Reading, Summarizing, Visualizing Data

Mark Andrews
Psychology Department, Nottingham Trent University

In this note, we'll read in a data set from a csv file. We'll then get some summary statistics on the variables, and then visualize the data.

Make sure the necessary R packages are loaded

We'll need three packages:

- `readr` for reading in the csv file
- `dplyr` for doing summaries on the data
- `ggplot2` for visualizing and plotting the data

These and any other package can be loaded by using the `library()` command like this:

```
library(readr)
library(dplyr)
library(ggplot2)
```

Also, if these packages are not yet installed, you must install them. They can all be installed by installing the package-of-packages called `tidyverse`.

Read in the data

The data file we will be using is called `fake_data_01.csv`. If this data file is in your so-called "working directory", then you can read it in, and save it by the name `Df`, with the following command:

```
Df <- read_csv('fake_data_01.csv')
```

You can also effectively run the above command by going to the `Environment` tab in Rstudio (upper right panel), and choose "Import Dataset" and then "From Text (readr)..".

Let's look at the data

- First, we can just type the name of the data-frame. By default, you'll see just the first 10 rows, and some general info about the size of the data set.

```
Df
```

```
## # A tibble: 100 x 3
##    sex       iq test_score
##    <chr>  <int>      <dbl>
##  1 male      74       14.1
##  2 female    83       16.0
##  3 female    88       15.6
##  4 male     105       11.8
##  5 male      92       13.5
##  6 male      92       13.4
##  7 male     102       12.7
##  8 female    80       14.9
##  9 male     102       14.1
## 10 female   114       15.7
## # ... with 90 more rows
```

- And then we can get another look via `glimpse`:

```
glimpse(Df)
```

```
## Observations: 100
## Variables: 3
## $ sex        <chr> "male", "female", "female", "male", "male", "male",...
## $ iq         <int> 74, 83, 88, 105, 92, 92, 102, 80, 102, 114, 107, 80...
## $ test_score <dbl> 14.14571, 16.01274, 15.64022, 11.82935, 13.48280, 1...
```

Summarize the data

- We can always get a general summary of a data frame via the `summary` command:

```
summary(Df)
```

```
##      sex                   iq          test_score
##  Length:100         Min.   : 60.0   Min.   :11.49
##  Class  :character  1st Qu.: 91.0   1st Qu.:13.26
##  Mode   :character  Median :102.0   Median :14.63
##                     Mean   :100.4   Mean   :14.50
##                     3rd Qu.:108.2   3rd Qu.:15.65
##                     Max.   :138.0   Max.   :18.40
```

- But we can get more specific information using the `summarize` (also known as `summarise`) command. With this command, we can ask for many different summary statistics, on as many or as few variables as we are interested in. Let's say we want to know the average `iq`, average `test_score`, standard deviation of `iq` and the variance of `test_score`. We'll also get the number of data points we have overall, using the `n()` function. Then we do the following:

```
summarize(Df,
          average_iq = mean(iq),
          average_score = mean(test_score),
          iq_stdev = sd(iq),
          score_var = var(test_score),
          n = n()
)
```

```
## # A tibble: 1 x 5
##   average_iq average_score iq_stdev score_var     n
##        <dbl>         <dbl>    <dbl>     <dbl> <int>
## 1       100.          14.5     14.5      2.30   100
```

Summarize by a group

We can get summary statistics per each group by doing a `group_by` operation first. So let's say we wanted to get, as before, the average `iq`, average `test_score`, standard deviation of `iq` and the variance of `test_score`, but this time, we want these summary statistics separately for the males and females, then we do this:

```
summarize(group_by(Df, sex),
          average_iq = mean(iq),
          average_score = mean(test_score),
```

