A10 - CSC 285

Suthi de Silva

1/22/2024

The roller coaster database

Scenario

Data is from rcdb.com – The roller coaster database. The data is from 2015 and has the following variables.

- **Name** : Name of the roller coaster
- **Park** : Name of the theme park
- **Track** : Type of material the roller coaster track is made of
- **Speed** : Top speed the ride reaches (mph)
- **Height** : Peak height the ride reaches (ft)
- **Drop** : Size of largest drop (ft)
- **Length** : Length of the roller coaster track (ft)
- **Duration** : Duration of ride (seconds)
- **Inversions** : Does the ride go upside down? Yes (1) or No (0)

Loading the data

```
rcd <- read.csv(file="/srv/R/CSC285_public/Suthi /Data/coasters-2015.csv",  header = TRUE)
data <- read.csv(file="/srv/R/CSC285_public/Shubha-Rakeb-Suthi-Kenna/BikeShare.txt",
                 sep="\t", quote="", comment.char="")

# Showing the first 10 rows
head(rcd, n = 10)
```

```
##                    Name                               Park Track Speed Height
## 1  Top Thrill Dragster                        Cedar Point Steel   120    420
## 2   Superman The Escap           Six Flags Magic Mountain Steel   100    415
## 3     Millennium Force                        Cedar Point Steel    93    310
## 4              Goliath           Six Flags Magic Mountain Steel    85    235
## 5                Titan               Six Flags Over Texas Steel    85    245
## 6    Phantom's Revenge                     Kennywood Park Steel    82    160
## 7           Xcelerator                 Knott's Berry Farm Steel    82    205
## 8            Desperado Buffalo Bill's Resort &amp; Casino Steel    80    209
## 9       HyperSonic XLC         Paramount's Kings Dominion Steel    80    165
## 10               Nitro          Six Flags Great Adventure Steel    80    230
##     Drop Length Duration Inversions
## 1    400   2800        .          0
## 2  328.1   1235        .          0
## 3    300   6595      165          0
## 4    255   4500      180          0
## 5    255   5312      210          0
## 6    228   3200        .          0
## 7     130   2202       62          0
## 8     225   5843      163          0
## 9     133   1560        .          0
## 10    215   5394      240          0
```

Removing erros

It seems like some of the columns have ".", which I assume is a mistake, and therefore I would replace it with zero, as removing those columns and rows would reduce the number of data significantly.

```
# Replacing the errors with zero.
rcd$Drop <-replace(rcd$Drop, rcd$Drop == ".", 0)
rcd$Duration <-replace(rcd$Duration, rcd$Duration == ".", 0)

# Checking the replacement.
head(rcd, n = 10)
```

