## **Basic Statistics & Plotting**

## You've already learned a lot!

Hopefully you're feeling more comfortable with some of terminology used in programming:

- Objects
- Classes
- Functions
- Arguments
- Vectors
- Data.frames
- Etc.

### Uhh...no?

# You already know everything you need to do stats & plots!

#### **Statistics**

All statistics are computed with functions.

If you know the type of analysis you want to run, find the corresponding function and go for it!

#### **Plotting**

All plots are made with functions.

Slightly different is that one particular package is *a lot* better at plotting than base R.

## **Today**

- Very basic statistics
- Introduction to plotting with ggplot2
  - There will be MUCH more on data visualization later
- PRACTICE PRACTICE

### About the MIDUS dataset

#### Variables available in this data file:

- Demographic variables: age, sex
- Physical health variables: self-rated physical health, heart problems, father had heart attack, BMI
- Mental health variables: self-rated meantal health, self-esteem, life satisfaction (life overall, work, health, relationship with spouse/partner, relationship with children), hostility (stress reactivity & agression)

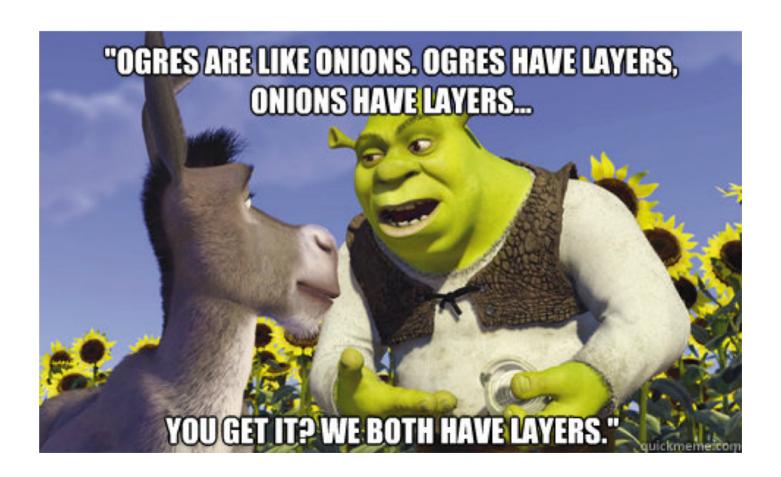
#### Please load in midus, make sure:

- Make sure the variables sex, heart\_self, and heart\_father are factor()
   variables (rather than characters)
- Use the same na.omit() function to remove all NA values

## Before we begin...

- Check to make sure you have the ggplot2 package installed
- Check to make sure the ggplot2 package is *loaded*
- If "no" to either, how can you solve this?

### Data visualization with ggplot2



ggplot2 has the following structure:

```
ggplot(things that impact the entire plot) +
  geom_something(things that impact just the something)
```

ggplot2 has the following structure:

```
ggplot(things that impact the entire plot) +
  geom_something(things that impact just the something)
```

#### Things like:

- data.frame used for plotting
- defining your x & y axes

ggplot2 has the following structure:

```
ggplot(things that impact the entire plot) +
  geom_something(things that impact just the something)
```

geom\_ typically means **shape**. What shapes do you want to use to represent your data in the plot?

- geom\_histogram -- histogram
- geom\_density -- distributions
- geom\_violin -- distributions
- geom\_point -- scatter plot
- geom\_col -- bar plot

The functions ggplot() and geom\_() can take on different aesthetics as an argument, using aes().

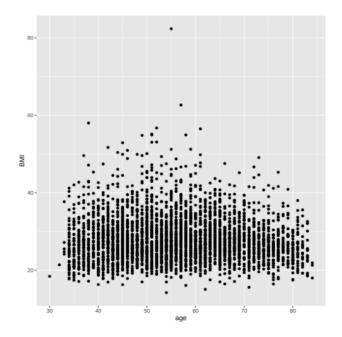
Aesthetics are how you control what you want your plot to look like; how can you make it pretty? Examples:

- Which variables are the x- and y- axes?
- color (should you color the plot by some variable?)
- fill (very similar to color, should you fill the plot in somehow; used for bar graphs and boxplots)
- shape (do you want groups to have different shaped points?)
- size (how big should plotted data be?)

Note: person that made this package is from New Zealand; the British spellings and American spellings work! Although using tab-complete my auto-fill the British spellings

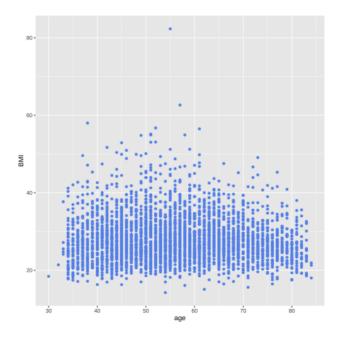
- Usually aes() contains some information that comes directly from the data
- If the information is *not* based on the data, it does not need to be inside an aes() argument.

```
ggplot(data = midus, aes(x = age, y = BMI)) +
   geom_point()
```



