STAT 33B Homework 2

Ming Fong (3035619833)

Sep 24, 2020

This homework is due **Sep 24, 2020** by 11:59pm PT.

Homeworks are graded for correctness.

As you work, write your answers in this notebook. Answer 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.

You need to submit your work in two places:

- Submit this Rmd file with your edits on bCourses.
- Knit and submit the generated PDF file on Gradescope.

Exercise 1

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

Load the Datasaurus Dozen data set and assign it to a variable named `dsaur`.

YOUR ANSWER GOES HERE:

```
dsaur = read.delim("data/DatasaurusDozen.tsv", header = TRUE)
```

Exercise 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
- The range of the `x` column
- The range of the `y` column
- Number of missing values in each column

YOUR ANSWER GOES HERE:

```
ncol(dsaur)
```

```
## [1] 3
```

```
nrow(dsaur)
```

```
## [1] 1846
```

```
unlist(lapply(dsaur, class))
```

```
##     dataset           x           y
## "character"   "numeric"   "numeric"
```

```
levels(factor(dsaur$dataset))
```

```
##  [1] "away"       "bullseye"   "circle"     "dino"       "dots"
##  [6] "h_lines"    "high_lines" "slant_down" "slant_up"   "star"
## [11] "v_lines"    "wide_lines" "x_shape"
```

```
range(dsaur$x)
```

```
## [1] 15.56075 98.28812
```

```
range(dsaur$y)
```

```
## [1]  0.01511933 99.69468014
```

```
sum(is.na(dsaur$dataset))
```

```
## [1] 0
```

```
sum(is.na(dsaur$x))
```

```
## [1] 0
```

```
sum(is.na(dsaur$y))
```

```
## [1] 0
```

Exercise 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.

1. Use subsetting to extract only the rows in the `dino` data set. Assign those rows to the `dino` variable.
2. Compute the mean and standard deviation for the `x` and `y` columns in the `dino` data set.
3. Repeat part 3.1 and 3.2 for the `star` dataset.

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

YOUR ANSWER GOES HERE:

```
dino = subset(dsaur, dataset == "dino")
mean(dino$x)
```

```
## [1] 54.26327
```

```
sd(dino$x)
```

```
## [1] 16.76514
```

```
mean(dino$y)
```

```
## [1] 47.83225
```

```
sd(dino$y)
```

```
## [1] 26.9354
```

```
star = subset(dsaur, dataset == "star")
mean(star$x)
```

```
## [1] 54.26734
```

```r
sd(star$x)
```

```
## [1] 16.76896
```

```r
mean(star$y)
```

```
## [1] 47.83955
```

```r
sd(star$y)
```

```
## [1] 26.93027
```

Both the `dino` and `star` datasets have very similar means and SDs. However this could be missleading.

Exercise 4

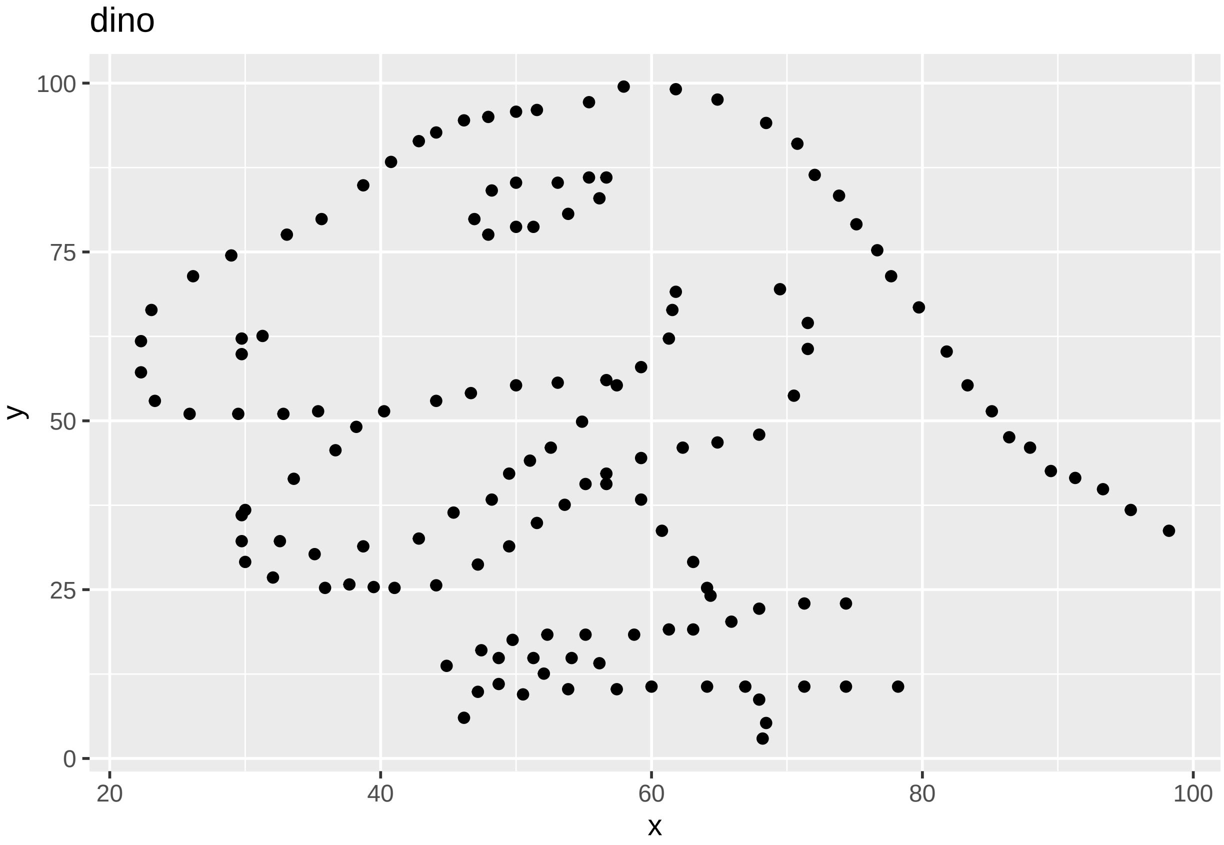
Note: Exercise 4-5 use ggplot2, which will be covered in the week 5 lectures.

