Agricultural Data Analysis & Visualization in R

Lily Northcutt
Data Scientist

Chile Pepper Conference 2026
🌶️ New Mexico
February 5, 2026

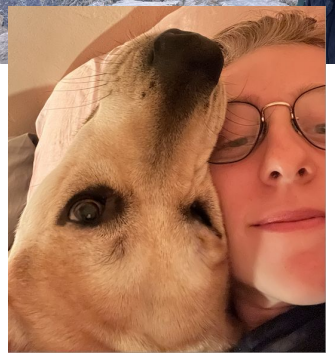
Hi 👋, I’m Lily

I’m a data scientist with an academic background in Mathematics and Machine Learning.

I’ve worked with NMSU’s Chile Breeding Program & CPI, as well as NMSU’s Entomology, Plant Pathology, and Weed Science group.

I primarily work in python now but I started programming with R and I still love the language.

When I’m not working can find me trying to pet dogs on a hike, singing karaoke (poorly), learning to knit, and trying out new restaurants!



Working With Data

Part 1

1. Setting up the environment
2. Reading in data
3. Tidy and Transform our data
4. Summarize the data

- This is an **interactive** and **hands-on** workshop!
- Not a lecture - be vocal, ask questions
- Need help?
    - Your Resources: Talk to your neighbor, **google**, ask me, R for Data Science Book, R Cheatsheets
- **There are stickers**

This slide means you have work to do!

Your Turn

(practice run)

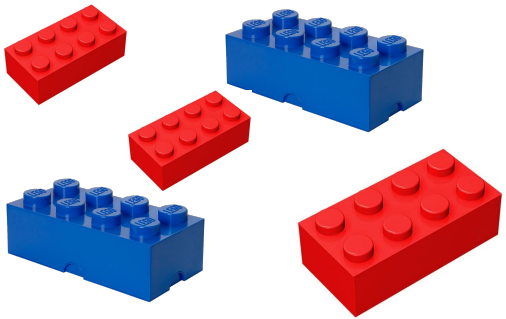
Introduce yourself to your neighbors

Your Turn

My Goal

My goal is to introduce some of (what I consider to be) the **essential** building blocks for working in R.

Through practice and creativity, you will be able to use the blocks to make something spectacular



