STAT 33B Homework 2

YOUR NAME (YOUR SID)

This assignment is due **Feb 19, 2020** by 11:59pm.

The purpose of this assignment is to practice working with data frames, including loading tabular data, taking subsets, and making plots.

Edit this file, knit to PDF, and:

- Submit the Rmd file on bCourses.
- Submit the PDF file on Gradescope.

If you think you'll need help with submission, please ask in office hours *before* the assignment is due.

Answer all questions with complete sentences, and put code in code chunks. You can make as many new code chunks as you like. Please do not delete the exercises already in this notebook, because it may interfere with our grading tools.

Working with Data

1. In lecture, you saw that the `readRDS()` function can read data stored in R's built-in RDS format. Tabular data is often distributed online as tab-separated value (TSV) or comma-separated value (CSV) files.

   In a TSV file, each row of the data set is one one line, with entries in the columns separated by tabs.

   For this assignment, you'll use the Datasaurus Dozen data set, which is available on the bCourse (`DatasaurusDozen.tsv`).

   Read the documentation for `read.delim()` to figure out how to load the Datasaurus Dozen data set into R.

   Assign the data set to the `dsaur` variable.

```r
# Your code goes here
dsaur = read.delim("DatasaurusDozen.tsv")
```

2. Now that you've loaded the data set, print out summary information, including:
   - Number of columns
   - Number of rows
   - Classes of the columns
   - Levels in the `dataset` column

```r
# Your code goes here
col_num = ncol(dsaur)
row_num = nrow(dsaur)
dsaur_class = class(dsaur$dataset)
dsaur_x = class(dsaur$x)
dsaur_y = class(dsaur$y)
dsaur_level = levels(dsaur$dataset)
```

3. The Datasaurus Dozen is actually a collection of 12 data sets stacked together. The `dataset` column indicates which data set each row comes from.

a. Use subsetting to extract only the rows in the `dino` data set. Assign those rows to the `dino` variable.

b. Compute the mean and standard deviation for the `x` and `y` columns in the `dino` data set.

Repeat these two steps for the `star` dataset.

Based on these statistics, are the two data sets similar?

```r
# Your `dino` code goes here
dino = subset(dsaur, dataset == "dino")
dino_x_mean = mean(dino$x)
dino_x_sd = sd(dino$x)
dino_y_mean = mean(dino$y)
dino_y_sd = sd(dino$y)
```

```r
# Your `star` code goes here
star = subset(dsaur, dataset == "star")
star_x_mean = mean(star$x)
star_x_sd = sd(star$x)
star_y_mean = mean(star$y)
star_y_sd = sd(star$y)
```

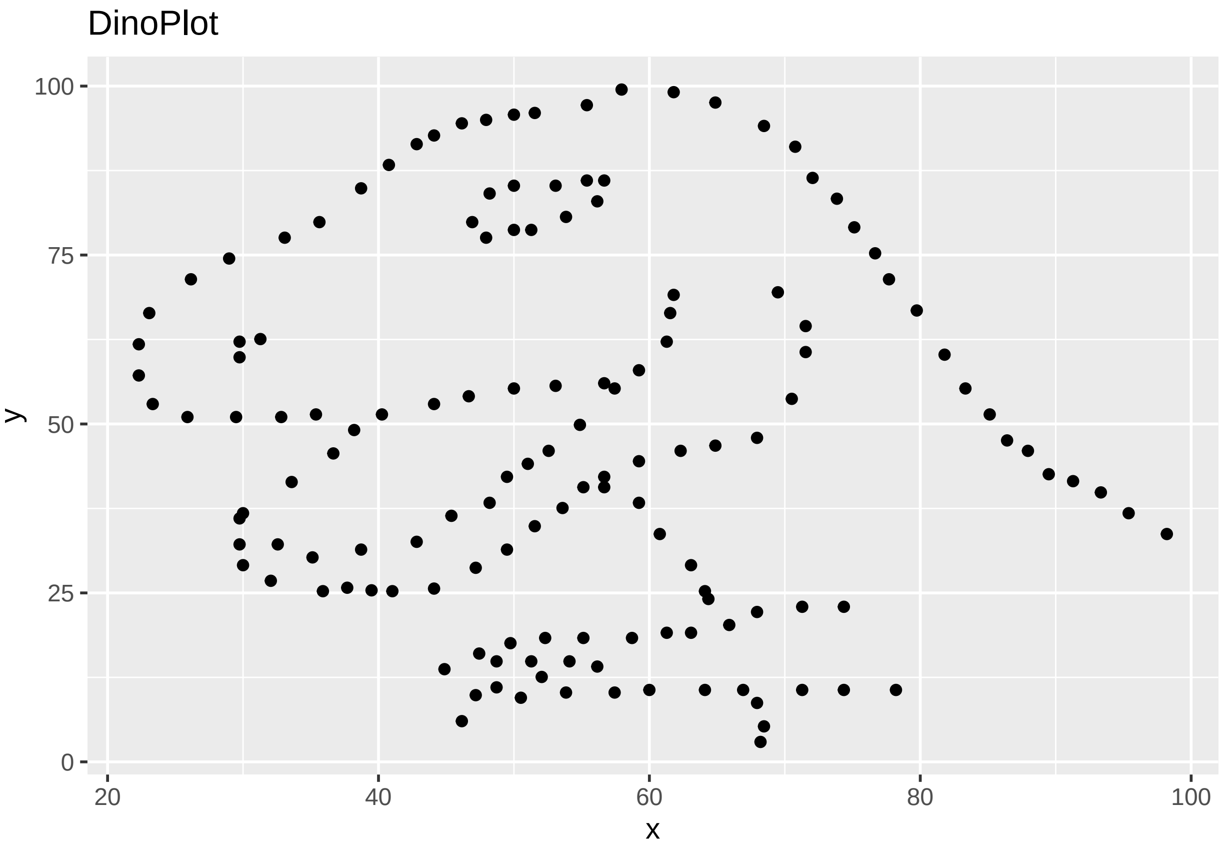
_Your written answer goes here:_They are similar, because the values of their x_mean, y_mean, x_sd, y_sd are very similar.