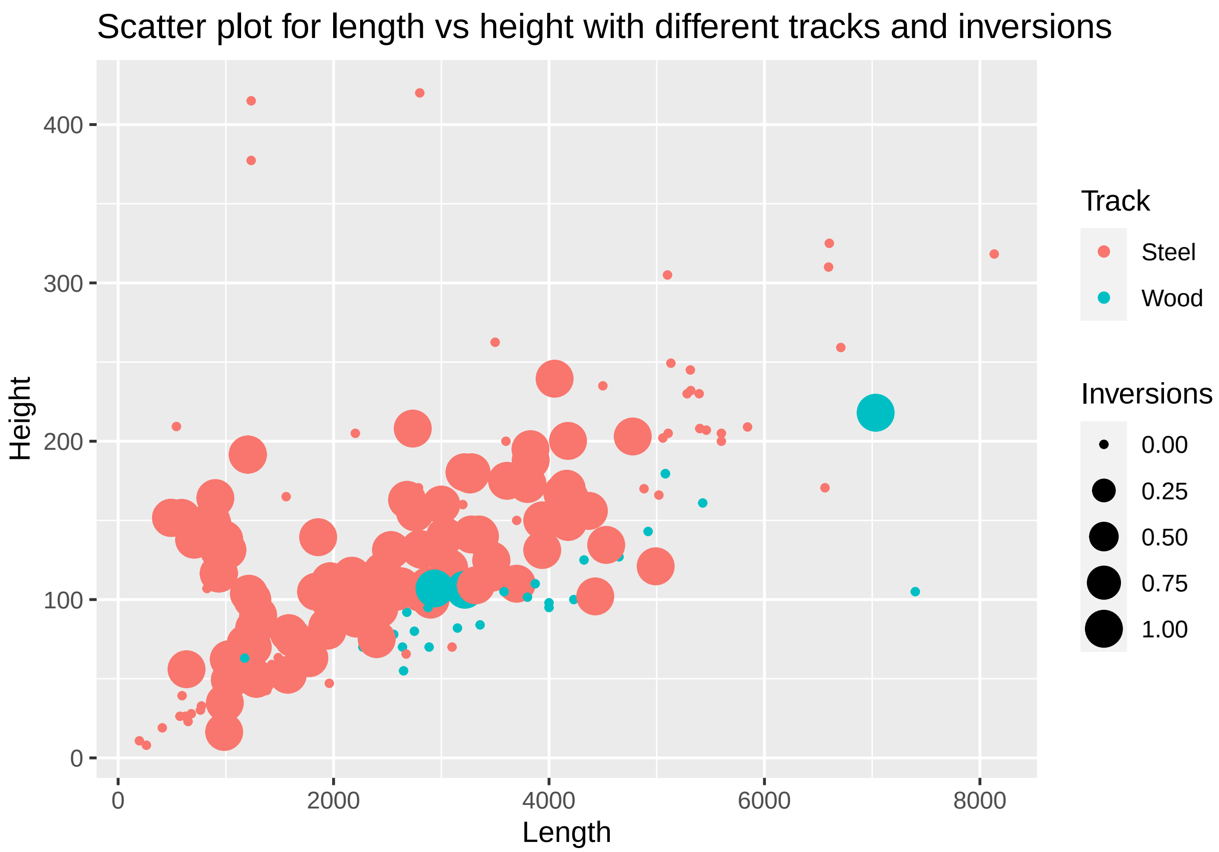
```
##                    Name                               Park Track Speed Height
## 1  Top Thrill Dragster                        Cedar Point Steel   120    420
## 2   Superman The Escap           Six Flags Magic Mountain Steel   100    415
## 3     Millennium Force                        Cedar Point Steel    93    310
## 4              Goliath           Six Flags Magic Mountain Steel    85    235
## 5                Titan               Six Flags Over Texas Steel    85    245
## 6    Phantom's Revenge                     Kennywood Park Steel    82    160
## 7           Xcelerator                 Knott's Berry Farm Steel    82    205
## 8            Desperado Buffalo Bill's Resort &amp; Casino Steel    80    209
## 9       HyperSonic XLC         Paramount's Kings Dominion Steel    80    165
## 10               Nitro          Six Flags Great Adventure Steel    80    230
##      Drop Length Duration Inversions
## 1     400   2800        0          0
## 2  328.1   1235        0          0
## 3     300   6595      165          0
## 4     255   4500      180          0
## 5     255   5312      210          0
## 6     228   3200        0          0
## 7     130   2202       62          0
## 8     225   5843      163          0
## 9     133   1560        0          0
## 10    215   5394      240          0
```

