Introduction to R Workshop

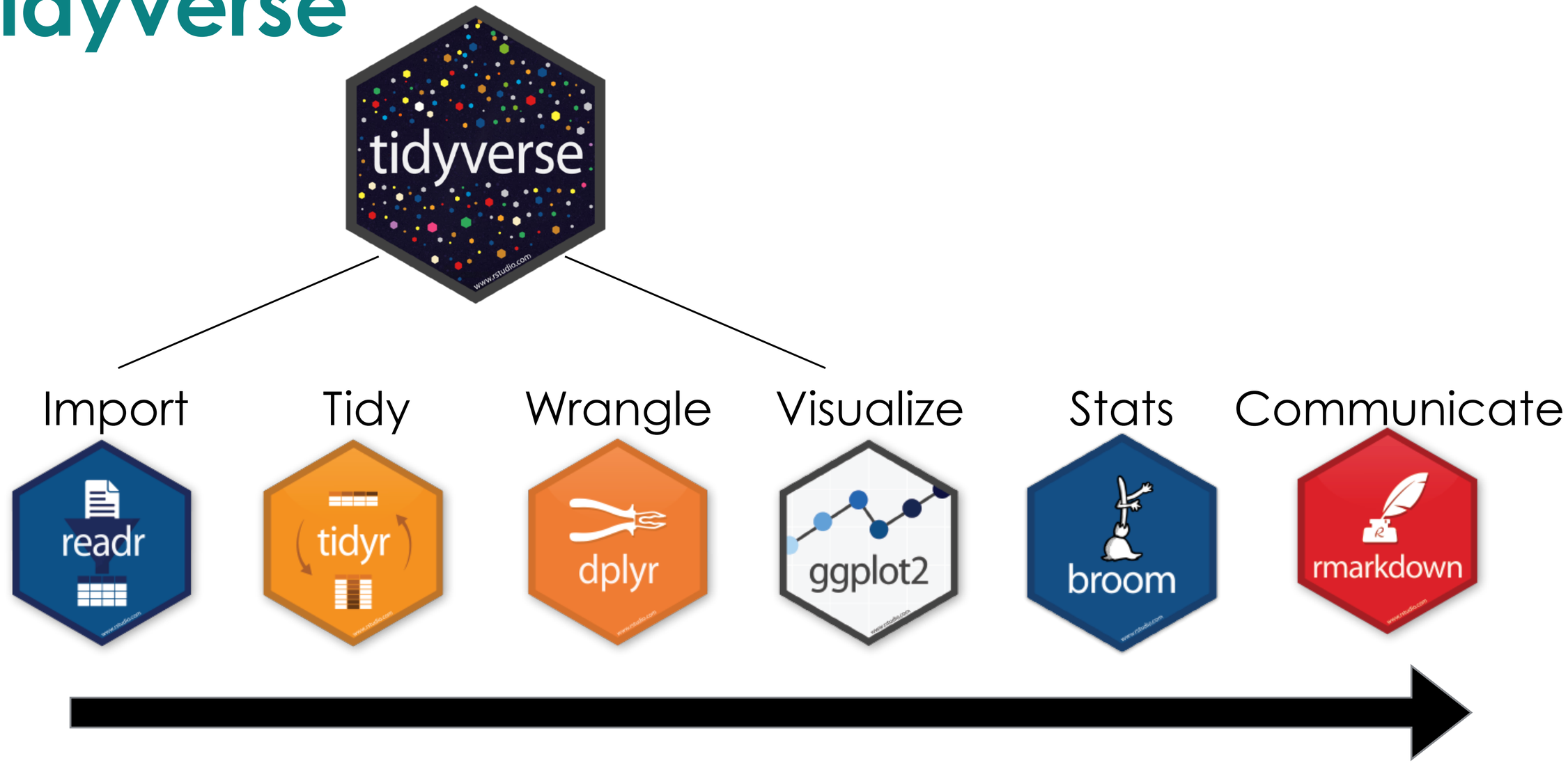
Session 4

Sean Nguyen

MSU BEST

Broadening Experiences in Scientific Training

Data Analysis in the **Tidyverse**



Data Analysis in the **Tidyverse**

Import

Tidy

Wrangle

Visualize

Stats

t.test()
aov()
TukeyHSD()
tidy()