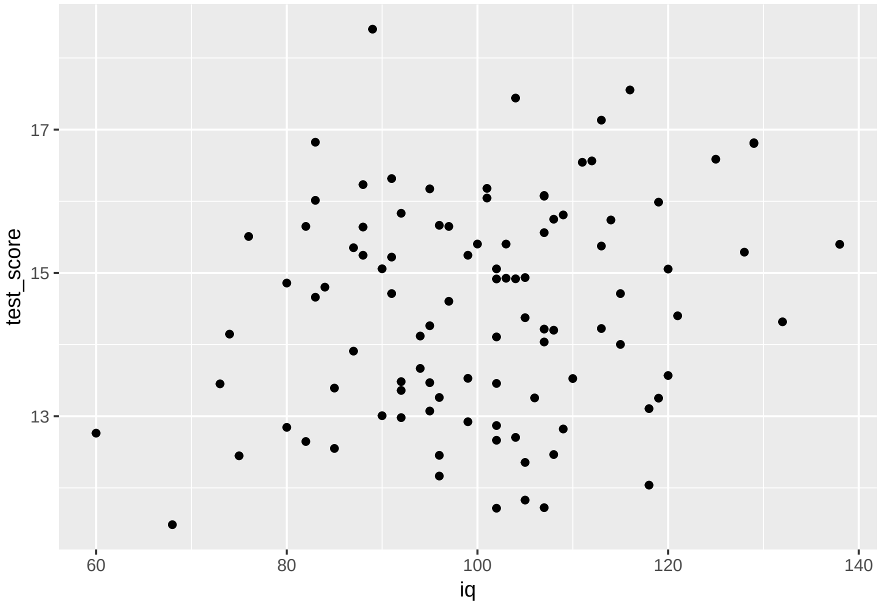

Figure 1: A scatterplot of each participant's test score as a function of their IQ score.

```
          iq_stdev = sd(iq),
          score_var = var(test_score),
          n = n()
)
```

```
## # A tibble: 2 x 6
##   sex    average_iq average_score iq_stdev score_var     n
##   <chr>       <dbl>         <dbl>    <dbl>     <dbl> <int>
## 1 female      101.           15.4     15.6      1.46    51
## 2 male         99.6          13.6     13.4      1.48    49
```

Visualization

- First, let's make a scatterplot. We can do so using the following `ggplot` code:

```
ggplot(Df,
       aes(x = iq, y = test_score)
) + geom_point()
```

The resulting figure is shown in Figure 1.

- We can add a line of best fit by adding `stat_smooth(method='lm')` like this:

```
ggplot(Df,
       aes(x = iq, y = test_score)
) + geom_point() +
  stat_smooth(method='lm')
```

This figure is shown in Figure 2.

- We can colour the points according to whether they are male of female by adding `col = sex` within the `aes()` command like this:

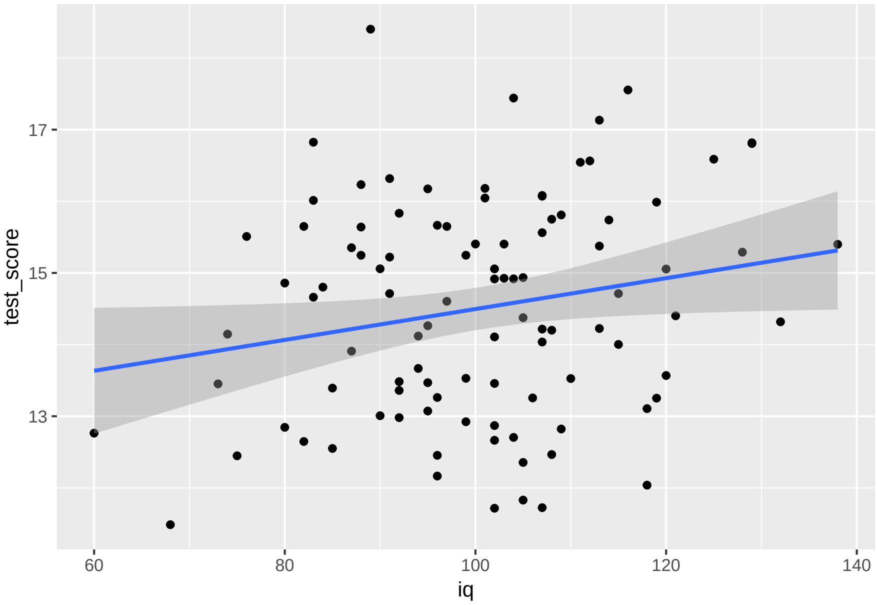

Figure 2: A scatterplot of each participant's test score as a function of their IQ score. Also shown is line of best fit to the data.

```
ggplot(Df,
       aes(x = iq, y = test_score, col=sex)
) + geom_point()
```

This is shown in Figure 3.

- And we can make a line of best fit for the males and females separately:

```
ggplot(Df,
       aes(x = iq, y = test_score, col=sex)
) + geom_point() +
  stat_smooth(method='lm')
```

This is shown in Figure 4.