Questions

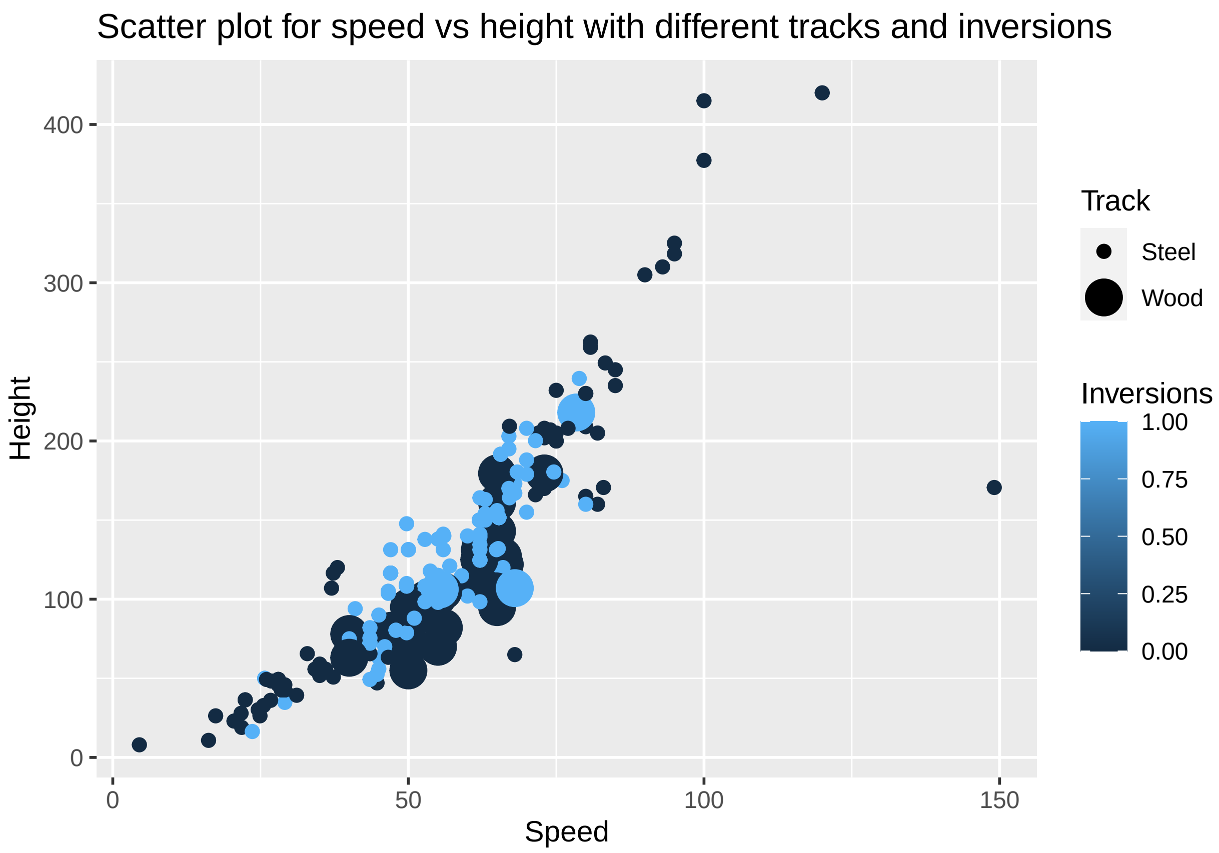
1. Choose any four variables (hint: choose at least one categorical variable)

a. Create two different scatter plots with all four variables (you may choose different variables, or visualize the same variables in two different ways)

```
#Creating the first scatter plot
ggplot(rcd,
       aes(x= Length,
           y= Height, # Assigning variables for dimensions or axes
           color = Track ,
           size = Inversions)) +
    geom_point() +   # Adding a title and custom labeling
  labs( title = "Scatter plot for length vs height with different tracks and inversions")
```



```
#Creating the second  scatter plot
ggplot(rcd,
       aes(x= Speed,
           y= Height, # Assigning variables for dimensions or axes
           color = Inversions ,
           size = Track)) +
    geom_point() +    # Adding a title and custom labeling
  labs( title = "Scatter plot for speed vs height with different tracks and inversions")
```



b. Describe your design choices, including:

i. How did you choose the variables that went on the x and y axes?

I realized that out of the four variables two of them could be quantitative variables (Speed, Height), which would be for the x, y axes of the plots.

ii. How did you include two other variables on the scatter plot?