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

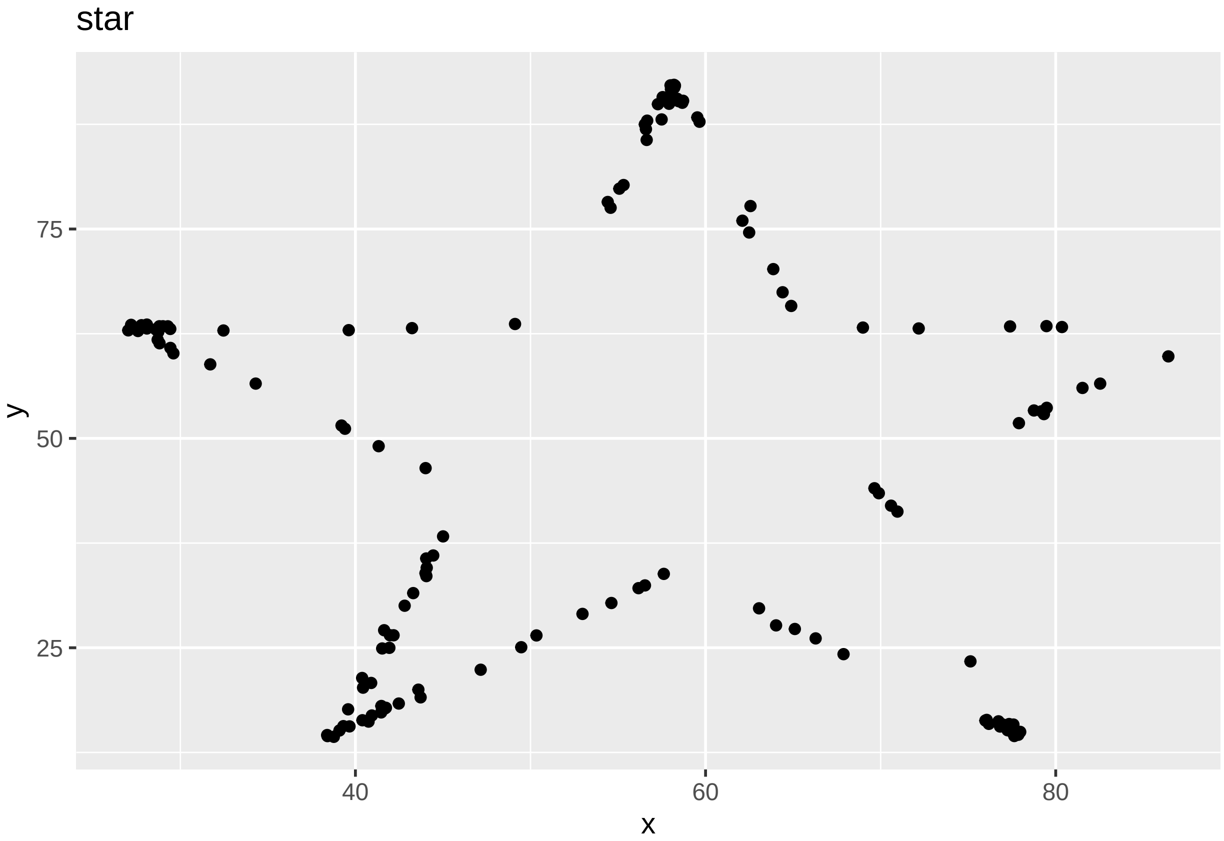
   Based on these plots, are the two data sets similar?