What R provides

you

What you will build

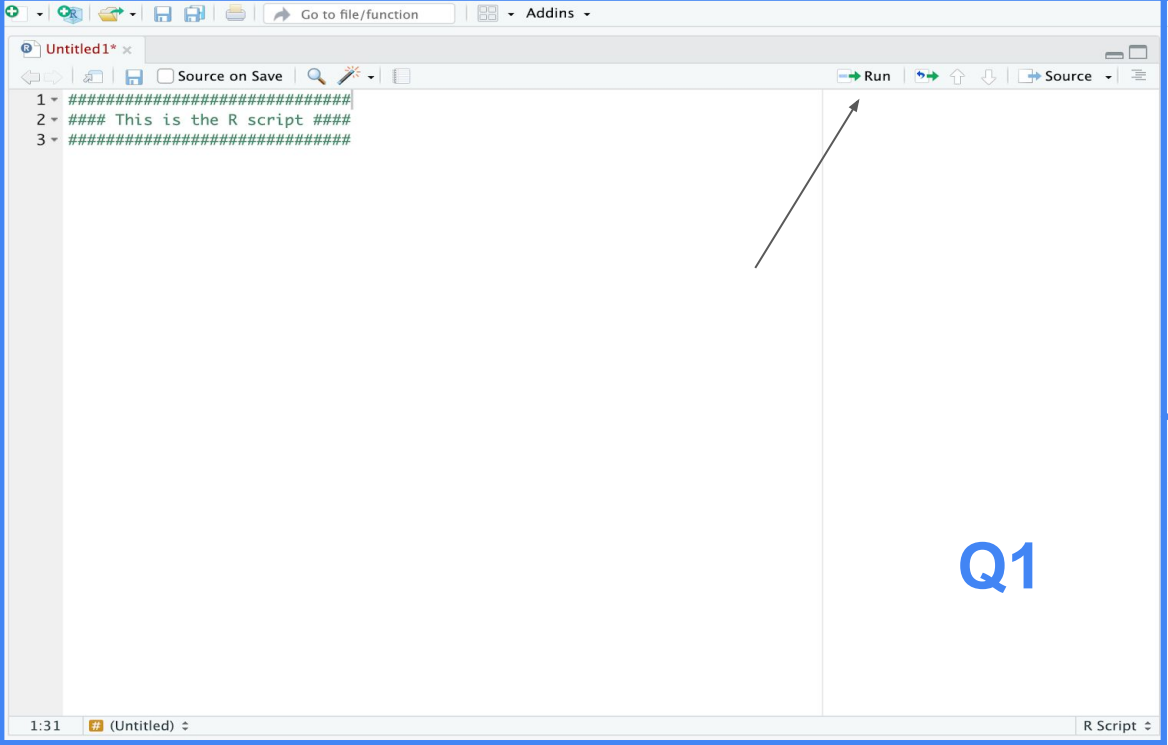
Section A: Intro

R is a programming language ……

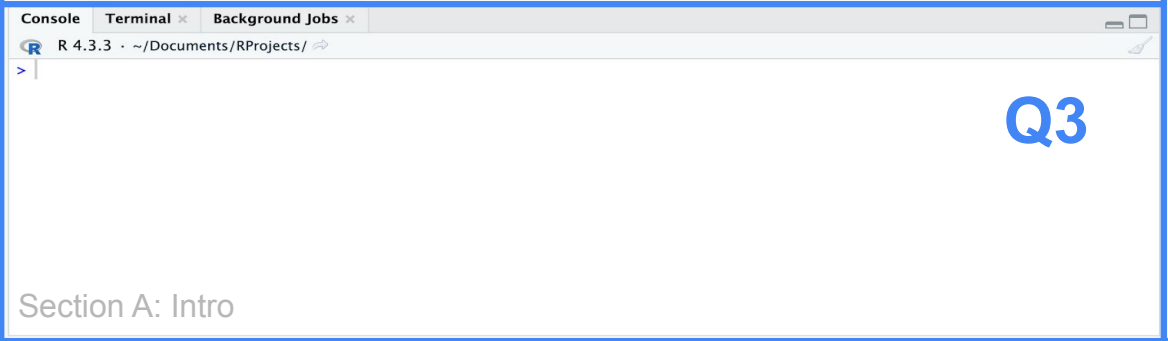
- Free, open-source programming language for statistics
- Widely used in agricultural research, data analysis, modeling
- Great for reproducibility - your code is your methods section!

RStudio is an Integrated Development Environment (IDE) ……

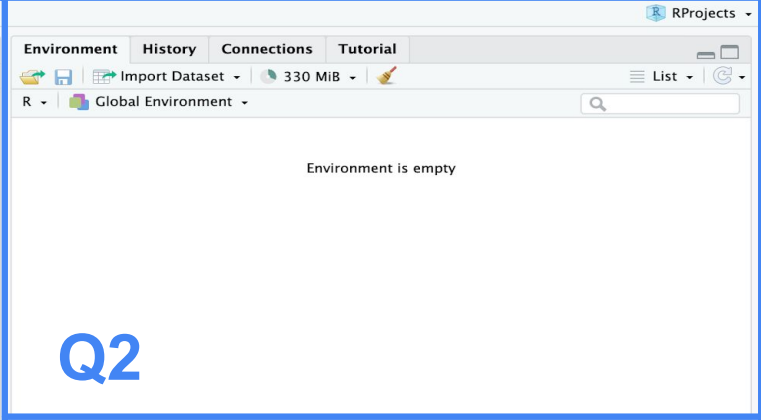
- The user-friendly interface to using R!


