Assignment 03

Jesse Conlon

2/9/2020

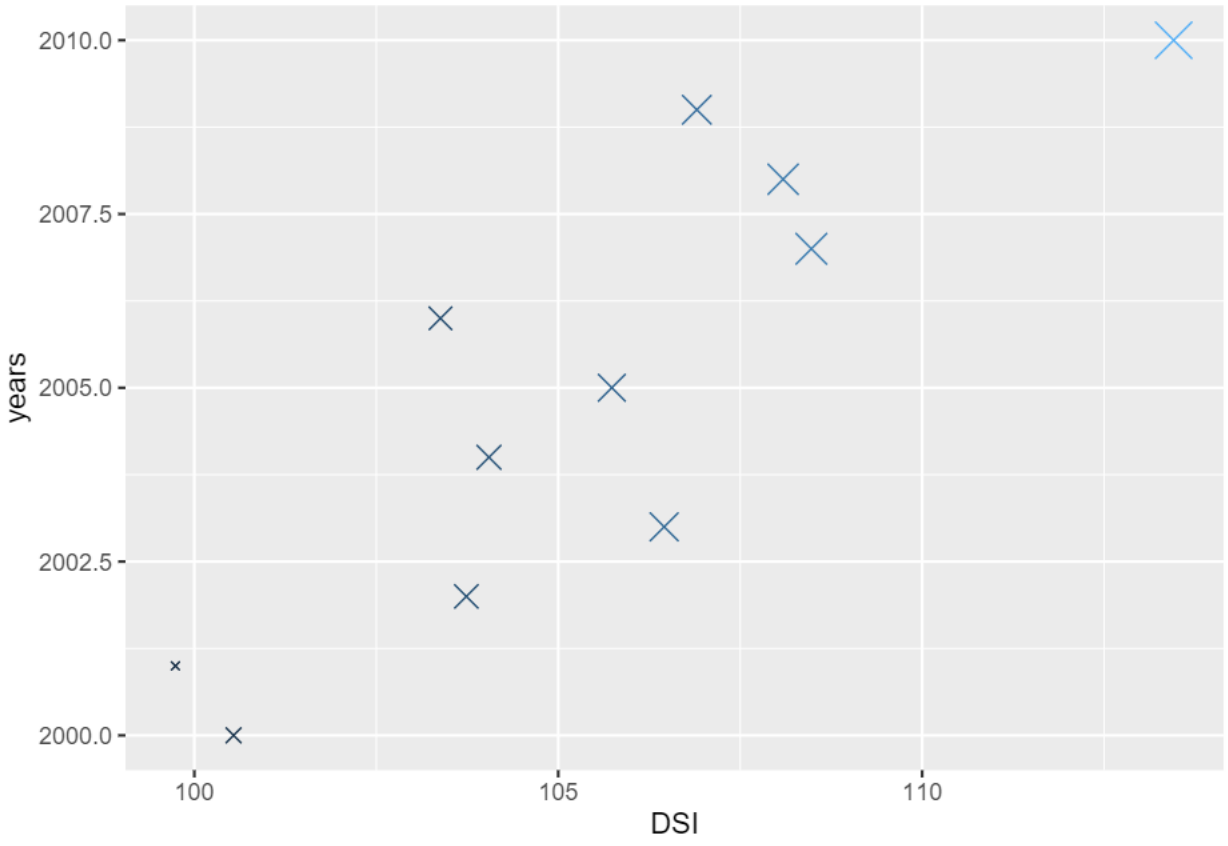
```
library(tidyverse)
library(scales)

datset1 <- read_csv("data/datset1.csv")
datset2 <- read_csv("data/datset2.csv")
datset3 <- read_csv("data/datset3.csv")
datset4 <- read_csv("data/datset4.csv")
```

Question 1

Below is a replication of the "plot1.pdf" graphic:

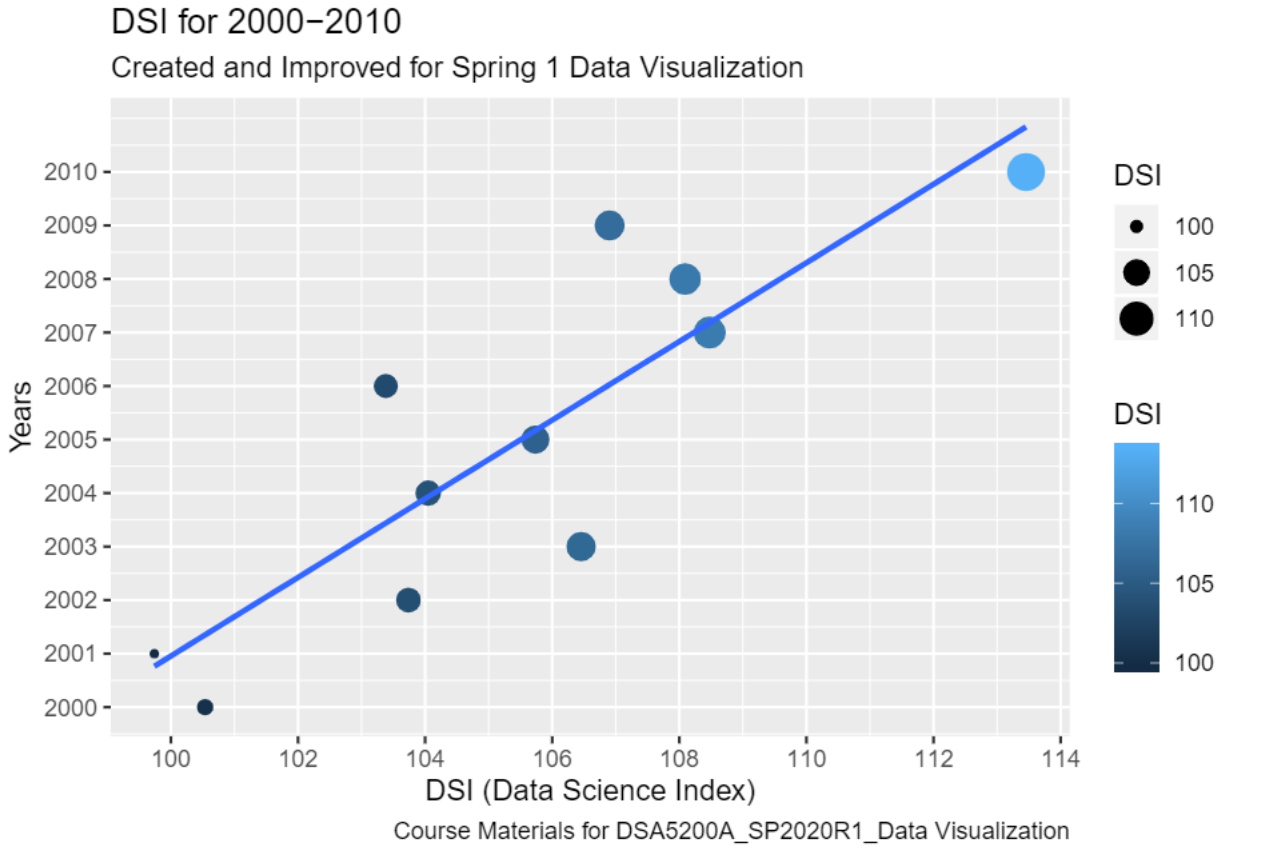
```
ggplot(datset1, aes(x = DSI, y = years)) + geom_point(aes(size = DSI, color = DSI), shape = 4) +
    theme(legend.position = "none")
```



Question 2

Below is how I improved the graph. I started by cutting out "shape = 4" as I found that to be inferior to the default circles. I then labeled various elements of the graph so that viewers had a better idea of what they were looking at. Next, I established intervals for the graph that make the points a bit more distinguishable while still being easy to read and not cluttered. Last, I added a trend line to clearly indicate that there is a positive coorelation occuring between the variables:

```
ggplot(datset1, aes(x = DSI, y = years)) + geom_point(aes(size = DSI, color = DSI)) +
    labs(x = "DSI (Data Science Index)", y = "Years",
    title = "DSI for 2000-2010",
    subtitle = "Created and Improved for Spring 1 Data Visualization",
    caption = "Course Materials for DSA5200A_SP2020R1_Data Visualization") +
    scale_y_continuous(breaks = seq(2000,2010, by = 1)) +
    scale_x_continuous(breaks = seq(96,115, by = 2)) +
    geom_smooth(method="lm", se=FALSE)
```