- We can change the style in many ways. Here, is one possibility:

```
ggplot(Df,
       aes(x = iq, y = test_score, col=sex)
) + geom_point() +
  stat_smooth(method='lm', se=F) +
  theme_classic() +
  xlab('IQ') +
  ylab('Test score') +
  ggtitle('A very important figure')
```

This is shown in Figure 5.

Tukey boxplots

```
fig_cap <- "A Tukey boxplot showing the distribution of the test score variables for males and females
```

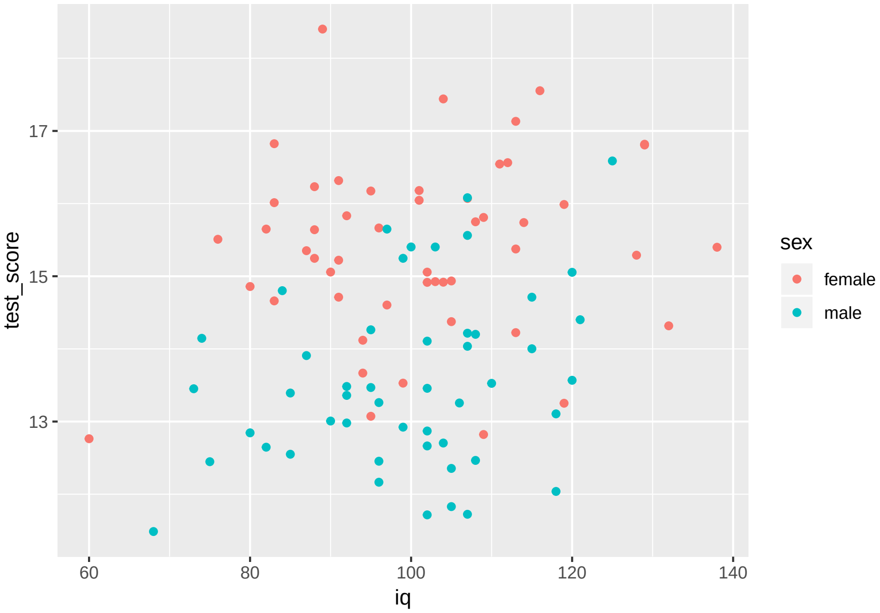

Figure 3: A scatterplot of each participant's test score as a function of their IQ score. The data points froms males and females are identified by different colours.

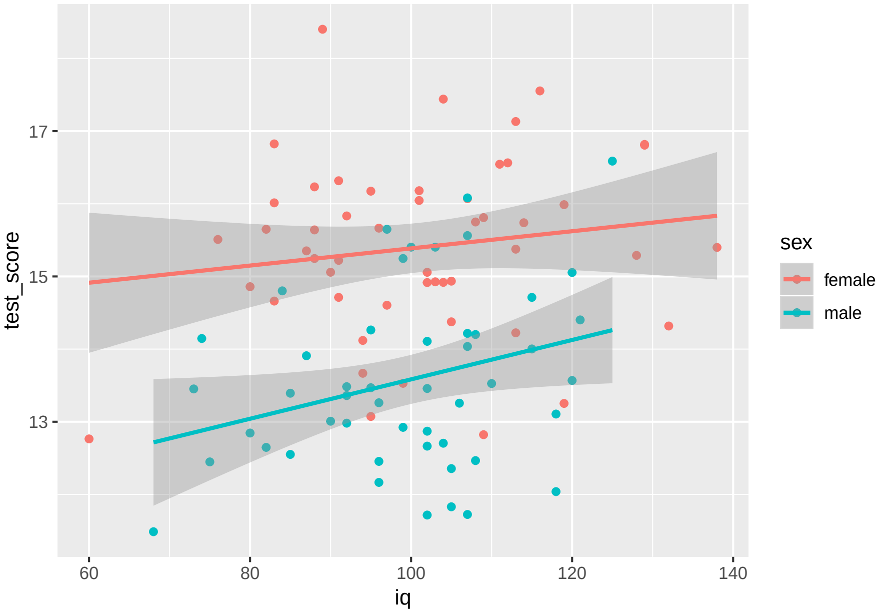

Figure 4: A scatterplot of each participant's test score as a function of their IQ score. The data points froms males and females are identified by different colours. The lines of best fit for the data from males and females separately is also shown.

