July 2, 2023

The results below are generated from an R script.

```
# Assignment: Week4 #1
# Name: Couto, Maria
# Date: 2023-06-30

setwd("C:/Users/ait0s/OneDrive/Documents/GitHub/Couto_DSC520")

# Load the 'scores.csv' data set

datasetforscores <- read.csv("scores.csv")
datasetforscores

##    Count Score Section
## 1     10   200  Sports
## 2     10   205  Sports
## 3     20   235  Sports
## 4     10   240  Sports
## 5     10   250  Sports
## 6     10   265 Regular
## 7     10   275 Regular
## 8     30   285  Sports
## 9     10   295 Regular
## 10    10   300 Regular
## 11    20   300  Sports
## 12    10   305  Sports
## 13    10   305 Regular
## 14    10   310 Regular
## 15    10   310  Sports
## 16    20   320 Regular
## 17    10   305 Regular
## 18    10   315  Sports
## 19    20   320 Regular
## 20    10   325 Regular
## 21    10   325  Sports
## 22    20   330 Regular
## 23    10   330  Sports
## 24    30   335  Sports
## 25    10   335 Regular
## 26    20   340 Regular
## 27    10   340  Sports
## 28    30   350 Regular
## 29    20   360 Regular
## 30    10   360  Sports
## 31    20   365 Regular
```

```
## 32    20   365  Sports
## 33    10   370  Sports
## 34    10   370 Regular
## 35    20   375 Regular
## 36    10   375  Sports
## 37    20   380 Regular
## 38    10   395  Sports
```