YOUR ANSWER GOES HERE:

```r
library(ggplot2)
ggplot(dino, aes(x = x, y = y)) + geom_point() + labs(title = "dino")
```



```r
ggplot(star, aes(x = x, y = y)) + geom_point() + labs(title = "star")
```



The plots of the two datasets are not similar. `dino` plots a dinosaur image while `star` plots a five-pointed star.

Exercise 5

A "faceted" plot is one that shows several subplots side-by-side, to aid comparison between them. Each subplot is called a "facet".

You can create a faceted plot with ggplot2 by using the facet layer. For instance, the `facet_wrap()` function creates a line of facets based on a single categorical variable. The facet layer should be added to a plot *after* the geometry layers.

1. Read the documentation for `facet_wrap()`, then create a faceted scatter plot that shows each dataset from the Datasaurus Dozen in a separate facet. Use `geom_smooth` with `method = "lm"` to add a linear regression line to each facet.

   Hint: Unlike other ggplot2 functions, variable names in facet functions need to be enclosed in a call to the `vars()` function. So to write the column `dataset`, you would write `vars(dataset)`. See the `facet_wrap()` documentation for more details.

2. Is there any pattern to the regression lines across the different data sets?

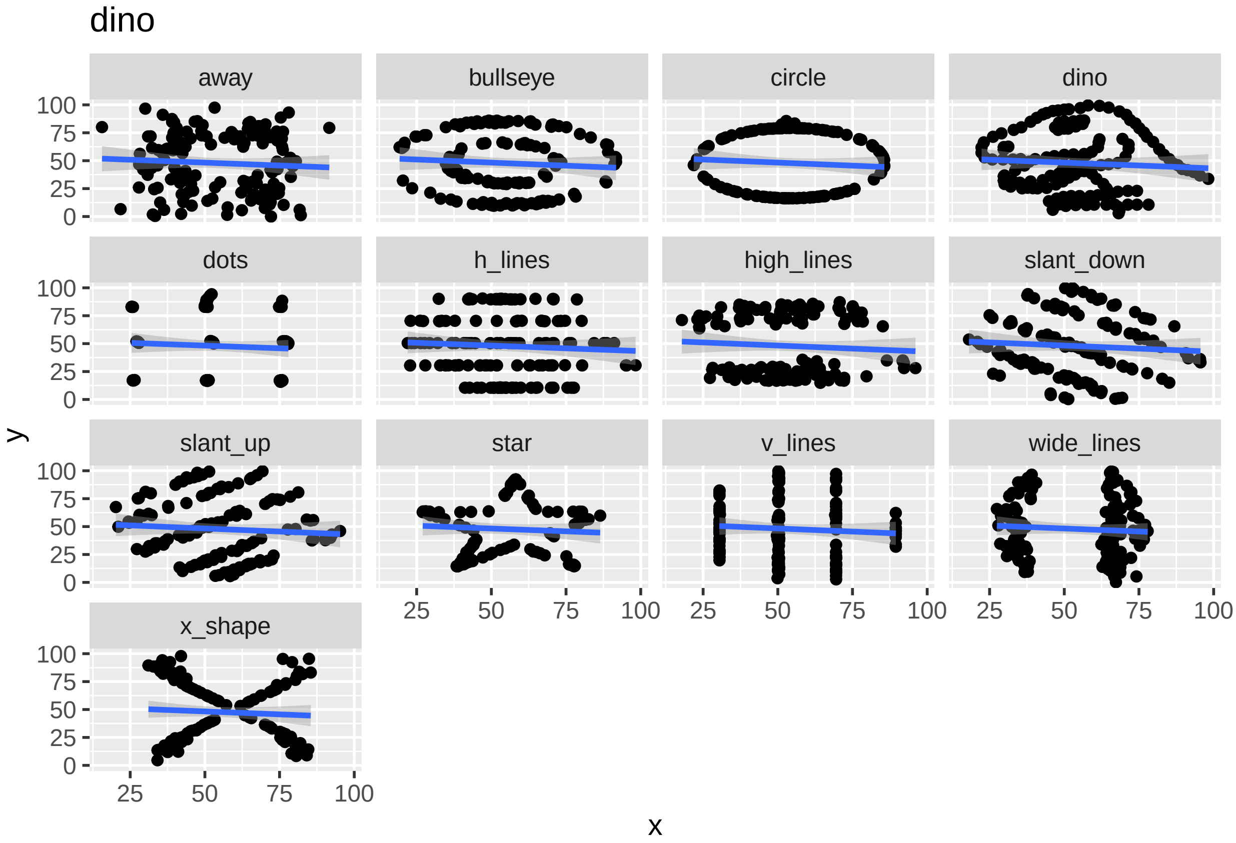
YOUR ANSWER GOES HERE:

1.

```r
plot = ggplot(dsaur, aes(x = x, y = y)) +
  geom_point() + labs(title = dsaur$dataset) +
  geom_smooth(method = "lm")

plot + facet_wrap(vars(dataset))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



2. The regression lines of all the plots are very similar if not identical. All the lines have a slight negative slope. However, the actual points of each dataset are completely different from each other. The very similar statistical summaries masks this.