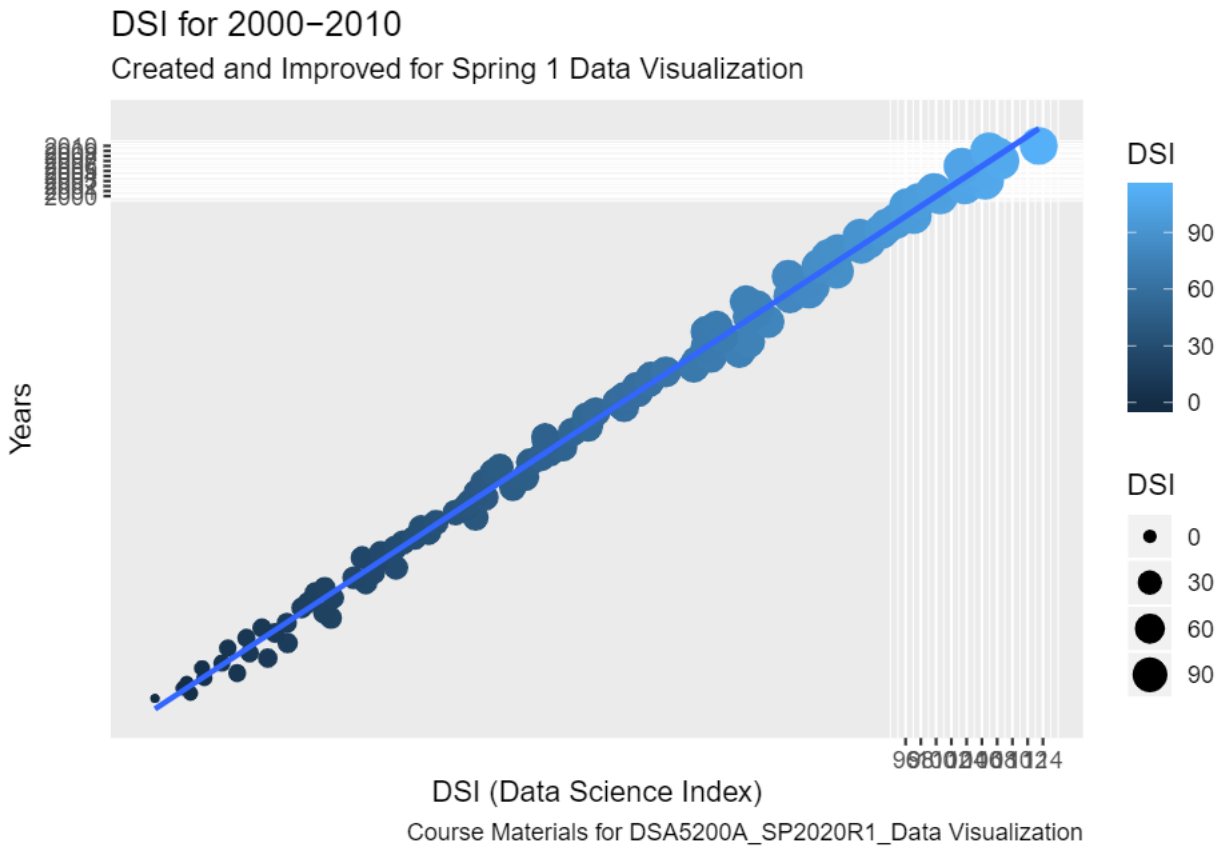
Question 3

After changing to the expanded data set, a lot of issues occurred. The intervals were a mess due to the graph being customized for 2000-2010 time frame. Additionally all the points were blurred since they were kind of squished together. Last the labels were inaccurate since the data now include observations from as far back as 1900:

```
datset2 <- read_csv("data/datset2.csv")
```

```
## Parsed with column specification:
## cols(
##   years = col_double(),
##   DSI = col_double()
## )
```

```
ggplot(datset2, aes(x = DSI, y = years)) + geom_point(aes(size = DSI, color = DSI)) +
    labs(x = "DSI (Data Science Index)", y = "Years",
    title = "DSI for 2000-2010",
    subtitle = "Created and Improved for Spring 1 Data Visualization",
    caption = "Course Materials for DSA5200A_SP2020R1_Data Visualization") +
    scale_y_continuous(breaks = seq(2000,2010, by = 1)) +
    scale_x_continuous(breaks = seq(96,115, by = 2)) +
    geom_smooth(method="lm", se=FALSE)
```