Go to file/function
Addins
Untitled1*
Source on Save
Run
Source
1 ##############################
2 #### This is the R script ####
3 ##############################
1:31
(Untitled)
R Script

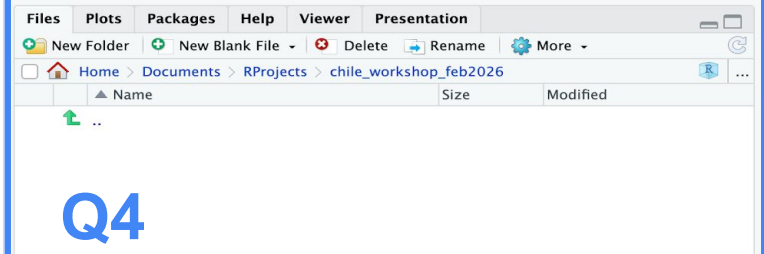
Q1


Console
Terminal
Background Jobs
R 4.3.3 · ~/Documents/RProjects/
>

Q3


RProjects
Environment
History
Connections
Tutorial
Import Dataset
330 MiB
List
R
Global Environment
Environment is empty

Q2


Files
Plots
Packages
Help
Viewer
Presentation
New Folder
New Blank File
Delete
Rename
More
Home
Documents
RProjects
chile_workshop_feb2026
Name
Size
Modified
..

Q4

Quick Demo: Ways to run code in RStudio

1. Console code
   a. `Type 2 + 2 in console → Enter`
2. Create a new .R file
   a. `File → New File → R Script`
   b. `Type 2 + 2 in script, highlight, click "Run" (or Ctrl/Cmd + Enter)`
   c. `Now press Source`

What shows up in the console?

Scripts are saved and reproducible, console is typically for a quick line of code!

Section B: Project Set-Up

Github Repo 🌶️

github.com/lilynorthcutt/chile_workshop_feb2026

Step 1: Navigate to my personal github repo for this workshop

(use link above or use QR or use bit.ly link)

Step 2: Go into the `working_with_data` folder and find the `problems.R` file

bit.ly/4kgUqqV

Quick Demo: Project Set-Up

1. Create a project
    a. `File → New Project → New Directory → New Project`
2. (Install your Packages)
    a. `To install the "dplyr" package you would run:`
        i. `install.packages("dplyr")`
3. Open an .R file

What kind of data can **R** use?

- Built-in datasets & Downloaded Package datasets
- CSV files (.csv)
- Excel files (.xlsx, .xls)
- Text files (.txt)
- Database connections (SQL)
- Web data (APIs, web scraping)

TODAY: We're using package datasets (`agridat`)

Follow Along Demo in RStudio

1. Load the Package

2. Explore the `agridat` Documentation in the Help Tab

```
# TODO: Follow along reading in data:
# 1. Load Required Package
library("agridat")

# Explore the agridat documentation
?agridat
```

Your Turn

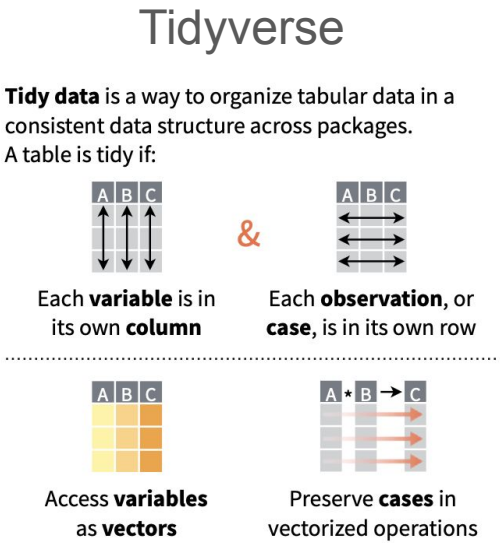
Follow Along Demo in RStudio

1. Read in the yates.oats data
2. Check out the quick ways to view our data

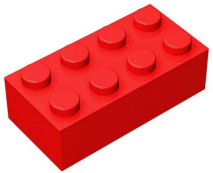
Q: If you were a researcher and this was your experiment, what questions would you want to know?