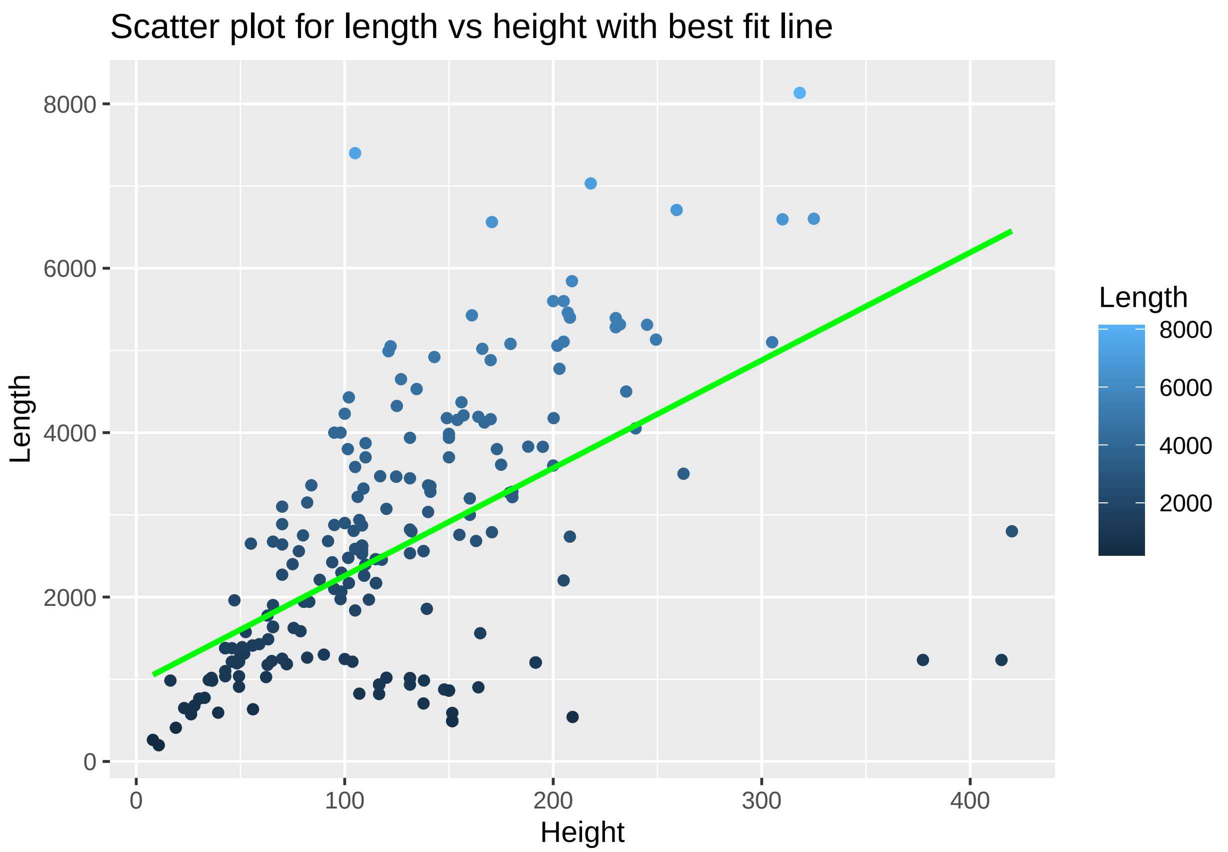
By selecting two categorical variables (Track, Inversions) for the other two dimensions (Size, Color) it gives me option to have two different point sizes, with two different colors.