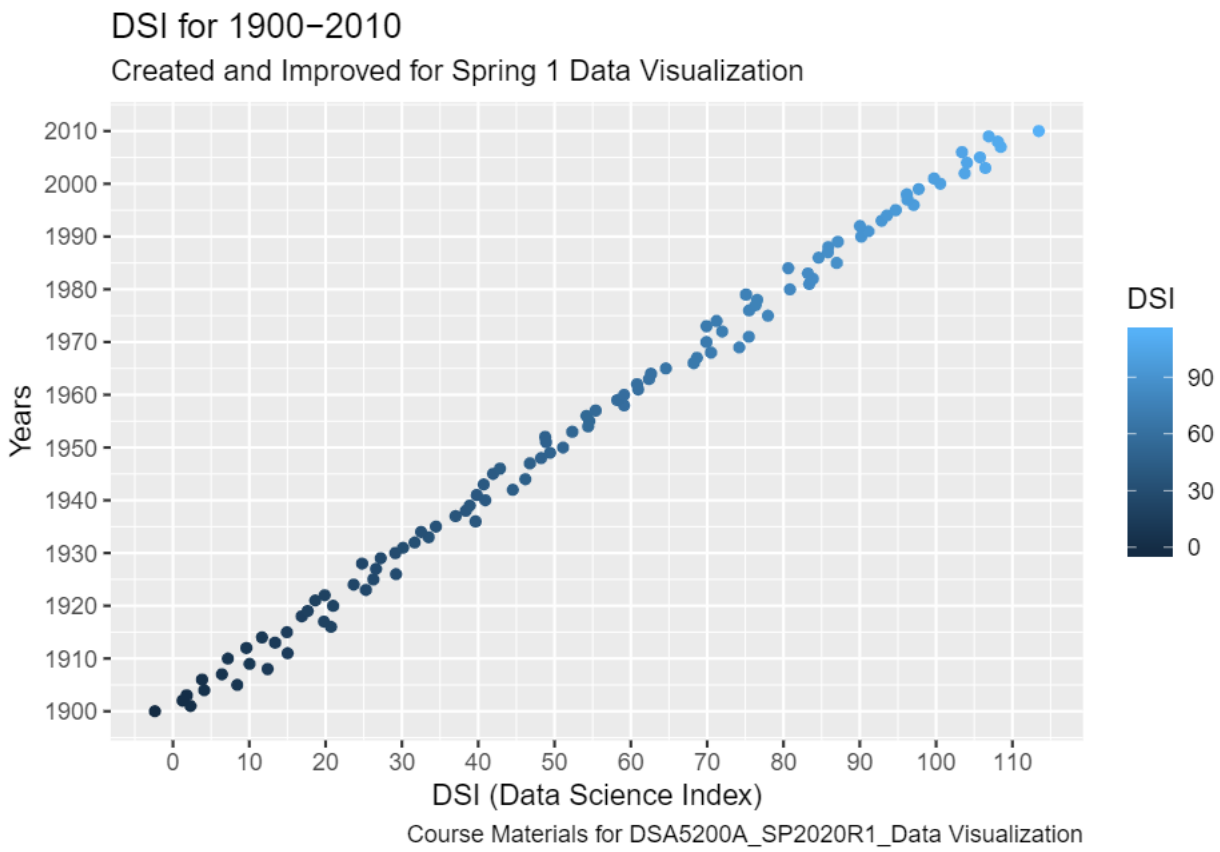
Question 4

The elements that were an issue in the above graph were remedied, mostly in a case of addition by subtraction. I think the point of this graph is to show how the DSI has progressively increased over the years. It's simple and I think should remain rather simple. I thought the trend line as well as modifying the sizes of the circles was doing more harm than good at this point and creating more of a distraction than anything else. I adjusted the intervals to every 10 years and did the same for the DSI. I left the color change because it does catch the eye and makes it less boring to look at in my opinion!:

```
datset2 <- read_csv("data/datset2.csv")
```

```
## Parsed with column specification:
## cols(
##   years = col_double(),
##   DSI = col_double()
## )
```

```
ggplot(datset2, aes(x = DSI, y = years)) + geom_point(aes(color = DSI)) +
    labs(x = "DSI (Data Science Index)", y = "Years",
    title = "DSI for 1900-2010",
    subtitle = "Created and Improved for Spring 1 Data Visualization",
    caption = "Course Materials for DSA5200A_SP2020R1_Data Visualization") +
    scale_y_continuous(breaks = seq(1900,2010, by = 10)) +
    scale_x_continuous(breaks = seq(-10,115, by = 10))
```