Communicate

Session 4: Goals

Data carpentry

Tidyverse review

Statistical tests

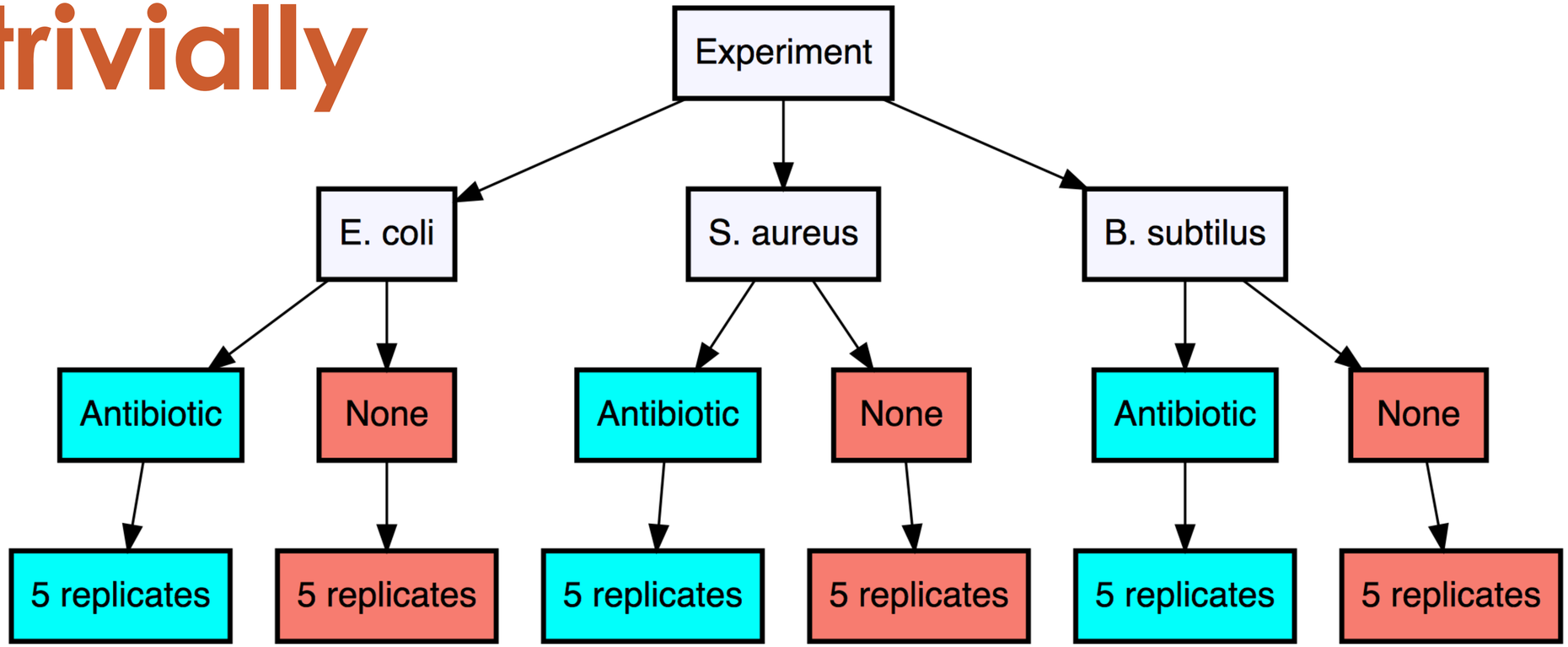
Name files so they're human readable and organize themselves by default

YYYY-MM-DD

figure 1.png
fig 1.png
myabstractMay2014.docx
Henry's file uses spaces and punctuation.xlsx
lolgoodluckfinding^thisfilein2years.doc

fig01_Scatterplot-of-KO-animals.png
fig02_Histogram-of-plants-by-year.png
2012-04-17_vacation-packing-list.docx
2017-10-31_survey-data-from-orientation.xlsx