```
# What are the observational units in this study?

  # In statistics, observational units are the entities (people,things,etc.)
  # and may sometimes be referred as subject if they are people.
  # In a datafarame,
  # however, every row is considered an observation and every column is a
  # variable. There are 38 rows and 3 columns in this dataset
```

```
nrow(datasetforscores)
```

```
## [1] 38
```

```
# Identify the variables mentioned in the narrative paragraph
#   and determine which are categorical and quantitative?

  # Categorical Variables: sections (sports and variety)
  # Quantitative: course grades and total points earned

# Create one variable to hold a subset of your data set that contains
#   only the Regular Section and one variable for the Sports Section.
```

```
library(plyr)
library(dplyr)
select(datasetforscores, "Section")
```

```
##    Section
## 1   Sports
## 2   Sports
## 3   Sports
## 4   Sports
## 5   Sports
## 6  Regular
## 7  Regular
## 8   Sports
## 9  Regular
## 10 Regular
## 11  Sports
## 12  Sports
## 13 Regular
## 14 Regular
## 15  Sports
## 16 Regular
## 17 Regular
## 18  Sports
## 19 Regular
## 20 Regular
## 21  Sports
```

```
## 22 Regular
## 23  Sports
## 24  Sports
## 25 Regular
## 26 Regular
## 27  Sports
## 28 Regular
## 29 Regular
## 30  Sports
## 31 Regular
## 32  Sports
## 33  Sports
## 34 Regular
## 35 Regular
## 36  Sports
## 37 Regular
## 38  Sports
```

```
regular_section <- filter(datasetforscores, Section == "Regular")
sports_section <- filter(datasetforscores, Section == "Sports")
regular_section
```

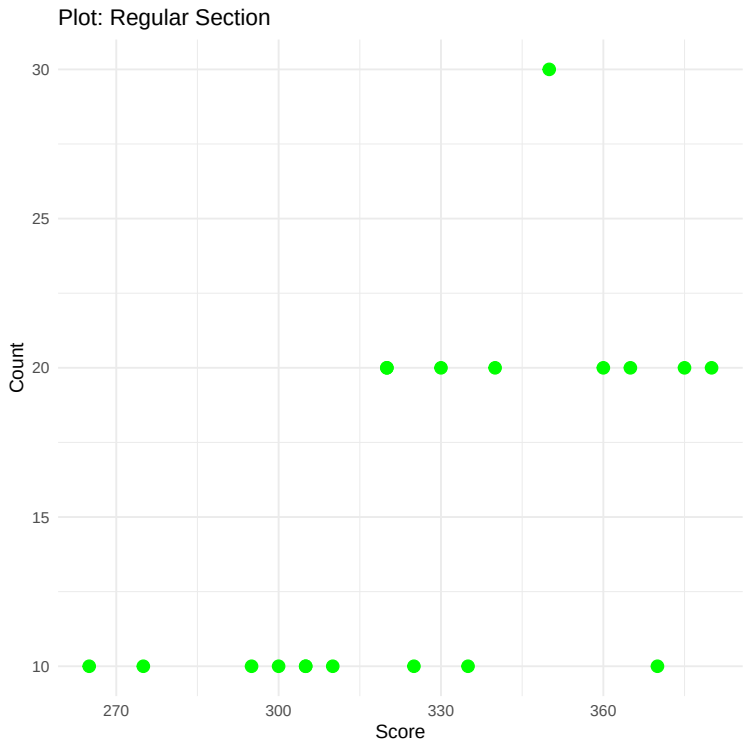
```
##    Count Score Section
## 1     10   265 Regular
## 2     10   275 Regular
## 3     10   295 Regular
## 4     10   300 Regular
## 5     10   305 Regular
## 6     10   310 Regular
## 7     20   320 Regular
## 8     10   305 Regular
## 9     20   320 Regular
## 10    10   325 Regular
## 11    20   330 Regular
## 12    10   335 Regular
## 13    20   340 Regular
## 14    30   350 Regular
## 15    20   360 Regular
## 16    20   365 Regular
## 17    10   370 Regular
## 18    20   375 Regular
## 19    20   380 Regular
```

```
sports_section
```

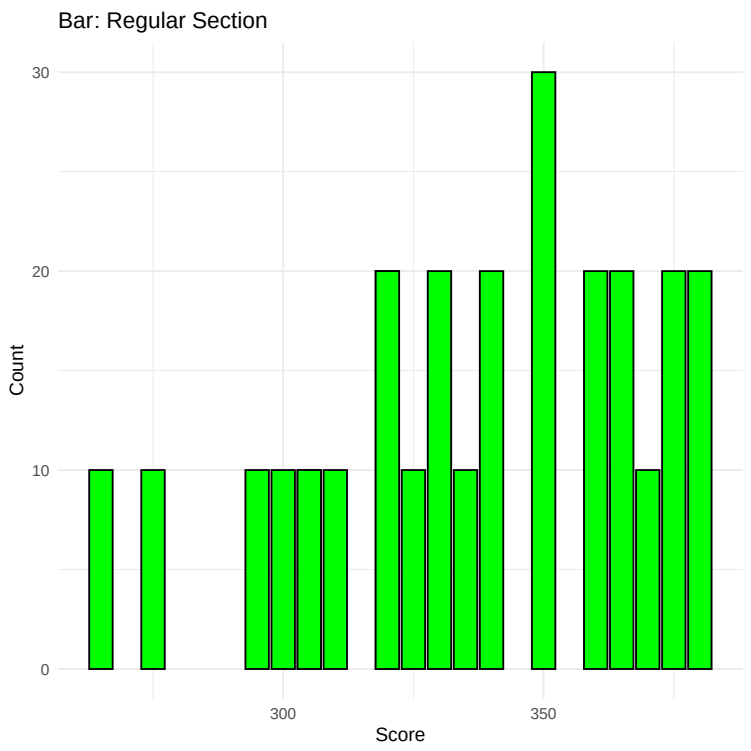
```
##    Count Score Section
## 1     10   200  Sports
## 2     10   205  Sports
## 3     20   235  Sports
## 4     10   240  Sports
## 5     10   250  Sports
## 6     30   285  Sports
## 7     20   300  Sports
## 8     10   305  Sports
## 9     10   310  Sports
```

```