Question 5

For this one, I found that reshaping the data was the best way to plot the data in a meaningful way. The goal was to show and compare the GPD between the countries. This is possible when "Country" is a variable as opposed to seprate columns for the countries being compared:

```
datset3 <- read_csv("data/datset3.csv")
```

```
## Parsed with column specification:
## cols(
##   year = col_double(),
##   Argentina = col_double(),
```

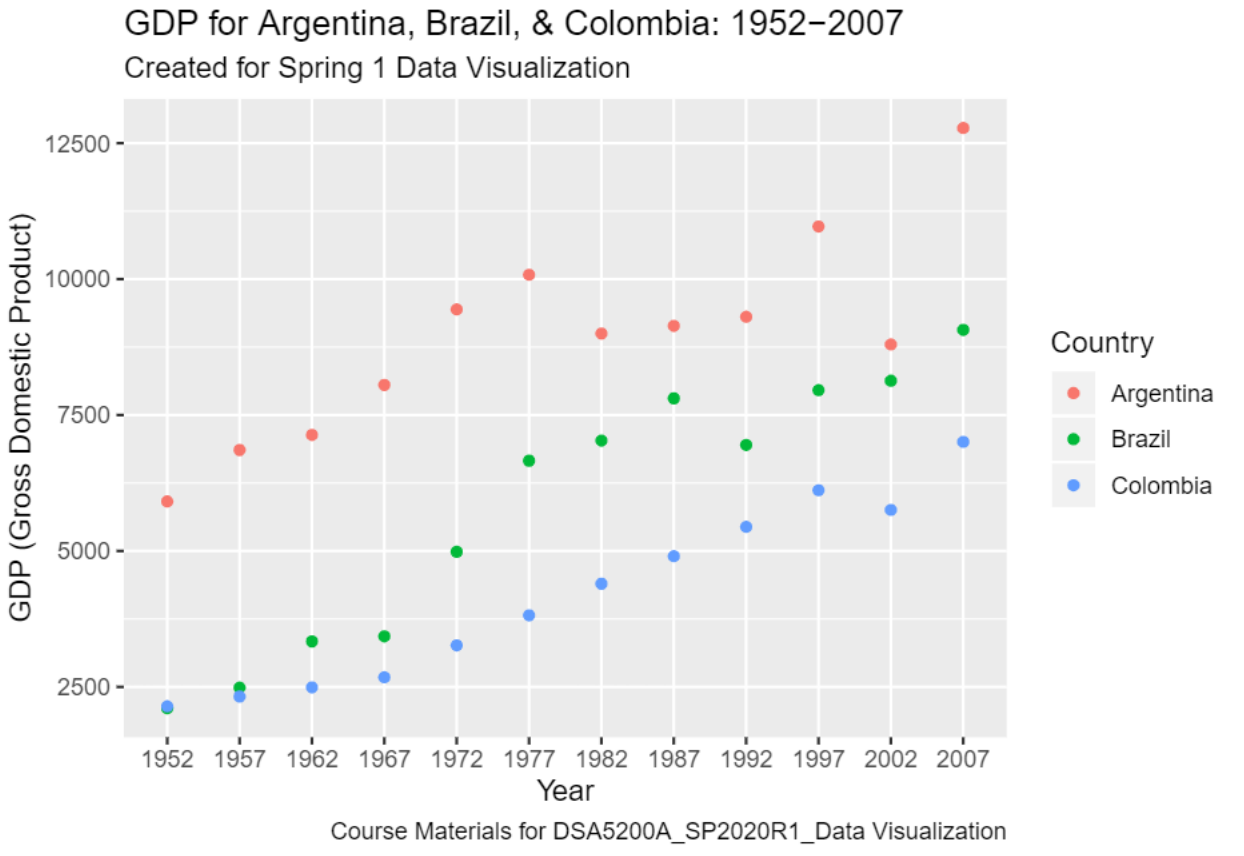
```
##   Brazil = col_double(),
##   Colombia = col_double()
## )
```

```
GDP <- gather(data = datset3, key = Country, value = GDP, -year)
```

There may be a better way to do this but I looked up this method on YouTube and it seemed to work just as I hoped. Now, with the data reshaped, I can make much more effective plots.

