to something more pre
nfl <-
  nfl %>%

  filter(complete.cases(.)) %>%

  mutate(drive_start = factor(drive_start,
                              levels = c("Kickoff", "Punt",
                                         "Fumble", "Interception")),

         drive_end = factor(drive_end,
                            levels = c("Turnover", "Punt",
                                       "Field Goal", "Touchdown")))
```

Part 2a) Bar Chart for Drive End

Create a single bar chart drive_end with the counts for each outcome on the y-axis. Have the bars all be the same color other than the default color and the x-axis label of "Drive Result".

```
# Create the bar chart for drive_end below.

ggplot(data=nfl,mapping=aes(x=drive_end))+
  geom_bar(fill="aquamarine",color="black")+
  labs(x="Drive Results")
```

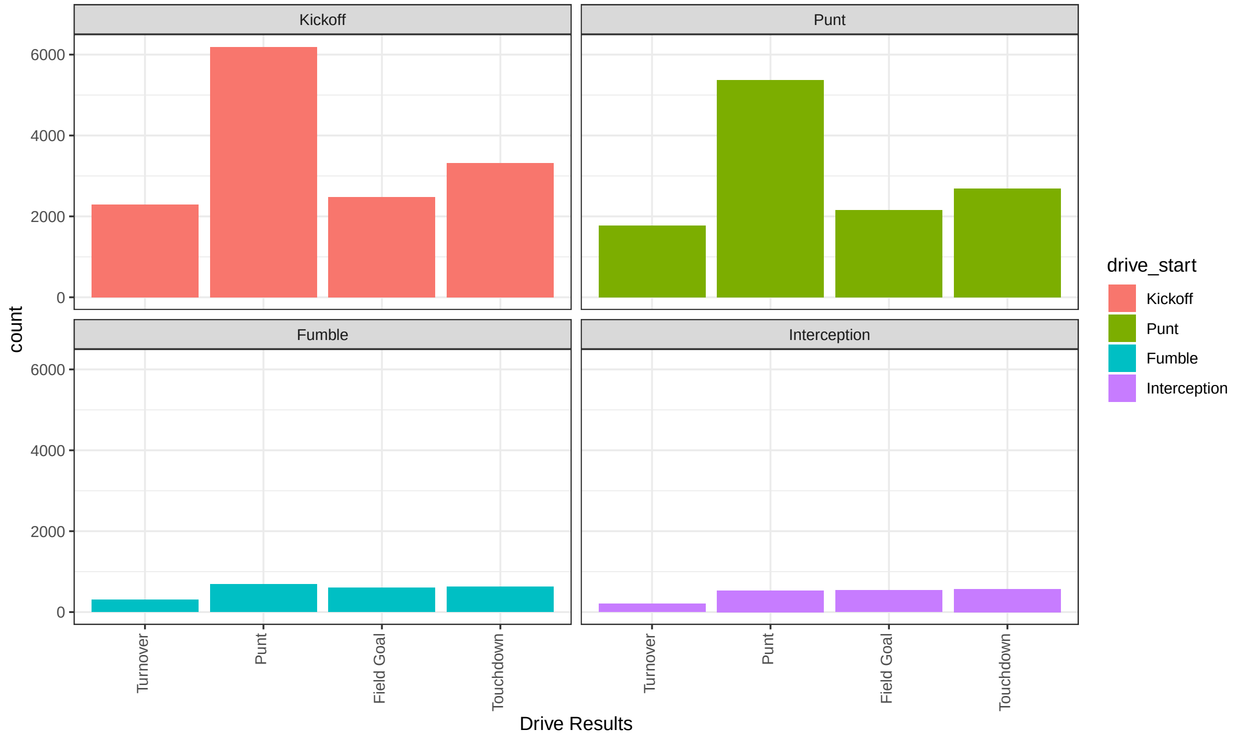


Part 2b) Bar Chart by Drive Beginning

Create 4 bar charts for drive_end: One for each way a drive can begin (drive_start). Have the proportion displayed on the y-axis instead of the counts and all 4 charts in the same row.

Adjust the plots so the x-axis labels can be seen properly (necessary for most of the graphs). You might use this code: + theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.25)) (you can adjust the values in element_text if you like).

```
# Write you code to create the bar charts of drive result with small multiples for each drive beginning

ggplot(data=nfl,mapping=aes(x=drive_end,fill=drive_start))+
  geom_bar()+
  facet_wrap(vars(drive_start),nrow=2)+
  labs(x="Drive Results")+
  theme(axis.text.x = element_text(angle = 90,
                                   hjust = 1,
                                   vjust = 0.25))
```



Part 2c) Drive Result by Drive Start

Does it appear that teams are more likely to score (drive_end = Touchdown or Field Goal) depending on how the drive started? Explain your answer just based on the graphs from part b).

It appears that when they choose kickoff or punt as their drive start strategies, there are more Field Goal and Touchdown than another strategies of driving start. But we may need to perform association test comparing these two strategies(kickoff and punt) to get specific results for association and causation.