4. Use `ggplot2` to make a scatter plot of `x` versus `y` for the `dino` data set. Make sure your plot includes a title.

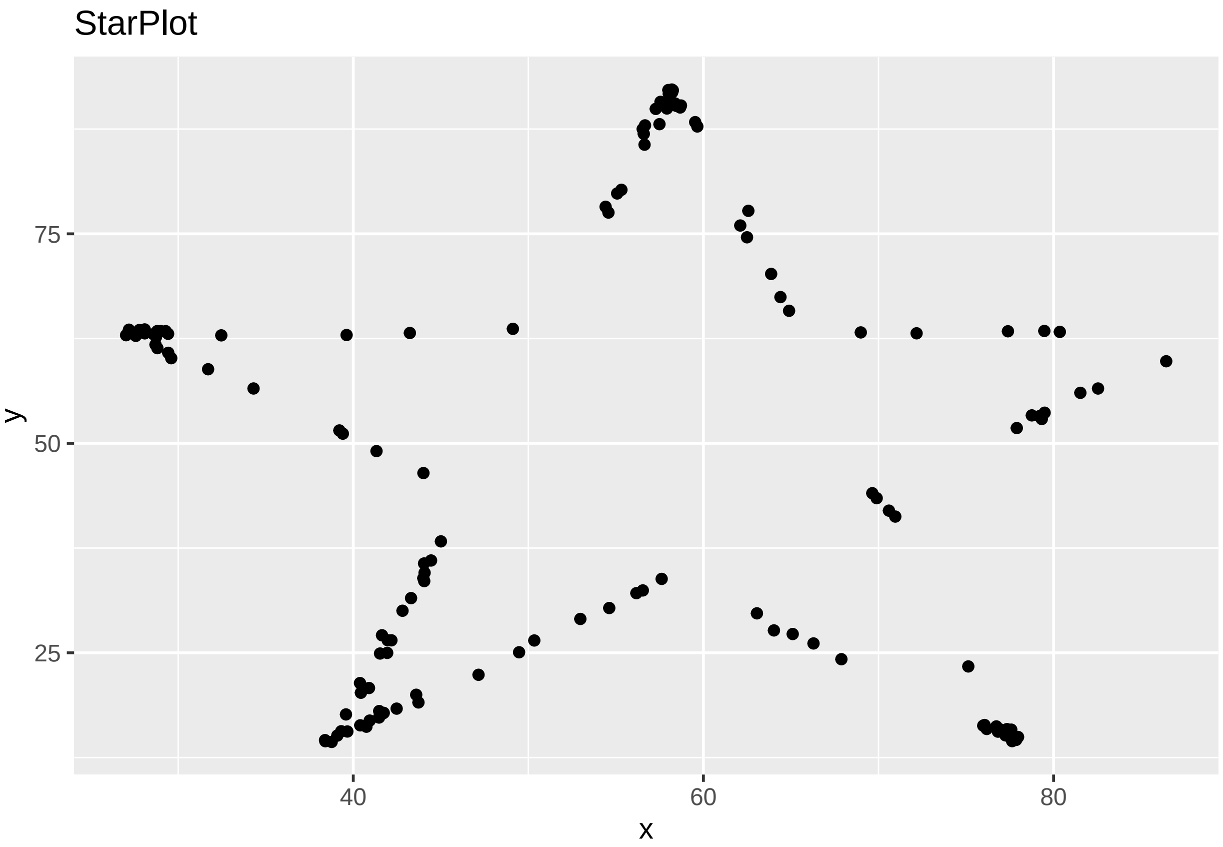
   Repeat for the `star` data set.

   Based on these plots, are the two data sets similar?

```r
# Your `dino` code goes here
library(ggplot2)
ggplot(dino, aes(dino$x,dino$y))+geom_point()+labs(x = "x", y = "y",title = "DinoPlot")
```



```
# Your `star` code goes here
ggplot(star, aes(star$x,star$y))+geom_point()+labs(x = "x", y = "y",title = "StarPlot")
```



_Your written answer goes here:_They are not, not at all. :)