2. Considering the full data set, are the variables correlated with one another?

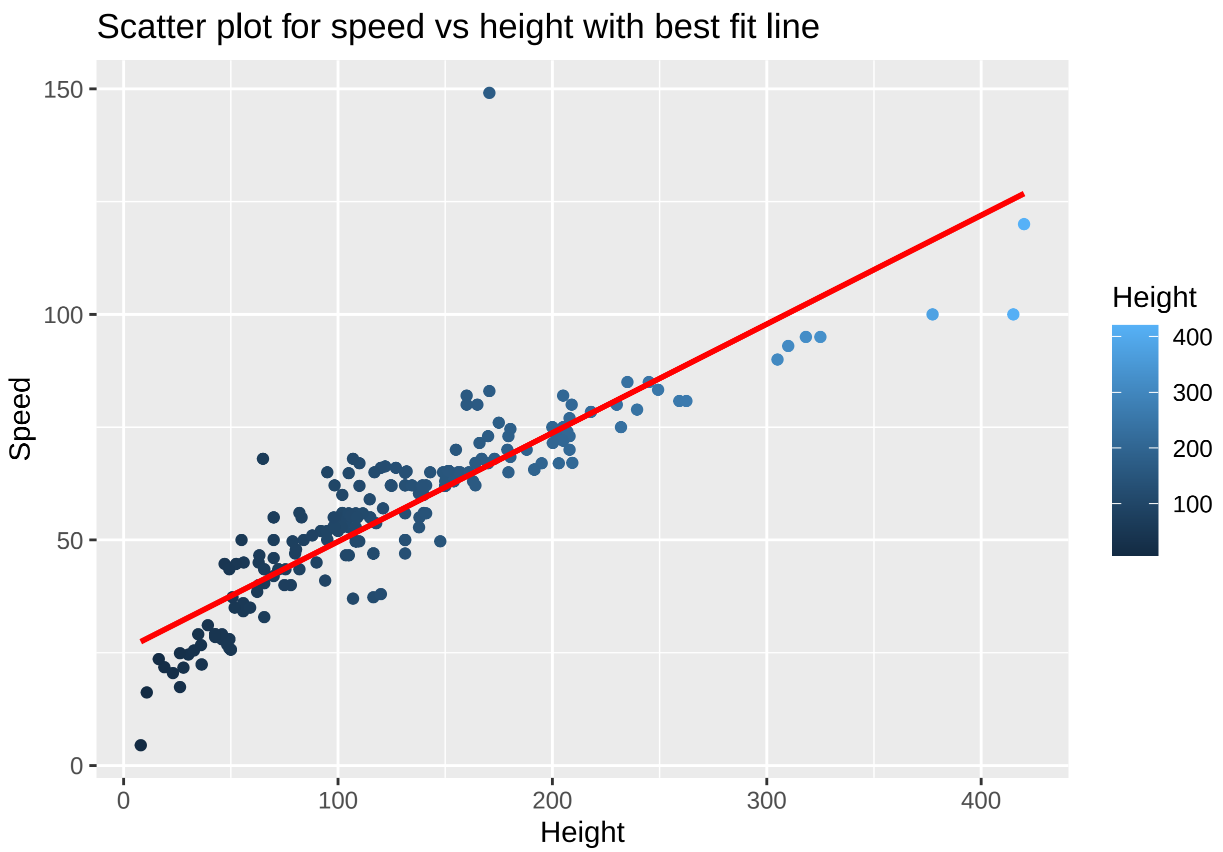
It seems like height and speed might, and also height and length might have a correlation with each other.

a. Visualize the correlation

```
#Creating the first scatter plot with best fit line
ggplot(rcd,
       aes(x= Height,
           y= Length,
           color = Length)) +  # Assigning variables for dimensions or axes
    geom_point() + geom_smooth(method=lm, se=FALSE,  col = "green") +  # Making a best fit line
  # Adding a title and custom labeling
  labs( title = "Scatter plot for length vs height with best fit line")
```



```
#Creating the second  scatter plot with best fit line
ggplot(rcd,
       aes(x= Height,
           y= Speed,
           color = Height)) +  # Assigning variables for dimensions or axes
    geom_point() + geom_smooth(method=lm, se=FALSE, col = "red") + # Making a best fit line
  # Adding a title and custom labeling
  labs( title = "Scatter plot for speed vs height with best fit line")
```



b. Describe your design choices

We know that, with two quantitative variables given a scatter plot could represent the exact values for each variable accurately using points. So using a scatter plot that has a best fit line would represent clear correlation if it exists.