Your Turn

Section C: Working with Data Basics



Accessing Columns and Basic Functions

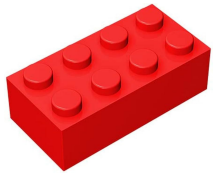


1. Access/Select our Columns

Concept: Data frames are like spreadsheets with named columns

Two ways to access columns:

- dollar sign: `oats$yield`
- bracket notation: `oats[, "yield"]`



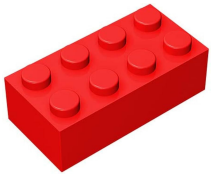
2. Basic Functions

- `mean()`
- `median()`
- `sum()`
- `sd()` - standard deviation
- `length()` - count of values

Demo in RStudio

PROBLEM 1 in the code

Your Turn 20



filter()

Concept: Selects specific ROWS based on the conditions you set

In order to specify your conditions, you will use:

Comparison Operators

- == means "equals" (not =)
- ! means NOT
- > < >= <= for comparisons

Combining Operators

- & means "and"
- | means "or"

```
(nitro > 0.2) & (gen == "Marvelous")
```

Translates to:

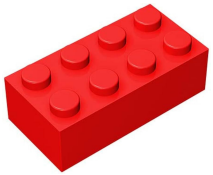
```
gen != "Marvelous" &
(grain > 30 | straw > 30)
```

Translates to:

Demo in RStudio

PROBLEM 2 in the code

Your Turn



mutate()

Concept: Creates NEW columns or modifies existing ones in your dataset

- We will use our previous blocks we acquired inside of mutate to build something bigger and better

case_when()

Concept: Similar to if/else statement that takes many conditions.

```
case_when(
        condition1 ~ "Category 1",
        condition2 ~ "Category 2",
        condition3 ~ "Category 3",
        TRUE ~ "Everything else"
    )
```

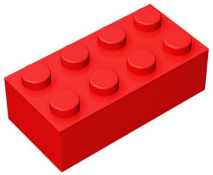
```
case_when(
      nitro == 0 ~ “Control”,
      nitro <=0.2 ~ “Low”,
      nitro <= 0.4 ~ “Mid”,
      TRUE ~ “High”
)
```

Translate to:

Demo in RStudio

PROBLEM 3 in the code

Your Turn 24



`merge()`

Concept: Combining datasets by matching rows on common column(s)

Think of it like adding additional information from a different table based on something in common - will commonly be metadata

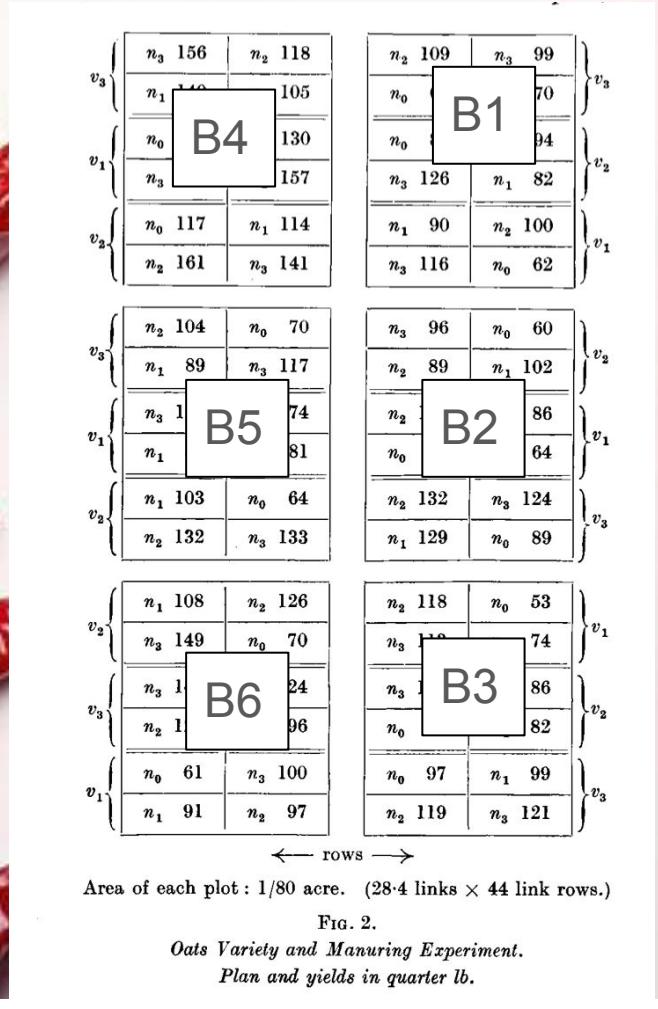
Example: We may merge a key with full variety or treatment names into a large datasets from a trial

Important! *The matching column must have THE SAME NAME in both datasets*

`data.frame()`

Instead of only reading in data, we can always custom create it in R

Demo in RStudio



Area of each plot : 1/80 acre. (28·4 links × 44 link rows.)