Following this I worked with a few different graphics. I prefer the faceted option (posted second), but the first option is also effective in showing the differences between the countries as well as trends:

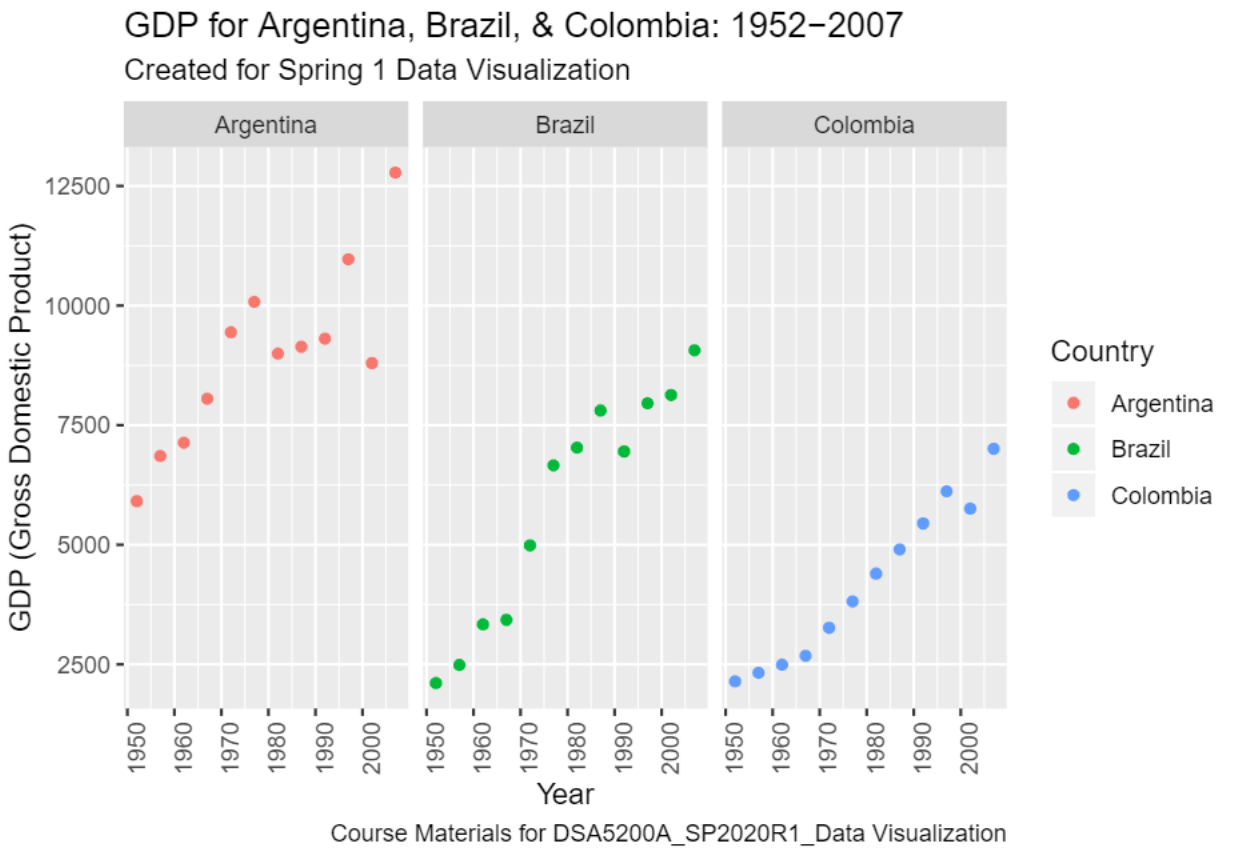
```
ggplot(GDP, aes(x = factor(year), y = GDP, col = Country)) +
    labs(x = "Year", y = "GDP (Gross Domestic Product)",
    title = "GDP for Argentina, Brazil, & Colombia: 1952-2007",
    subtitle = "Created for Spring 1 Data Visualization",
    caption = "Course Materials for DSA5200A_SP2020R1_Data Visualization") +
    geom_point()
```



Facet example, small adjustments to change the direction of the text on the x-axis so that it's a bit more legible:

```
ggplot(GDP, aes(x = year, y = GDP, col = Country))  +
    labs(x = "Year", y = "GDP (Gross Domestic Product)",
    title = "GDP for Argentina, Brazil, & Colombia: 1952-2007",
    subtitle = "Created for Spring 1 Data Visualization",
    caption = "Course Materials for DSA5200A_SP2020R1_Data Visualization") +
    geom_point() +
    facet_wrap(~Country) +
```