c. What does your correlation visualization tell you?

```
# Getting the coefficients from the best fit lines for two plots
summary(lm(rcd$Length ~ rcd$Height))$coefficients
```

```
##                 Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 948.29749 181.496597  5.224877 3.794538e-07
## rcd$Height   13.10897   1.287272 10.183528 1.834257e-20
```

```
summary(lm(rcd$Speed ~ rcd$Height))$coefficients
```

```
##                Estimate  Std. Error  t value     Pr(>|t|)
## (Intercept) 25.5247928 1.214415887 21.01816 3.025077e-56
## rcd$Height   0.2410853 0.008613297 27.98990 2.171079e-77
```

According to the summary above, using the best fit line, correlation between Length and Height is, Length = 948.29749 + 13.10897(Height).

According to the summary above, using the best fit line, correlation between Speed and Height is, Speed = 25.5247928 + 0.2410853(Height).

3. Describe why a paired data visualizations would not work well here.

It is because, a paired data visualization is typically used to show the relationship between two paired variables. In the context of the roller coaster data frame here, a paired data visualization might not work well because the *variables seem to be independent of each other.For example, variables like "Name," "Park," and "Track Type" are categorical and don't have a direct numerical relationship with other variables. Additionally, variables such as "Speed," "Height," "Drop," "Length," "Duration," and "Inversions" are numerical, but there may not be a clear pairing or correlation between them. Paired data visualizations are more suitable when we have paired measurements or observations for the same set of entities. In this case, a scatter plot or a line plot might be more appropriate for exploring relationships between individual numerical variables.*

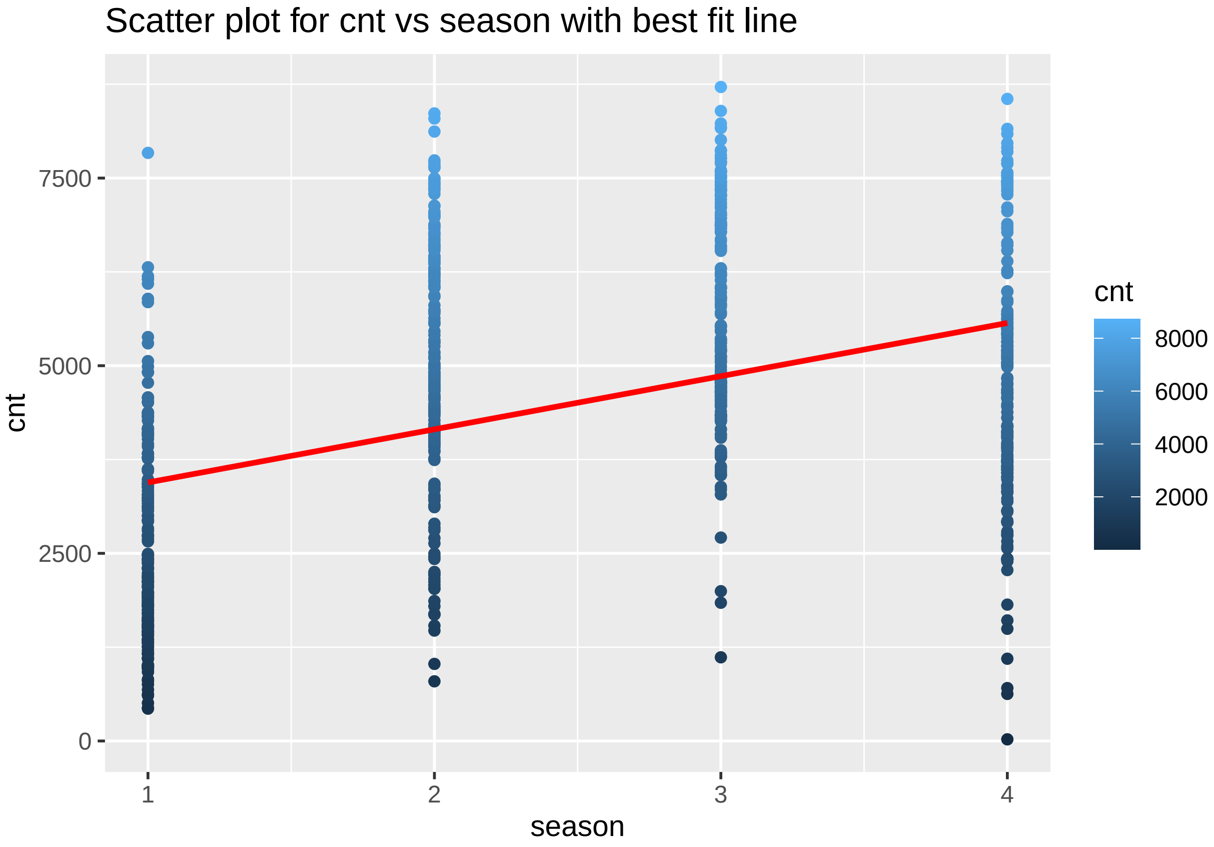
4. Refer (back) to the bikeshare.txt data from visualizing amounts and distributions. We DO have paired data in the bike share data.