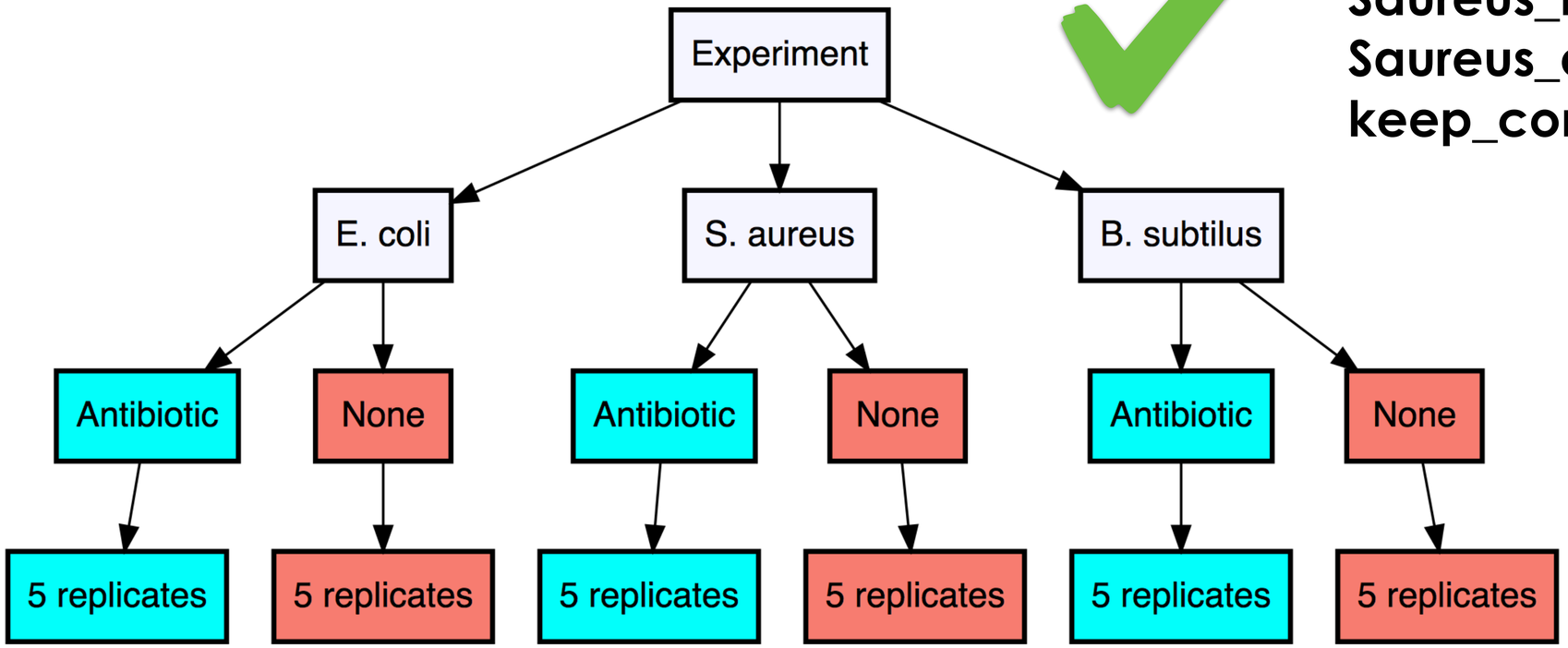
Clear nomenclature design makes data organization and analysis trivially easy



Consistently label your experimental samples

E.coli No antibiotic Replicate 1
S.aureus Antibiotic 50ug/ml Rep. 1
Really-inconsistent_file-to Read/organize

Ecoli_antibiotic_exp1_rep1
Saureus_none_exp1_rep2
Saureus_antibiotic_50_exp2_rep4
keep_consistent_formatting