FIG. 2.

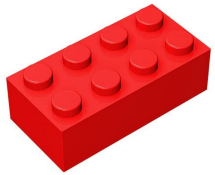
Oats Variety and Manuring Experiment.
Plan and yields in quarter lb.

Graphic: https://academic.oup.com/jrsssb/article/2/2/181/7026418?login=false

PROBLEM 4 in the code

Your Turn

Section D: Summarizing Data



`summarize()`

Concept: Collapses your dataset down to whatever type of summary you specify

This will commonly be things like mean(), min(), max(), nrow()

`group_by()`

`summarize()` really SHINES when used with `group_by()`

Concept: Calculate summaries FOR EACH GROUP instead of the whole dataset

Demo in RStudio

PROBLEM 5

Your Turn

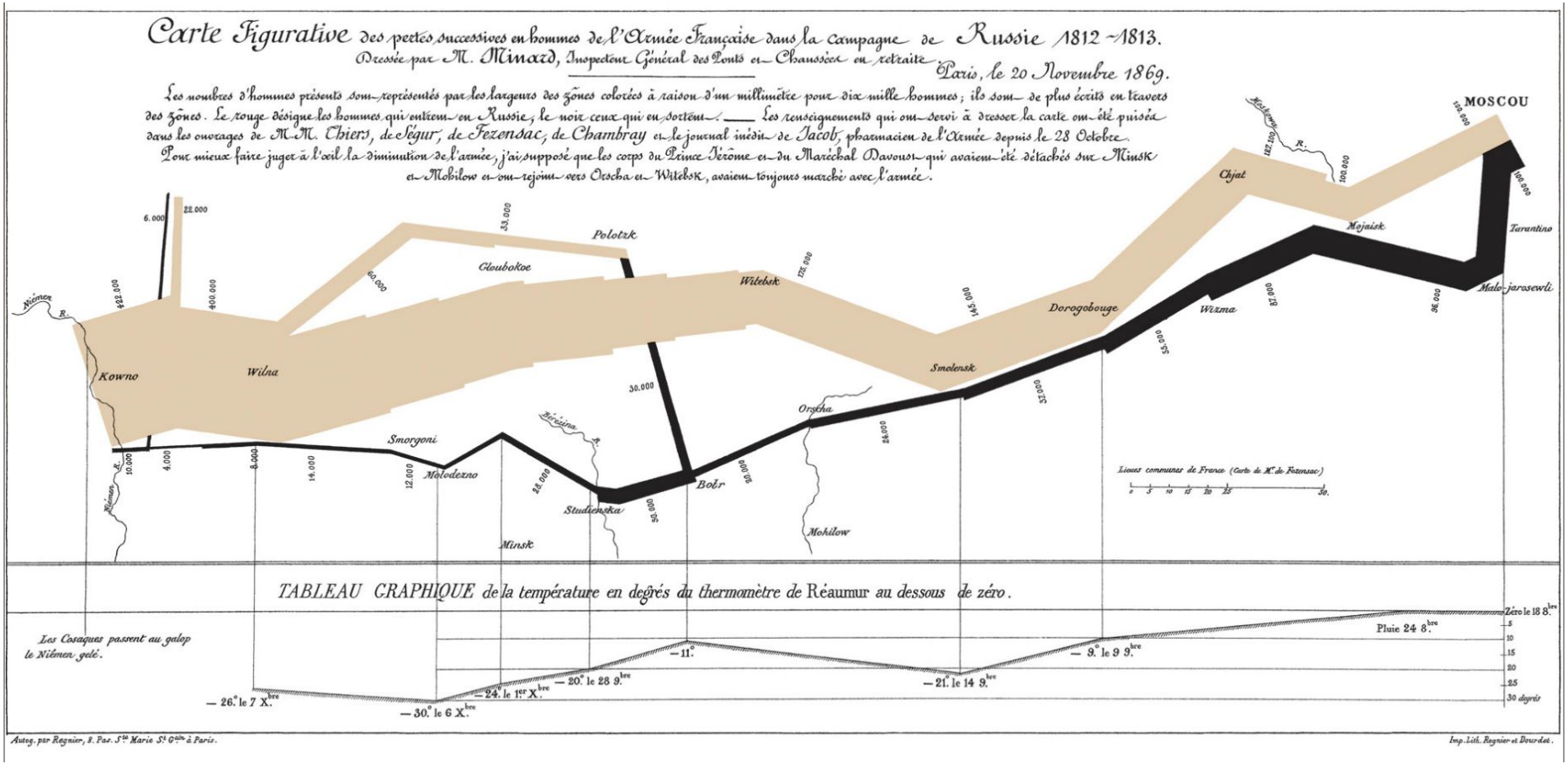
BONUS CHALLENGE!