a. Describe the paired data you will be using

Measuring count of total users in each season would be paired data, "cnt" measures total users through out the time in between seasons, which would show the change of users between seasons.

b. Create a scatter plot with the paired data.

```
#Creating the first scatter plot with best fit line
ggplot(data,
       aes(x= season,
           y= cnt,
           color = cnt)) +  # Assigning variables for dimensions or axes
    geom_point() + geom_smooth(method=lm, se=FALSE,  col = "red") + # Making a best fit line
  labs( title = "Scatter plot for cnt vs season with best fit line")
```



i. Be sure to include the line y=x

Added

ii. Describe your design choices for the paired scatter plot.

We know that, with a quantitative variable and a categorical given, a scatter plot could represent the exact quantitative values for each category or period accurately using points. So using a scatter plot that has a best fit line would represent clear correlation if it exists.

iii. What does your paired scatter plot tell you?

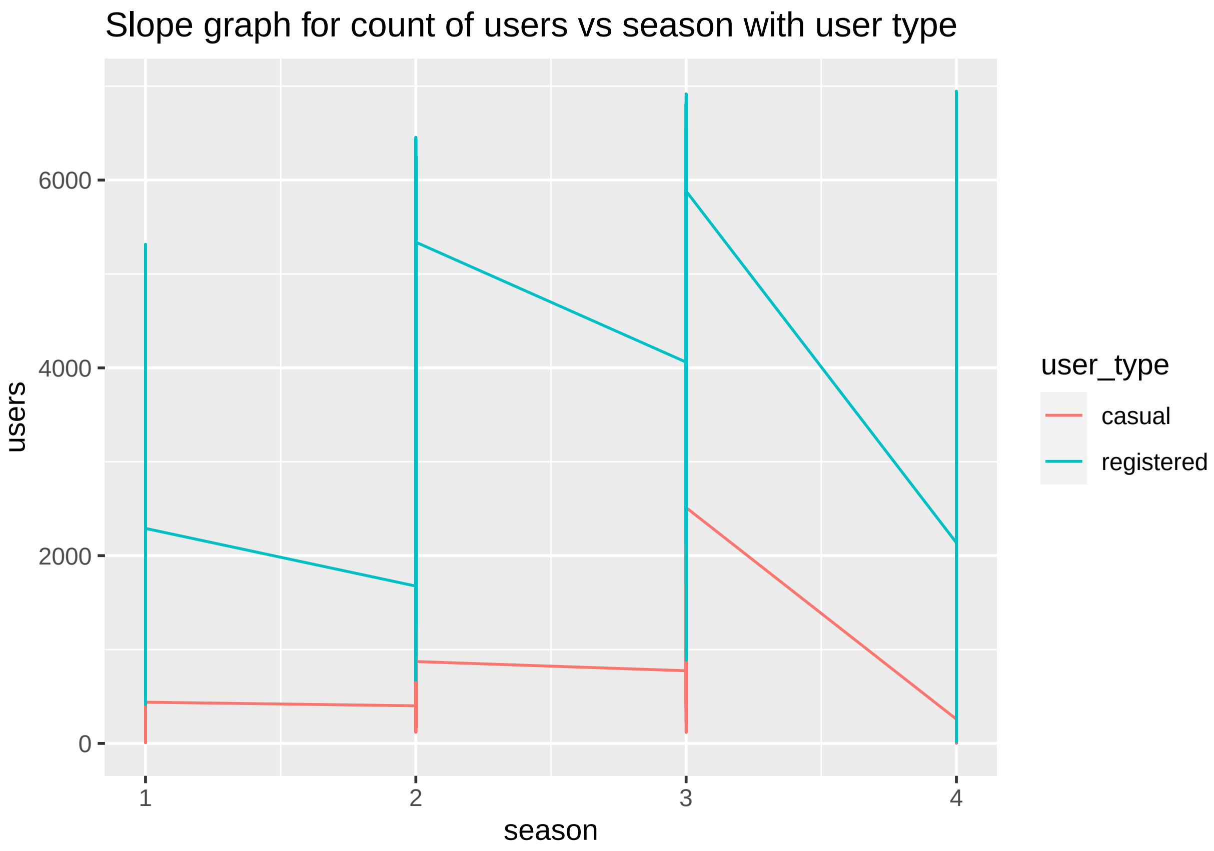
```
# Getting the coefficients from the best fit lines for two plots
summary(lm(data$cnt ~ data$season))$coefficients
```

```
##              Estimate Std. Error  t value     Pr(>|t|)
## (Intercept) 2736.2063  161.27149 16.96646 1.149062e-54
## data$season  708.2258   59.02545 11.99865 2.133997e-30
```