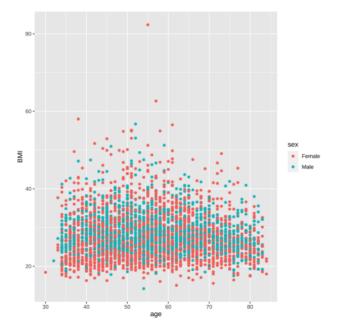
- Usually aes() contains some information that comes directly from the data
- If the information is *not* based on the data, it does not need to be inside an aes() argument.

```
ggplot(data = midus, aes(x = age, y = BMI)) +
   geom_point(color = "cornflowerblue")
```



- Usually aes() contains some information that comes directly from the data
- If the information is *not* based on the data, it does not need to be inside an aes() argument.

```
ggplot(data = midus, aes(x = age, y = BMI)) +
   geom_point(aes(color = sex))
```



### **Exercise 1**

Make a scatter plot of self\_esteem (x-axis) against life\_satisfaction (y-axis)

Make the points of the scatter plot a different shape based on the sex variable (for example, males might be circles and females might be squares)

Make the color of the points different based on sex

Set the size of all points equal to 3

### Remember this???

## Statistical Analyses

We're going to practice plotting with ggplot2 while learning some really basic statistical tests.

Is there a difference in hostility between men and women in the midus sample?

**Statistic: T-test** 

Plot: boxplot

### A note about formulas

You can read the ~ (tilda) as "by" or "predicted by"

hostility ~ sex means...

- "hostility by sex"
- "is hostility predicted by sex?"

### **T-tests**

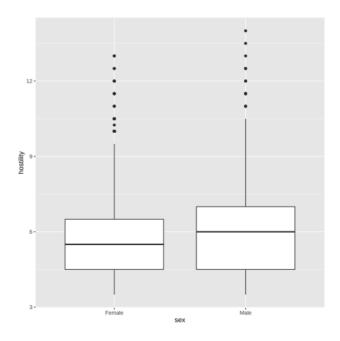
Is there a difference in hostility between men and women in the midus sample?

```
t.test(hostility ~ sex,
       data = midus)
##
##
      Welch Two Sample t-test
##
## data: hostility by sex
## t = -6.097, df = 3519.4, p-value = 1.198e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
   -0.4491034 -0.2305455
##
## sample estimates:
## mean in group Female mean in group Male
               5.638040
                                    5.977864
##
```

### **T-tests**

Is there a difference in hostility between men and women in the midus sample?

```
ggplot(data = midus, aes(x = sex, y = hostility)) +
  geom_boxplot()
```



### **Correlations**

Does self\_esteem correlate with life\_satisfaction?

Statistic: Correlation Plot: scatter plot

#### Lots of options for correlations!

- cor() gives straight correlation; no frills
- cor.test() gives probabilities but only for one pair of values at a time
- corr.test() is part of the psych package and reports sample sizes along with probabilities

### **Correlations**

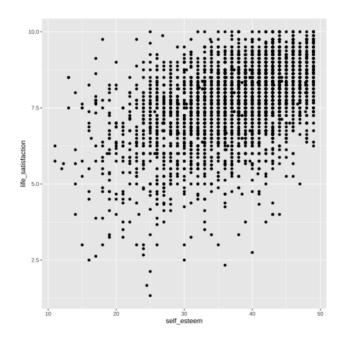
#### Does self\_esteem correlate with life\_satisfaction?

```
##
## Pearson's product-moment correlation
##
## data: midus$self_esteem and midus$life_satisfaction
## t = 34.292, df = 3738, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.4644257 0.5131989
## sample estimates:
## cor
## 0.4891947</pre>
```

### **Correlations**

#### Does self\_esteem correlate with life\_satisfaction?

```
ggplot(data = midus, aes(x = self_esteem, y = life_satisfaction)) +
  geom_point()
```



Say you wanted to dichotomize your self\_esteem variable into those with high self-esteem (above the mean) and those with low self-esteem (below the mean).

You want to see if sex and your newly dichotomized self\_esteem variables predict BMI.

**Statistic: 2x2 ANOVA** 

Plot: bar plot

## Dichotomizing variables

#### As a general rule, don't do this

BUT, it does make for a nice teaching example  $\stackrel{\textbf{\tiny (4)}}{=}$ 

```
sex age self_esteem_di
##
        ID
## 1 10001 Male
                   61
                                high
## 2 10002 Male 69
                                low
## 6 10011 Female 52
                                high
## 8 10015 Female 53
                                low
## 10 10018
             Male
                   49
                                high
```

Does sex and your newly dichotomized self\_esteem variable predict BMI? (no interaction)

### Does sex and your newly dichotomized self\_esteem variable predict BMI? (WITH interaction)

```
anova2 <- aov(BMI ~ sex * self_esteem_di, data = midus)
summary(anova2)</pre>
```