Your Turn 30

Exploring & Visualizing Data

Part 2

1. Using %>% operator
2. Intro to ggplot2
    a. Scatter plots + Customizing
    b. Other Plots + Error Bars
    c. Faceting

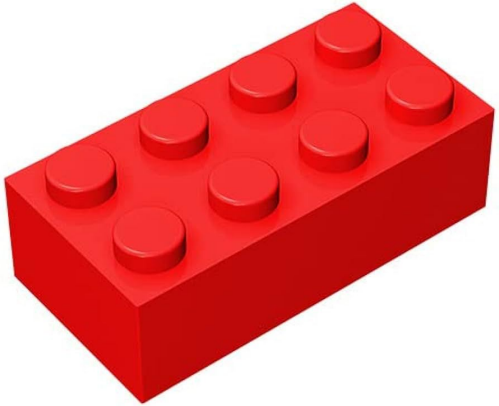


Map by Charles Joseph Minard

https://ageofrevolution.org/200-object/flow-map-of-napoleons-invasion-of-russia/

- This is an **interactive** and **hands-on** workshop!
- Will be faster paced than last section
- Need Help?
    - Your Resources: Talk to your neighbor, google, ask me, R for Data Science Book, R Cheatsheets
- There are still stickers!

Section A: Pipe Operator %>%



Pipe operator: %>%

Problem: Nested code is hard to read, constantly assigning new variables is not sustainable

Solution: %>%

Concept: A way to chain operations together so code reads like a sentence

"... and then …"

Basic Syntax:

```
data %>%
  function1() %>%
  function2() %>%
  function3()
```

Demo in RStudio

(There is another pipe: |> it is virtually the same)

PROBLEM 1

Your Turn

Section B: Intro to `ggplot2`

https://ggplot2-book.org/facet.html

https://rstudio.github.io/cheatsheets/data-visualization.pdf

`ggplot2` Building Beautiful Plots

What is `ggplot2`?

Library of for creating graphics based on "The Grammar of Graphics"

We build plots in 3 main layers:

1. Data
   a. What data are you plotting?
2. Aesthetics (aes)
   a. x-axis & y-axis
   b. shape
   c. color
   d. size
   e. fill
3. Geoms (how the data will be plotted)
   a. Scatter plot `geom_point()`
   b. Boxplot `geom_boxplot()`
   c. Barchart `geom_bar()`
   d. Heatmap `geom_tile()`

Unfortunately I can’t go through every plot…..

But I can show the iterative process for improving one single plot, which can be applied to all other ggplots!