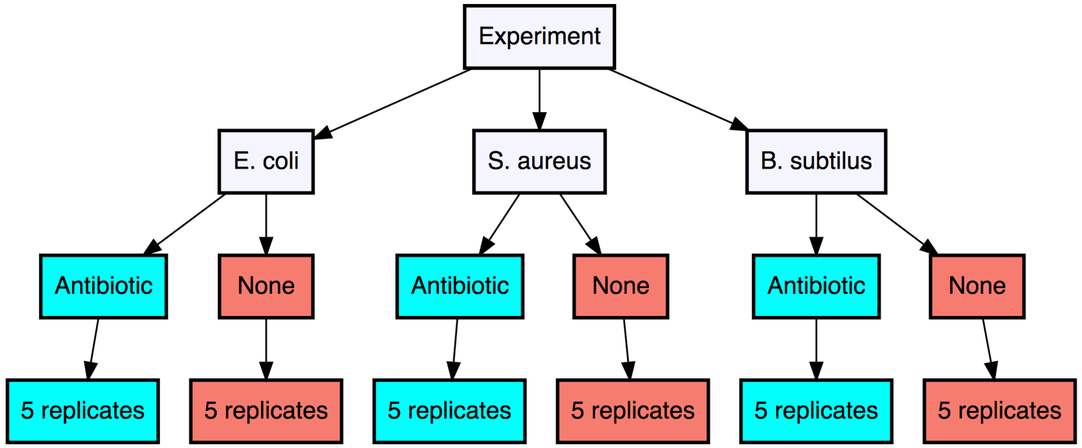


Data Carpentry

Naming Files

2015-08-17_Descriptive_File_Name

Experimental Design

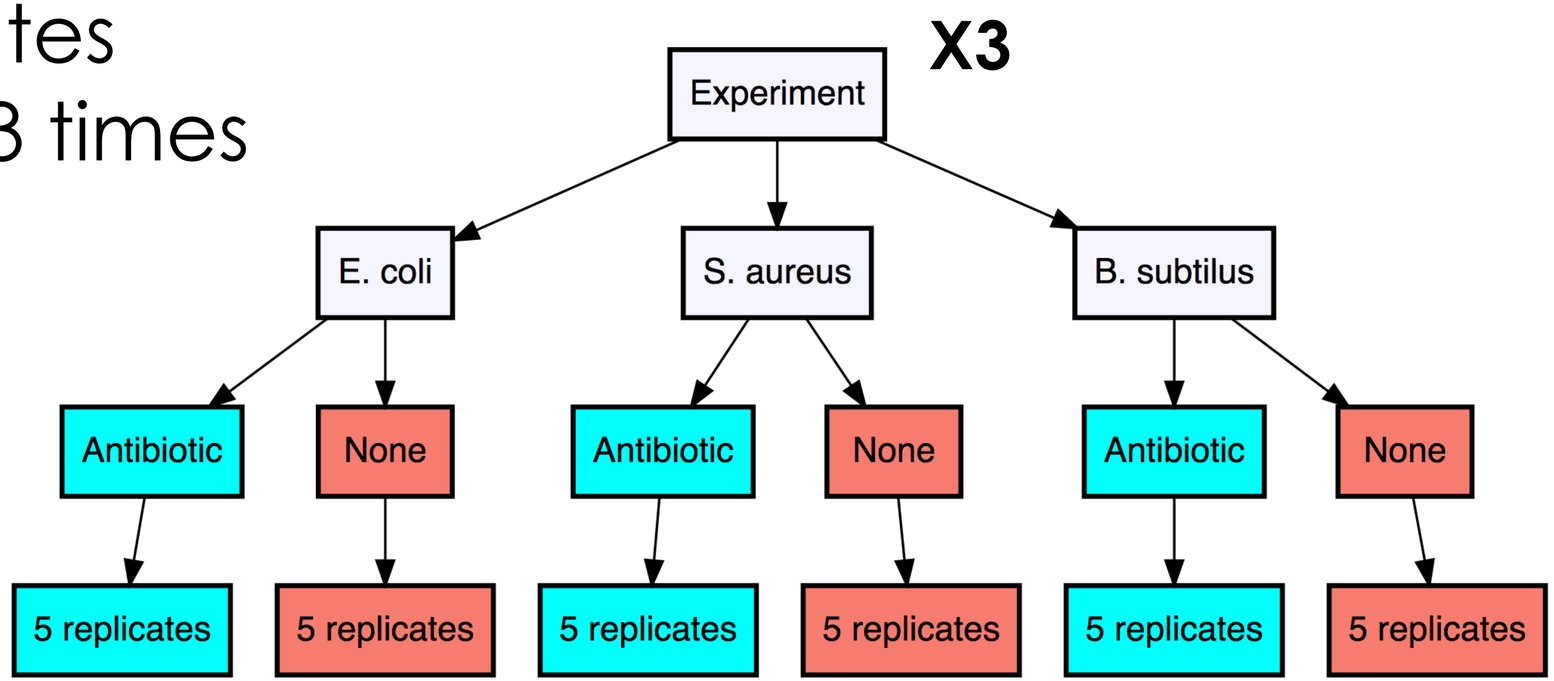


Sample Labels

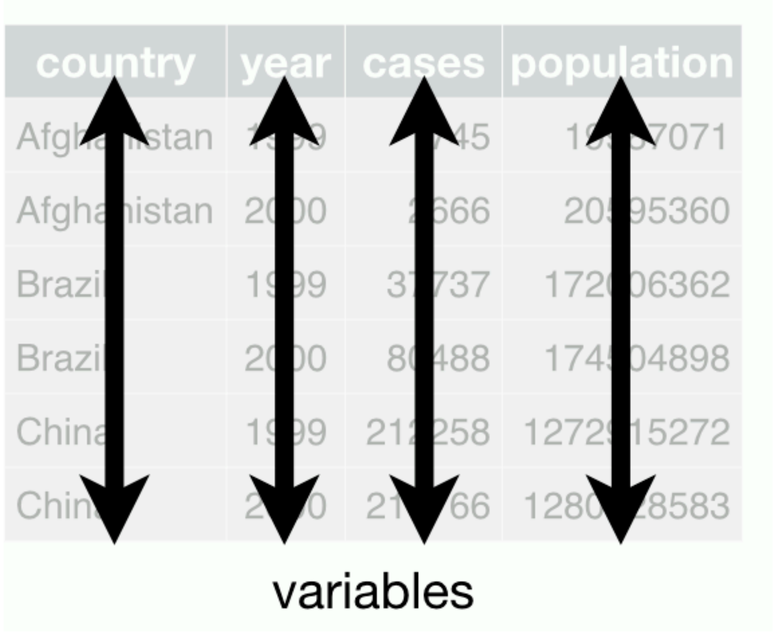
Treatment _Sample_ TechRep_BioRep_Antibody