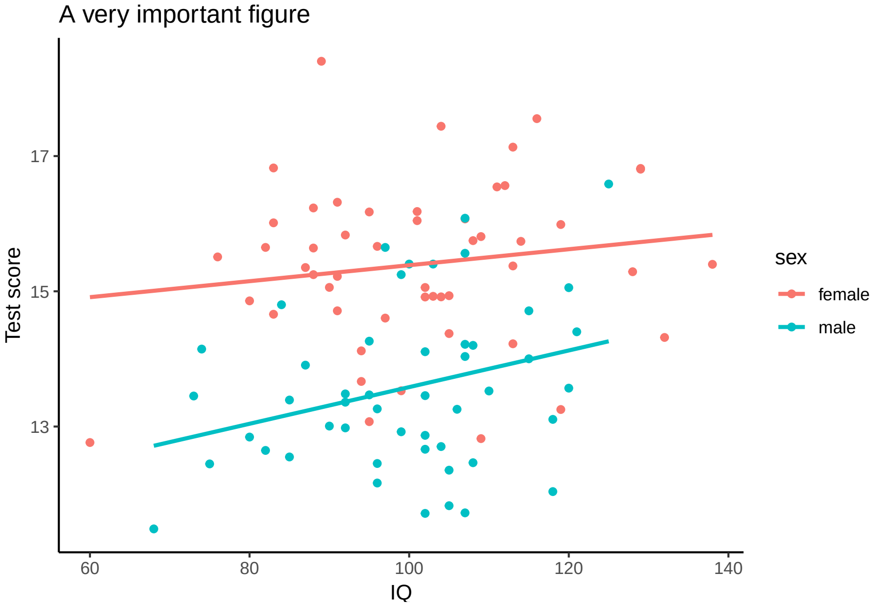

Figure 5: A scatterplot of each participant's test score as a function of their IQ score. The data points froms males and females are identified by different colours. The lines of best fit for the data from males and females separately is also shown. In this figure, we change the plotting style, add new axis labels, and add a figure title.

To visualize distributions of data, a venerable old method is the Tukey boxplot. Here, we will visualize the distribution of `test_score` for the males and females separately.

```
ggplot(Df,
       aes(x = sex, y = test_score)
) + geom_boxplot()
```

The resulting figure is shown in Figure 6. By default, boxplots show, in the box part of the plot, the lower quartile, median, and upper quartile values. The whiskers extend from the upper and lower quartiles to the furtherest points that are not outliers, where outliers are defined as being within 1.5 times the inter-quartile range from the box. We can, in addition, provide all of the points by adding a so-called jitter plot on top of the boxplot.

```
ggplot(Df,
       aes(x = sex, y = test_score)
) + geom_boxplot() +
  geom_jitter()
```

The resulting figure, Figure 7, is a bit of unsightly. But we can make it more presentable very easily.

```
ggplot(Df,
       aes(x = sex, y = test_score, col = sex)
) +
  geom_boxplot(width=0.2, outlier.shape = NA) +
  geom_jitter(width = 0.1, size=0.5, alpha=0.5, fill=NA) +
  theme_classic() +
  xlab('Sex') +
  ylab('Test score') +
  guides(col = FALSE)
```

The resulting figure in Figure 8.

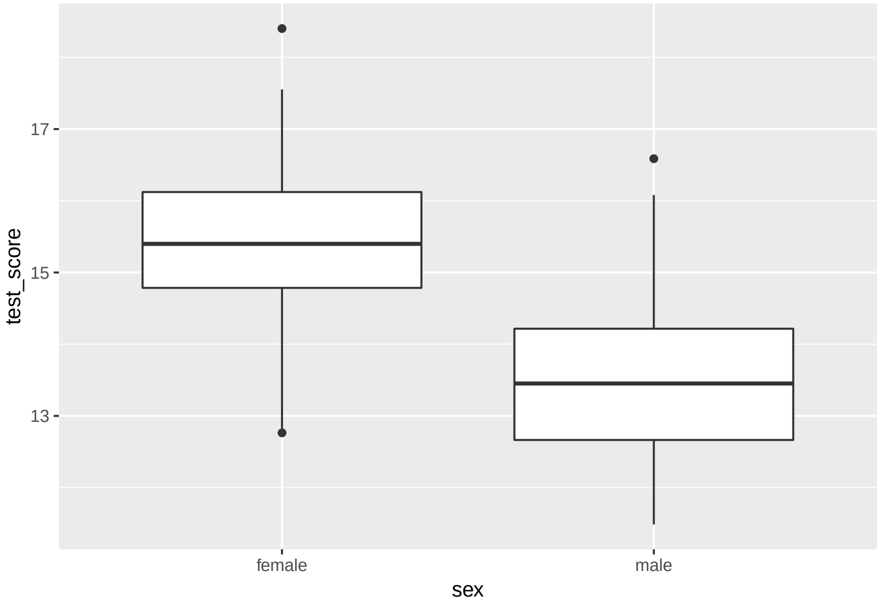

Figure 6: A Tukey boxplot showing the distribution of the test score variables for males and females separately.