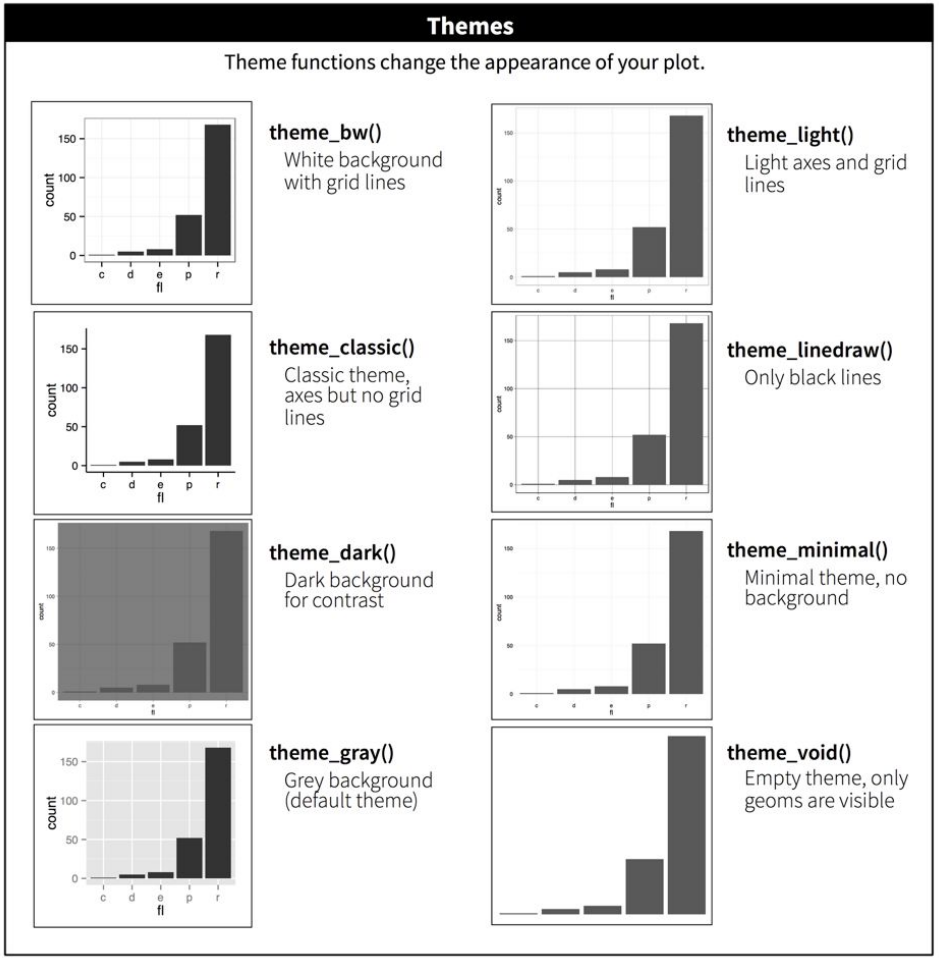
We will take 1 plot and improve it over multiple iterations