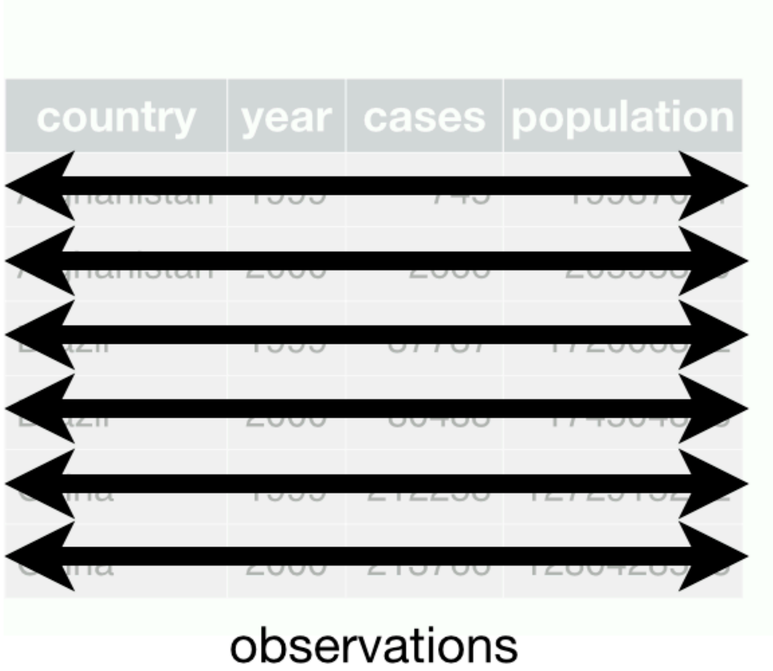
Experimental Design

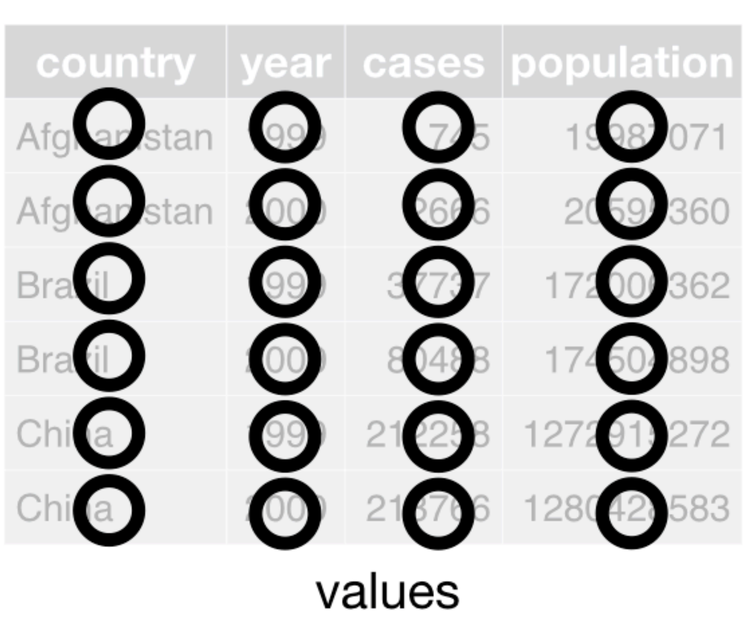
Three organisms – *E. coli, S. aureus, B. subtilus*
Two treatments - Antibiotic, None
Experiment - 5 replicates
Repeat experiment - 3 times



