

Data Visualization with R

Gladstone Institutes

Krishna Choudhary

Bioinformatics Core, GIDB

2020

Outline (contd. on next slide)

- ◆ Introduction
- ◆ Hands-on analysis
 - 1. Explore *Iris* data and save a publication-ready figure.
 - ◆ Data: “iris.csv”
 - ◆ Result: “Iris.pdf”
 - ◆ Script: “1. iris.R”
 - 2. Brief digression to save a heatmap.
 - ◆ Data: “norm_counts.txt”
 - ◆ Results: “Heatmap.pdf” and “Heatmap.tiff”
 - ◆ Script: “5_Heatmap.R”
 - 3. Plot and annotate nucleotide-resolution data.
 - ◆ Data: “nucleotide_resolution.RData”
 - ◆ Result: “Nucleotide_resolution.pdf”
 - ◆ Script: “2. nucleotide_resolution.R”

Outline (contd.)

- 4. Miscellaneous
 - ◆ Interactive data visualization
 - ◆ Visualization options beyond ggplot2
- ◆ Conclusion

R topics covered

- ◆ Hands on
 - ◆ Plotting with ggplot2.
 - ◆ Writing loops.
 - ◆ Saving figures in high resolution.
 - ◆ Plotting heatmaps.
- ◆ Time-permitting
 - ◆ Interactive data visualization with Shiny.

Assumed background

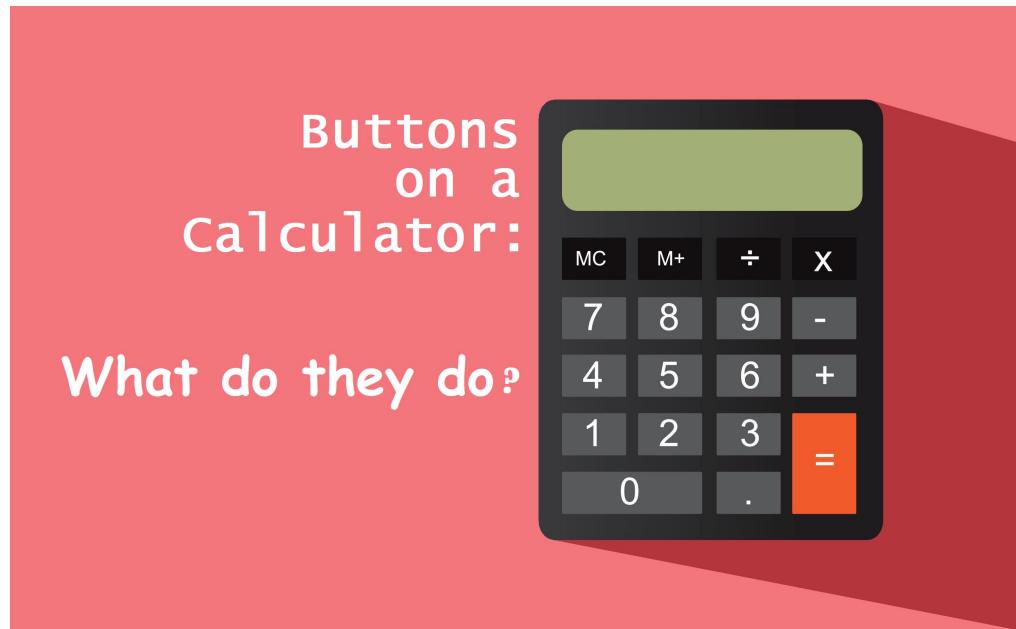
- ◆ Familiarity with R and RStudio.
- ◆ Basic arithmetic operations in R.
- ◆ Variables and assignments.
- ◆ Commenting in scripts.
- ◆ Functions and libraries.
- ◆ Read, subset and explore data.
- ◆ Basic plotting of data.
- ◆ Data structures available in R.
- ◆ Troubleshooting.

Reading resources

- ◆ ggplot2 cheatsheet. (download link in description)
- ◆ *ggplot2: Elegant Graphics for Data Analysis* by Hadley Wickham. (download link in description)
- ◆ *The Grammar of Graphics* by Leland Wilkinson. (download link in description)
- ◆ Material from *Introduction to R* workshop on GitHub.

Refresher

R:= Calculator with more “buttons” than you'll ever use.



- ◆ R can do all that a calculator can.
 - ◆ Open R.
 - ◆ Try $2+3$.
 - ◆ Try $2*3$.
 - ◆ Try $(2+3)/5$.
- ◆ Conclusion: Off with the calculators!

R is a console application.