#### Does sex and your newly dichotomized self\_esteem variable predict BMI?

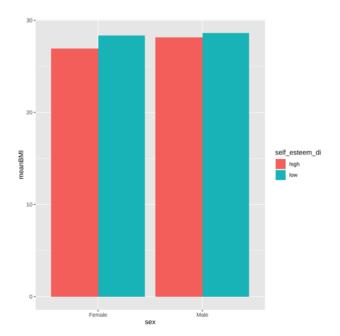
Bar plots suck. Height of each bar should reflect that group's **mean**. So we first need to calculate the means, and store them in a data.frame.

```
femaleHigh <- subset(midus, sex == "Female" & self_esteem_di == "high")</pre>
femaleHighMean <- mean(femaleHigh$BMI)</pre>
femaleLow <- subset(midus, sex == "Female" & self_esteem_di == "low")</pre>
femaleLowMean <- mean(femaleLow$BMI)</pre>
maleHigh <- subset(midus, sex == "Male" & self_esteem_di == "high")</pre>
maleHighMean <- mean(maleHigh$BMI)</pre>
maleLow <- subset(midus, sex == "Male" & self_esteem_di == "low")</pre>
maleLowMean <- mean(maleLow$BMI)</pre>
meansData <- data.frame(sex = c("Female", "Female", "Male"),</pre>
                          self_esteem_di = c("high", "low", "high", "low")
                          meanBMI = c(femaleHighMean,
                                       femaleLowMean,
                                       maleHighMean,
                                                                             28 / 35
                                       maleLowMean))
```

#### Does sex and your newly dichotomized self\_esteem variable predict BMI?

Then we can plot, using our **NEW** data.frame (Note: we will cover a **MUCH** easier way of doing this when we talk about **tidyverse** in the next section)

```
ggplot(data = meansData, aes(x = sex, y = meanBMI)) +
  geom_col(aes(fill = self_esteem_di), position = "dodge")
```



### Regression

- 1. Is life\_satisfaction predicted by self\_esteem?
- 2. Are self\_esteem and hostility both independent predictors of life\_satisfaction?
- 3. Is there an interaction between self\_esteem and hostility predicting life\_satsifaction?

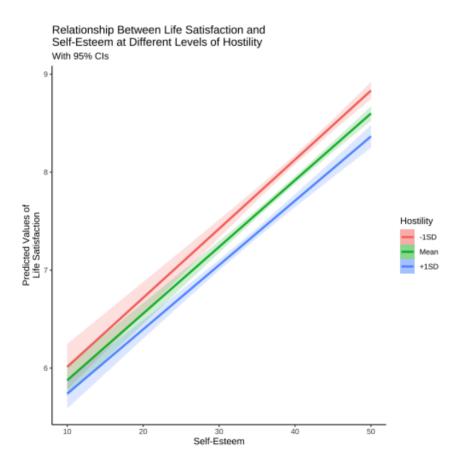
Statistic: Simple & Multiple Regression
Plot: scatter plot with mean, +1SD, and -1SD of
hostility

## Regression

- 1. Is life\_satisfaction predicted by self\_esteem?
  - Simple regression
  - o lm(life\_satisfaction ~ self\_esteem, data = midus)
- 2. Are self\_esteem and hostility both independent predictors of life\_satisfaction?
  - Multiple regression; no interaction
  - o lm(life\_satisfaction ~ self\_esteem + hostility, data =
     midus)
- 3. Is there an interaction between self\_esteem and hostility predicting life\_satsifaction?
  - Multiple regression; with interaction
  - o lm(life\_satisfaction ~ self\_esteem \* hostility, data =
     midus)

## An extra pacakge: ggeffects

Has a function called ggpredict that makes it very easy to visualize interactions of continuous variables.



## Things to note about regression

- Assign your lm() object to your global environment. You can get coefficients, predicted values etc.
- If you want the relationship betwen X1 and Y, *after controlling for* X2, you can make a scatter plot with the model's *fitted* values.
- If you want to view the output table of a regression, use summary() (just like we did with ANOVA).
- If you want to be able to extract the  $\mathbb{R}^2$ , F-statistic etc., assign the summary (model) object to your global environment.
- Check out the <a href="https://example.com">broom</a> package to format your regression outputs into a nice data.frame.

### R Resources

The **only** way to get better is to **PRACTICE!** Some helpful resources:

- Online tutorials like Coursera, Code School/Pluralsight, and Code Academy
- swirl package helps you learn R from inside RStudio! Strong recommend!
- Favorite websites for reference:
  - Quick-R
  - Cookbook for R
  - STACK OVERFLOW (almost always top answer from Google search)
- Reddit has a shocking number of R-related subreddits
- Jenine Harris's new **Statistics with R** book
- R for Data Science is very tidyverse-heavy; go through our tidyverse
  portion first, then check the book out
- Learning Statistics with R by Danielle Navarro; textbook for grad stats
- #rstats on Twitter is a huge and welcoming community!

### Google, Google!

## Congratulations!

You made it through our R Basic Training!

#### Up next:

- Learn to clean and prepare your data more effectively with tidyverse. This is a HUGE part of the R ecosystem, so please don't skip this! It will make your life a lot easier!
- How to generate reports (PDF, Word, or HTML) files that integrate your thoughts and your code. This is the core of reproducibility and will allow you to share code with your advisors, collaborators, and journals in a much prettier and easier manner.
- Here we covered the basics of plotting with ggplot2, but learn just how flexible it can be for data visualization. Make your plots incredible!