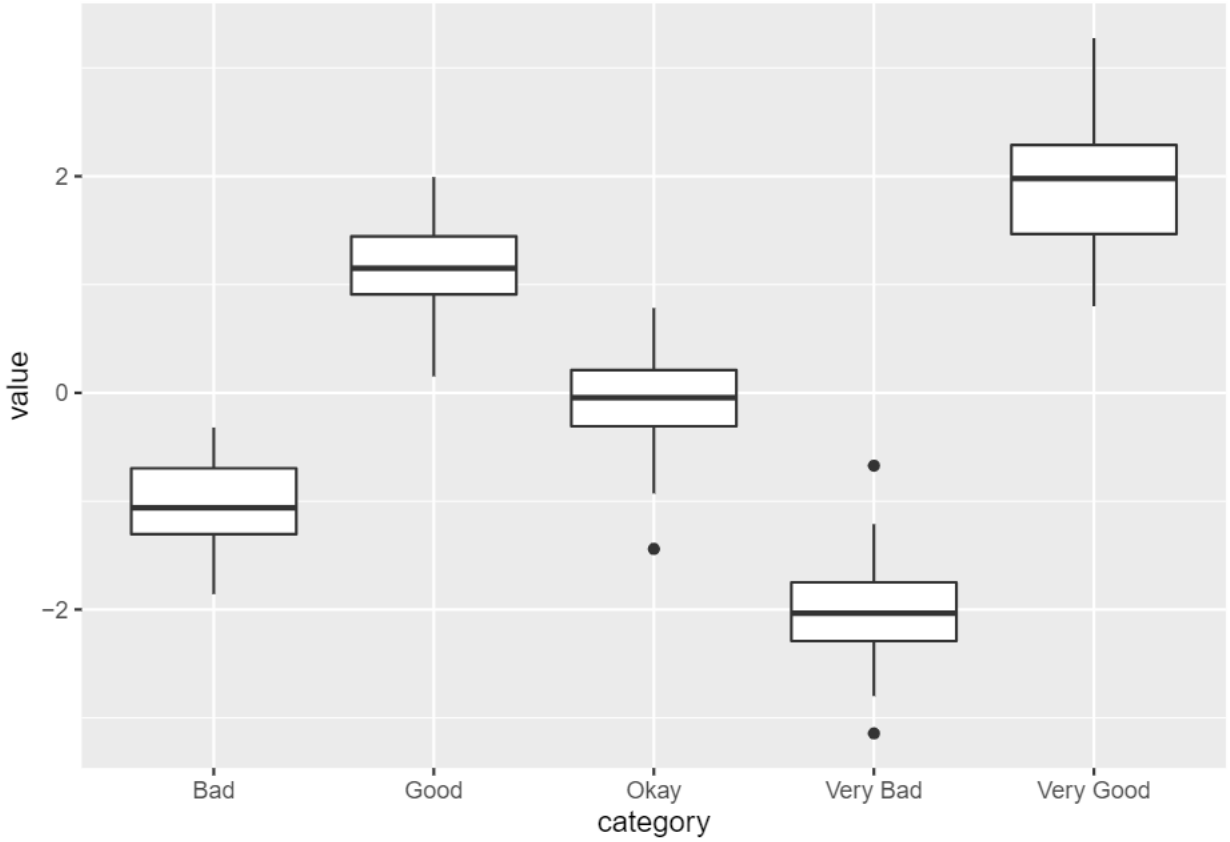
```
    theme(axis.text.x = element_text(angle=90))
```



Question 6

I tried a couple different types of plots but ultimately settled for a box plot.

```
ggplot(datset4, aes(x = category, y = value)) + geom_boxplot()
```