According to the summary above, using the best fit line, correlation between season and count of users is, cnt = 2736.2063 + 708.2258(season).

c. Create a slope graph with the paired data

```
# I would create variables to longer format
data2 <- data %>% pivot_longer(cols=c('casual', 'registered'),
                               names_to='user_type',
                               values_to='users')
# making slope graph
data2 %>% ggplot(aes(x = season ,
                     y = users,
                     group = user_type,
                     color = user_type)) +
  geom_line() +
  labs( title = "Slope graph for count of users vs season with user type")
```



i. Feel free to reduce the data (such as limit to one month, weekends, etc.)

Done

ii. Describe your design choices

A slope graph is a well-suited choice for interpreting the count of users, specifically casual and registered users, in the BikeShare dataset across different seasons. This design choice facilitates a comparative analysis, allowing easy identification of trends and variations in user counts throughout the year. By employing lines to connect data points for casual and registered users separately, the slope graph enables a clear representation of how each user type responds to seasonal changes. This approach focuses on highlighting differences and revealing any distinct patterns associated with different seasons, providing a concise and visually effective means to understand the variations in bike rentals over time. Using dual axes for casual and registered user counts ensures that both aspects can be comprehensively observed on the same graph without overlap, enhancing the overall interpretability of the data. Consideration of color differentiation and clear labeling further enhances the clarity of the slope graph for effective communication of seasonal trends in bike-sharing user counts.

d. Do you prefer the slope graph or the paired scatter plot? Why?

Choosing a slope graph over a paired scatter plot for interpreting the count of users across different seasons in the BikeShare dataset offers several advantages. Firstly, a slope graph is particularly effective for revealing trends and changes over categories, such as seasons, providing a clear visual representation of how counts vary. Unlike a paired scatter plot, a slope graph directly connects data points with lines, facilitating a more straightforward comparison between casual and registered users. This direct line connection emphasizes the sequential relationship between the seasons and enables a focused analysis of the trends. Additionally, a slope graph is well-suited for highlighting differences between user types, making it easier to discern patterns and variations in bike rentals throughout the year. The simplicity and efficiency of the slope graph make it a preferred choice when the emphasis is on comparative trends and seasonal patterns in the count of users.