## 10    10   315  Sports
## 11    10   325  Sports
## 12    10   330  Sports
## 13    30   335  Sports
## 14    10   340  Sports
## 15    10   360  Sports
## 16    20   365  Sports
## 17    10   370  Sports
## 18    10   375  Sports
## 19    10   395  Sports
```

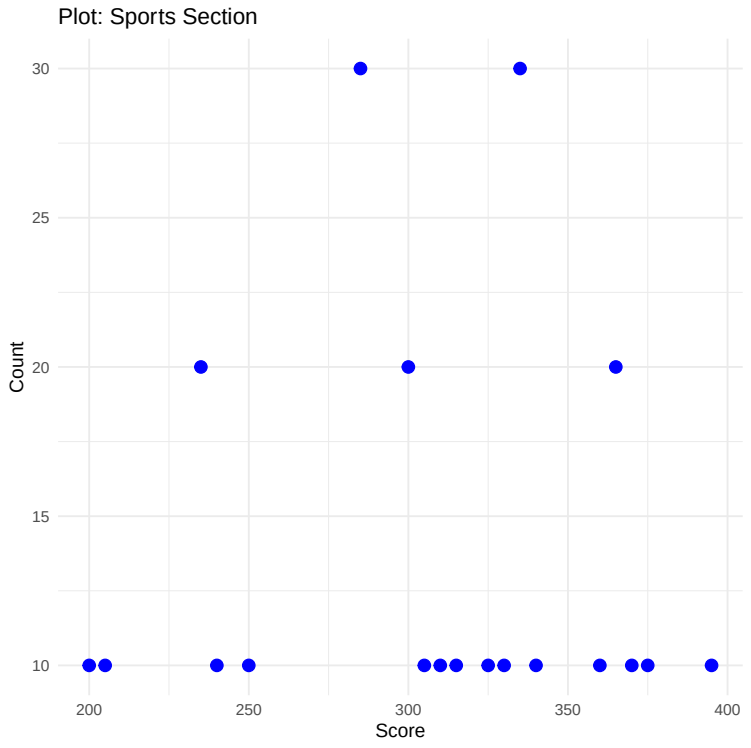
```
# Use the Plot function to plot each Sections scores and
# the number of students achieving that score.
# Use additional Plot Arguments to label the graph
# and give each axis an appropriate label.


library(ggplot2)
theme_set(theme_minimal())

# Plot each Section (Regular Scatter & Histogram)
ggplot(regular_section, aes(Score, Count)) +
  geom_point(colour = "green",size = 3) +
  ggtitle("Plot: Regular Section")
```



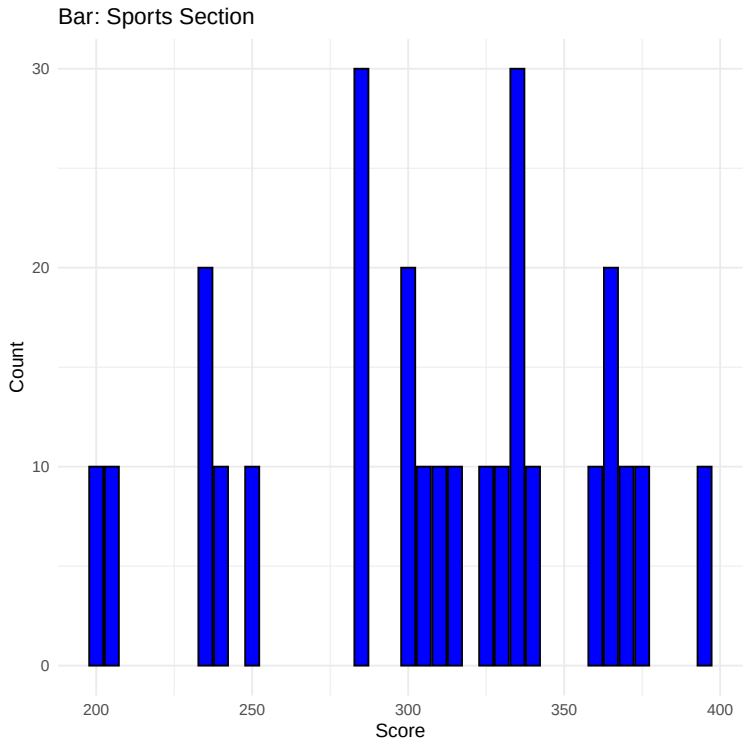

```
ggplot(regular_section, aes(y=Count,x=Score)) +
  geom_bar(position = 'dodge', stat='identity',colour="black",fill="green") +
  ggtitle("Bar: Regular Section")
```




```
# Plot each Section (Sports Scatter & Histogram)

ggplot(sports_section, aes(Score, Count)) +
  geom_point(colour = "blue",size = 3) +
  ggtitle("Plot: Sports Section")
```

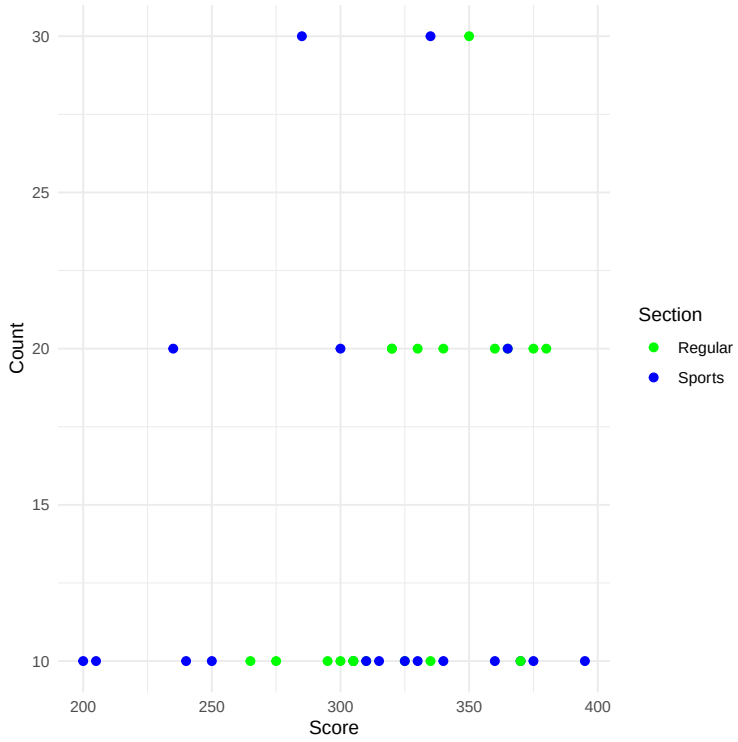


```r
ggplot(sports_section, aes(y=Count,x=Score)) +
  geom_bar(position = 'dodge', stat='identity',colour="black",fill="blue") +
  ggtitle("Bar: Sports Section")
```

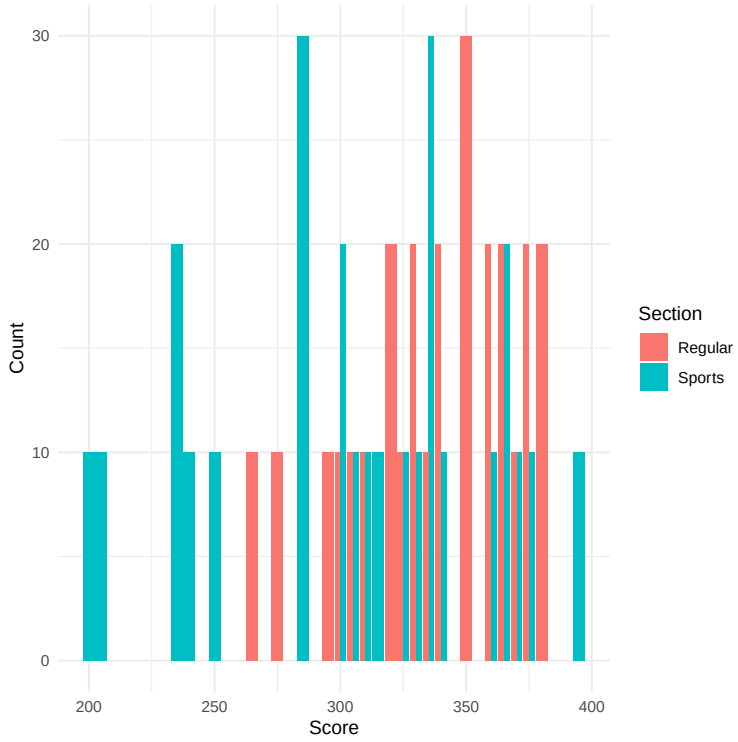