Tidy data


country
year
cases
population
variables


country
year
cases
population
observations


country
year
cases
population
values

Wide format

Treatment	1_Ecoli	1_Saureus	1_Bsubtilis	2_Ecoli	2_Saureus	2_Bsubtilis
Antibiotic	285	240	312	362	244	415
Antibiotic	345	371	461	368	375	315
Antibiotic	298	337	352	287	228	370
Antibiotic	286	394	494	378	302	314
Antibiotic	354	213	311	363	349	303
None	146	286	340	228	284	363
None	180	300	285	246	262	381
None	137	279	271	166	266	325
None	179	253	355	226	270	398
None	168	272	424	175	258	336

Long format - (tidy)

Treatment	Experiment	Organism	Count
Antibiotic	1	Ecoli	285
Antibiotic	1	Ecoli	345
Antibiotic	1	Ecoli	298
Antibiotic	1	Ecoli	286
Antibiotic	1	Ecoli	354
None	1	Ecoli	146
None	1	Ecoli	180
None	1	Ecoli	137
None	1	Ecoli	179
None	1	Ecoli	168

group_by() %>% summarize()

Treatment	Experiment	Organism	Count
Antibiotic	1	Ecoli	285
Antibiotic	1	Ecoli	345
Antibiotic	1	Ecoli	298
Antibiotic	1	Ecoli	286
Antibiotic	1	Ecoli	354
None	1	Ecoli	146
None	1	Ecoli	180
None	1	Ecoli	137
None	1	Ecoli	179
None	1	Ecoli	168

Organism	Treatment	Experiment	N	mean	sd	se
Ecoli	Antibiotic	1	5	313.6	33.32116445	14.90167776
Ecoli	Antibiotic	2	5	351.6	36.66469692	16.39695094
Ecoli	Antibiotic	3	5	346.2	44.80736547	20.03846301
Ecoli	None	1	5	162	19.55760722	8.746427842
Ecoli	None	2	5	208.2	35.42880184	15.84424186
Ecoli	None	3	5	177.6	40.14722905	17.95438665

Demo!

Statistical tests

shapiro.test() - normal distribution

t.test() - T test

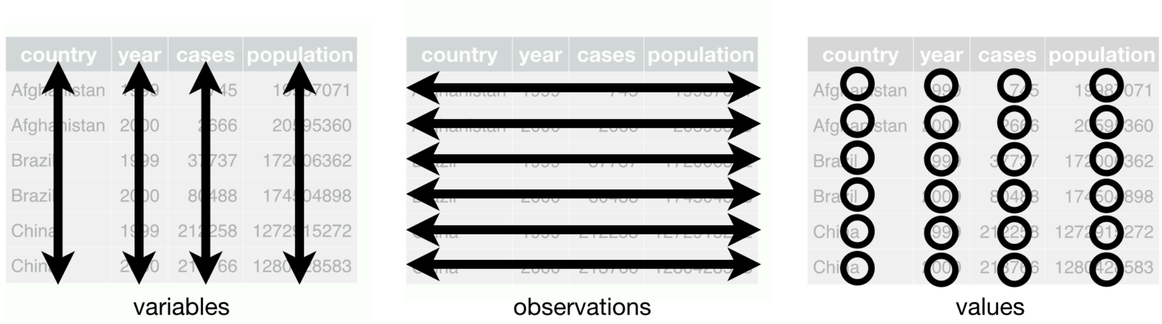
aov() - ANOVA

TukeyHSD() - Tukey post hoc test

wilcox.test() - Mann Whitney U test

kruskal.test() - Kruskal Wallis test

Much easier to run on 'tidy data'