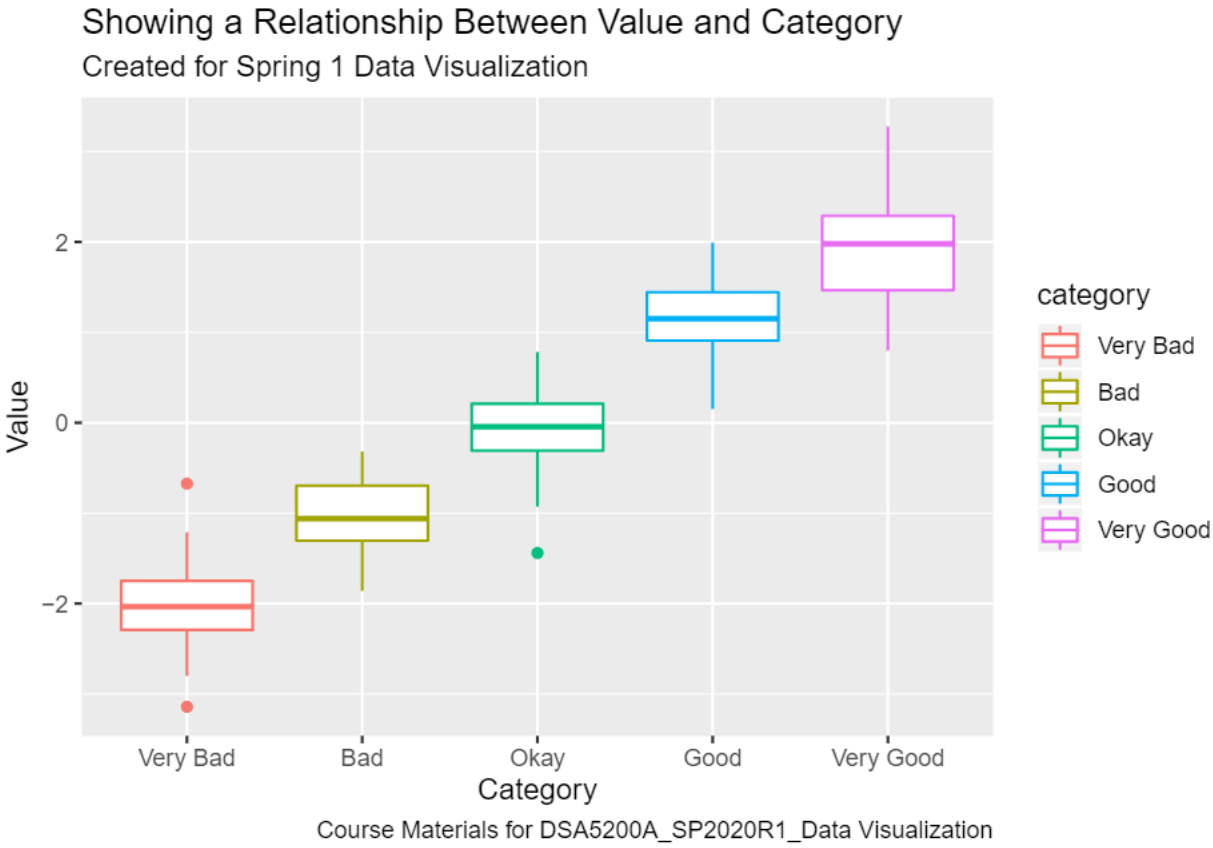


I felt that the box plot showed an excellent relationship between the variables. The reason for this is because it gives an outline of the distribution of points in that category. While the nature of the boxplot is to show distribution, it indicates where most of the distribution is occuring depending on the category. What I see when I look at this is that something that's rated "Very Good" has consistently higher value, in the 2.0 range, and from here it just steps down categorically until "Very Bad", consistently scoring a -2.0. Therefore there is a relationship between value and category in that the higher the value it is the more likely the category will be higher as well.

I made some small adjustments to show this relationship more clearly and how the values increase as the rating progresses from "Very Bad" to "Very Good" by mutating the category levels. I also added some color to bring it to life a little bit:

```
Category <- datset4 %>%
  mutate(category = fct_relevel(category, "Very Bad", "Bad", "Okay", "Good", "Very Good"))

ggplot(Category, aes(x = category, y = value, col = category)) + geom_boxplot() +
    labs(x = "Category", y = "Value",
    title = "Showing a Relationship Between Value and Category",
    subtitle = "Created for Spring 1 Data Visualization",
    caption = "Course Materials for DSA5200A_SP2020R1_Data Visualization")
```



The pros of this plot are that it shows a clear relationship between the variables and also where the median is. Additionally, any outliers are on display.

The cons with the boxplot is that there is no way to tell how many instances there are of each category. This could be something useful depending on the situation and number of observations.