Demo in RStudio

PROBLEM 2

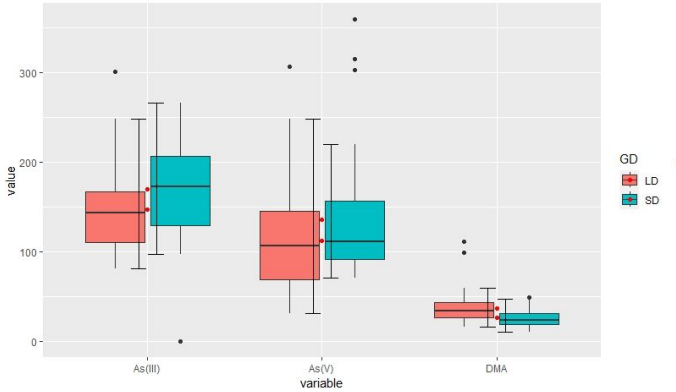
Your Turn

Themes

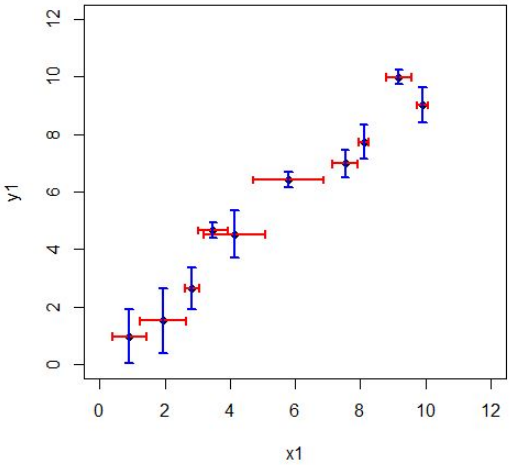

Themes
Theme functions change the appearance of your plot.
theme_bw()
White background with grid lines
theme_light()
Light axes and grid lines
theme_classic()
Classic theme, axes but no grid lines
theme_linedraw()
Only black lines
theme_dark()
Dark background for contrast
theme_minimal()
Minimal theme, no background
theme_gray()
Grey background (default theme)
theme_void()
Empty theme, only geoms are visible
count
fl
c
d
e
p
r
150
100
50
0

Section C: More Plots

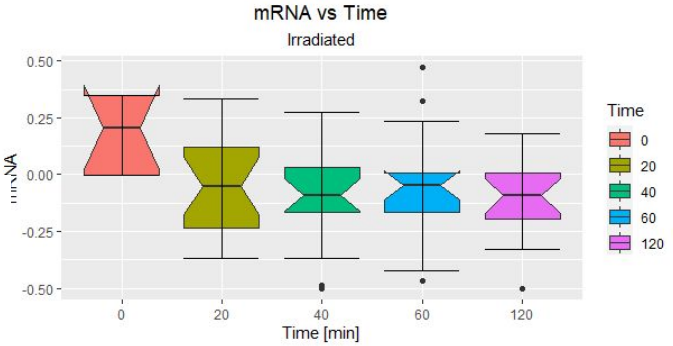
Boxplots and Error Bars (examples)



https://forum.posit.co/t/adding-error-bar-and-mean-value-in-box-plot-with-multiple-variables/174908



https://chitchatr.wordpress.com/2013/06/25/add-error-bars-to-a-plot-in-r/



https://stackoverflow.com/questions/63860118/why-doesnt-ggplot-show-the-error-bar-of-a-boxplot

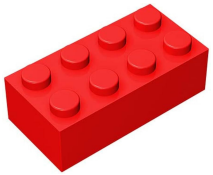
https://www.sthda.com/english/wiki/ggplot2-violin-plot-quick-start-guide-r-software-and-data-visualization#google_vignette

Demo in RStudio

PROBLEM 3

Your Turn

Section D: Faceting

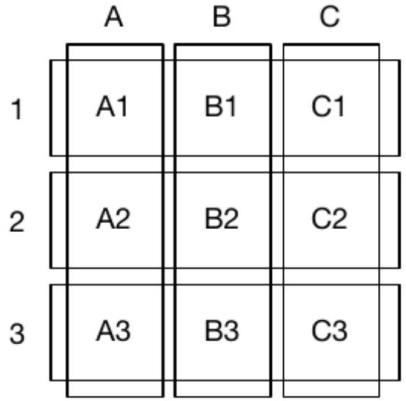


Faceting ❤️

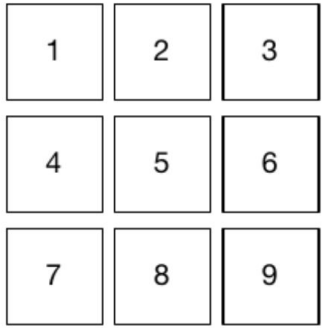
Concept: Plot subplots by the group(s) of your choosing

Why it's so great:

- Separate panels for each group
- Compare patterns side-by-side
- Professional, publication-quality



facet_grid



facet_wrap

There are three types of faceting:

- `facet_null()`: a single plot, the default.
- `facet_wrap()`: "wraps" a 1d ribbon of panels into 2d.
- `facet_grid()`: produces a 2d grid of panels defined by variables which form the rows and columns.

Demo in RStudio

https://ggplot2-book.org/facet.html

PROBLEM 4

Your Turn

Bringing it all together!

Question 1:
Did any locations in the field have higher yields?

Question 2:
What's the best variety x nitrogen combination?

Lily’s 2 cents

1. Use .Rdata - super helpful and can save time
2. Never use a pie chart
3. Marker Understanding: Color > Size > Shape
4. Good figures + graphics can take as much time as the analysis
5. Take time to explain