```r
# Comparing and contrasting the point distributions between the two section,
# looking at both tendency and consistency: Can you say that one section tended
# to score more points than the other? Justify and explain your answer.
  # we can see that the regular section has a tendency to score more.
  # the bar chart for the regular section is unimodal with one peak and
  # has a distribution that is left -skewed. Meaning, more participants
  # scored higher and the lower tail is longer on the left side.
  # The sports section,on the other hand, has a bimodal distribution with two
  # distinct peaks and has multiple modes where different values
  # appear more in the dataset.


# Did every student in one section score more points than every student
# in the other section? If not, explain what a statistical tendency means
# in this context.
# For this question, I plotted the values of both sections side by side
# to show the comparison between the two. The visuals show us
# that the sports section has both the highest and the lowest score
# in the dataset. So not every student in one section score more
# point than the other. Rather, the scores are more distributed
# between the two sections. Statistical tendency helps us
# describe a dataset by showing the frequency of the distribution of the
# observations. The charts help us see the mode or the frequency of the
# occurence in each data points. The graph shows that in the regular section,
# the data points gravitate toward the higher end of the x axis which is a good
```

```
# indicator that the regular section, overall, scored higher points than the
# sports section.

#Side-by_Side Plot comparison for Regular and Sports Section

ggplot(datasetforscores, aes(x=Score,y=Count,colour=Section)) +
  geom_point(size = 2) +
  scale_color_manual(values = c("Regular"="green","Sports"="blue"))
```




```
#Side-by_Side Bar comparison for Regular and Sports Section


ggplot(datasetforscores, aes(fill=Section,y=Count,x=Score)) +
  geom_bar(position = 'dodge', stat='identity')
```




```
# What could be one additional variable that was not mentioned
# in the narrative that could be influencing the point distributions
# between the two sections
# On the narrative- it speaks to  course grades and total points
# earned in the course as the quantitative value. However, the columns
# in the dataset shows counts, scores, and section. I'm assuming then,
# that the count refers to the number of students who achieved the score
# and their respective sections for each row. For example, does
# gender play a role on whether or not a student would choose
# to go to a course exclusive to sports application? While there are a lot of
# variables that could affect the score, I would also be interested in
# seeing the grades if the students prior to the professor teaching
# the section. This would tell us if students who tend to perform better
# has a tendency to enroll in the sports section or would they prefer to be
# given a variety of application areas when they are learning their
# lesson.
```

The R session information (including the OS info, R version and all packages used):

```
sessionInfo()

## R version 4.3.0 (2023-04-21 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 11 x64 (build 22621)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8  LC_CTYPE=English_United States.utf8
```

```
## [3] LC_MONETARY=English_United States.utf8 LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: America/New_York
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggplot2_3.4.2 dplyr_1.1.2   plyr_1.8.8    RSQLite_2.3.1
##
## loaded via a namespace (and not attached):
##  [1] bit_4.0.5        gtable_0.3.3     compiler_4.3.0   highr_0.10       crayon_1.5.2
##  [6] tinytex_0.45     tidyselect_1.2.0 Rcpp_1.0.10      blob_1.2.4       scales_1.2.1
## [11] fastmap_1.1.1    R6_2.5.1         labeling_0.4.2   generics_0.1.3   knitr_1.43
## [16] tibble_3.2.1     munsell_0.5.0    DBI_1.1.3        pillar_1.9.0     rlang_1.1.1
## [21] utf8_1.2.3       cachem_1.0.8     xfun_0.39        bit64_4.0.5      memoise_2.0.1
## [26] cli_3.6.1        withr_2.5.0      magrittr_2.0.3   grid_4.3.0       rstudioapi_0.14
## [31] lifecycle_1.0.3  vctrs_0.6.2      evaluate_0.21    glue_1.6.2       farver_2.1.1
## [36] fansi_1.0.4      colorspace_2.1-0 tools_4.3.0      pkgconfig_2.0.3

Sys.time()

## [1] "2023-07-02 00:44:05 EDT"
```