Shapiro Test - normality test

Need all values in a single column

Tests the null hypothesis that data is normally distributed

Formula: shapiro.test(dataframe$column)

```
normality <- shapiro.test(data3$Count)
```

Student's t-Test : compare differences between means

Formula: **t.test(**y**~**x, **data=** dataframe**)** # where y is numeric and x is a binary factor

Formula: **t.test(**y1, y2, **data=** dataframe**)** # where y1 and y2 are numeric

Formula: **t.test(**y1, y2, **paired =** TRUE**)** # where y1 and y2 are numeric

tt <- **t.test(**mpg~am, data = mtcars**)**

Analysis of Variance (ANOVA)

Formula: **aov(numerical~factor***factor2*factor3,
data = dataframe**)**

ANOVA <- **aov(**mean**~**Organism*****Treatment, **data =** data4**)**

Tukey's HSD - post hoc test

Formula: **TukeyHSD(**aov_output**)**

```
ANOVA <- aov(mean~Organism*Treatment,
             data = data4)

TukeyHSD(ANOVA)
```

Wilcoxon/Mann-Whitney U test

Formula: `wilcox.test(y~A)` # where y is numeric, A is binary factor

Formula: `wilcox.test(y, x)` # where x and y are numeric

Formula: `wilcox.test(y1, y2)` # where y1 and y2 are numeric

Kruskal Wallis test

Non-parametric test

one way ANOVA by ranks

Formula: `kruskal.test(y~A)` # where y is numeric and A is a factor

library(broom) - tidy up statistical tests

Formula: tidy(statisical_analysis)

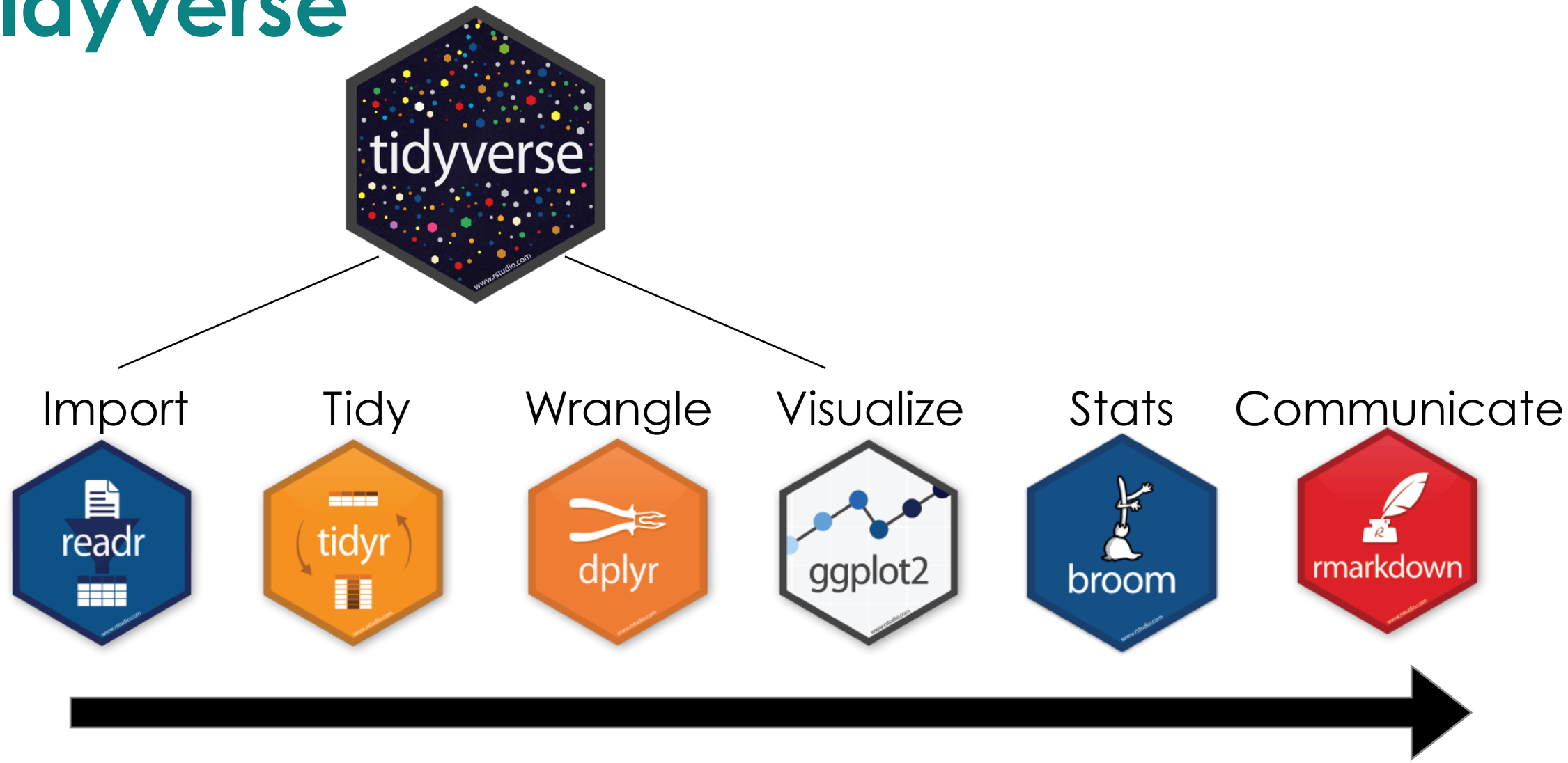
```
        Welch Two Sample t-test

data:  mpg by cyl
t = 4.7191, df = 12.956, p-value = 0.0004048
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  3.751376 10.090182
sample estimates:
mean in group 4 mean in group 6
       26.66364        19.74286
```


```
> tidy(ttest)
  estimate estimate1 estimate2 statistic      p.value parameter conf.low conf.high                 method alternative
1 6.920779  26.66364  19.74286  4.719059 0.0004048495  12.95598 3.751376  10.09018 Welch Two Sample t-test   two.sided
```