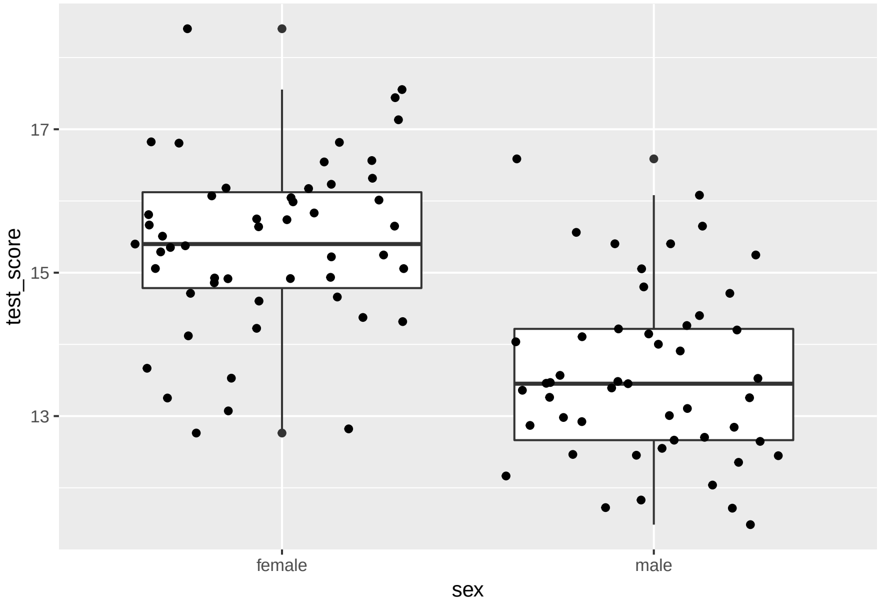

Figure 7: A Tukey boxplot showing the distribution of the test score variables for males and females separately. In adddition, a jitter plot of all points is shown.

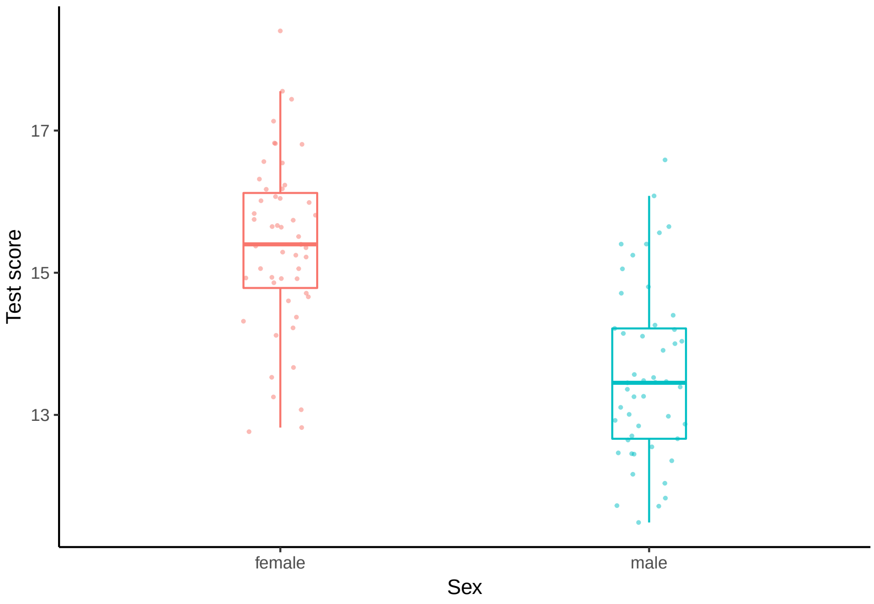

Figure 8: A Tukey boxplot showing the distribution of the test score variables for males and females separately. In adddition, a jitter plot of all points is shown. This version is just a bit nicer to look at than the previous one.