Demo!

Data Analysis in the **Tidyverse**

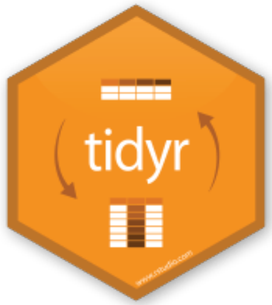


Data Analysis in the **Tidyverse**

Import

read_csv()
write_csv()

Tidy



gather()
spread()
separate()
unite()

Wrangle

filter()
rename()
select()
mutate()
group_by()
summarise()

Visualize

ggplot()
geom_bar()
geom_point()
geom_boxplot()
geom_hist()
geom_violin()
ggsave()

Stats

t.test()
aov()
TukeyHSD()
tidy()

Communicate

.md
.Rmd
.pdf
.html

Thank you!

library(**pwr**) – power analysis

Need three to calculate the fourth

- **sample size = n**
- **effect size = d**
- **significance level (P value) = sig.level**
- **power 1-P = power**

(ANOVA)

number of groups = k

effect size = f (0.1, 0.25, 0.4)

pwr.anova.test(k = ,n = ,f = , sig.level = , power =)

pwr.t.test(n = , d = , sig.level =, power = ,
type = c("two.sample",
"one.sample",
"paired"))

Markdown

- Lightweight markup language
- Easy formatting
- Easy to read
- Simple syntax

```
**Bold text**
*italics*
Plain text

#Big Header
##Smaller Header
###Smaller
####Even maller

Easily create lists

- item one
- item two
- item three

It's really easy to make tables

|header|header|header|
|---|---|---|
|value1|value2|value3|
```

Bold text

italics

Plain text

Big Header

Smaller Header

Smaller

Even maller

Easily create lists

- item one
- item two
- item three

It's really easy to make tables

header	header	header
value1	value2	value3

Rmarkdown

- Markdown
- LaTeX
- R code
- renders to .md, .pdf, .html
- Great for formatting dissertation

```
---
title: "Untitled"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```{r cars}
summary(cars)
```
```

```
     speed           dist       
 Min.   : 4.0   Min.   :  2.00  
 1st Qu.:12.0   1st Qu.: 26.00  
 Median :15.0   Median : 36.00  
 Mean   :15.4   Mean   : 42.98  
 3rd Qu.:19.0   3rd Qu.: 56.00  
 Max.   :25.0   Max.   :120.00  
```