- ◆ Text only interface. Inputs and outputs can be images.
- ◆ RStudio: Brings more Graphical User Interface (GUI) features to R.
- ◆ RStudio is to R as Microsoft Word is to Notepad in Windows orTextEdit in Mac. (~kinda)

Commenting inside a source script

- ◆ Looking at an old script, or someone else's script can be scary.
- ◆ Speak to machine in code.
- ◆ Leave comments in natural language (e.g., English).
 - ◆ Anything written after # is interpreted by R as a comment for humans.

Empty workspace/environment



Data analysis creates objects that store data.

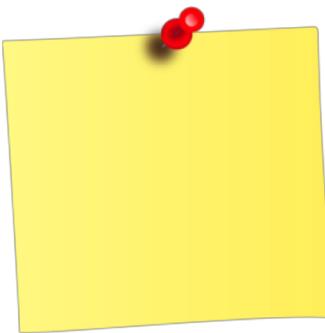
(top-right pane in RStudio)



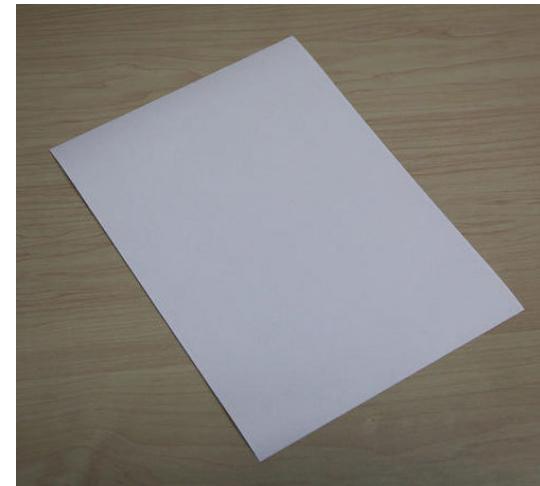
Data structures.

Data structures are data organization/storage formats.

- ♦ Storing data physically? Options:



Sticky note



Printer paper



Task pad

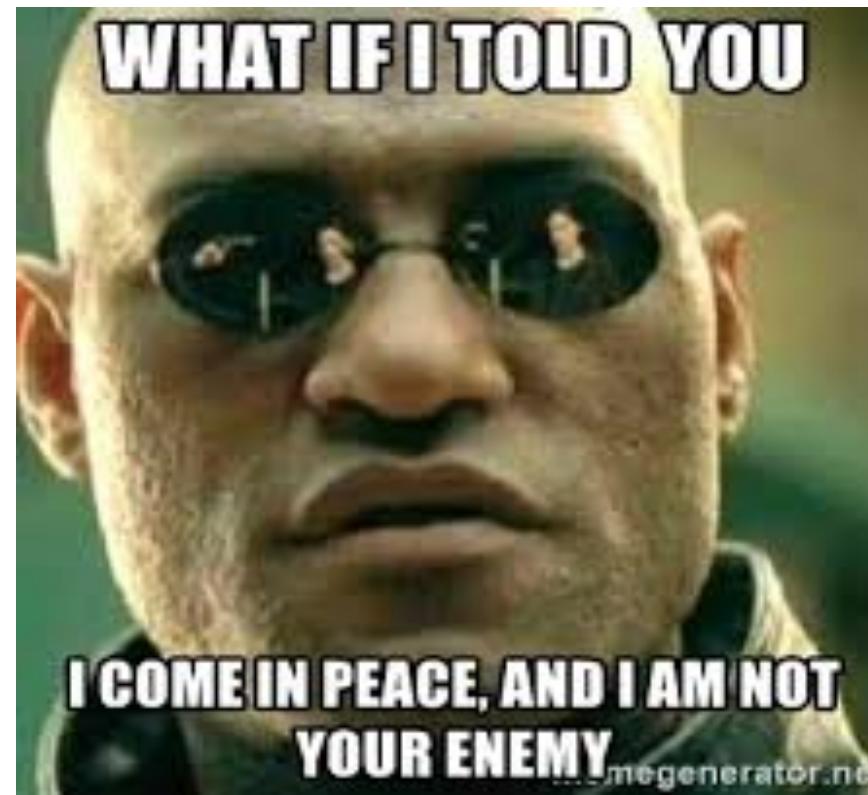


Long scroll

Data structures are data organization/storage formats.

- ◆ Working with data in R? Options:
 - ◆ Numeric vector
 - ◆ `c(2, 5, 10, 11)`
 - ◆ Character vector
 - ◆ `c("apples", "oranges", "butter", "dry yeast")`
 - ◆ Data frames
 - ◆ `dat` from Iris example.
 - ◆ Matrix
 - ◆ Example counts for genes in several samples.
 - ◆ Lists
 - ◆ Flexible data structures.
 - ◆ `mylist <- list(a = c(2, 5, 10, 11),
b = c("apples", "oranges", "butter", "dry yeast"))`

Error!!!



Functions

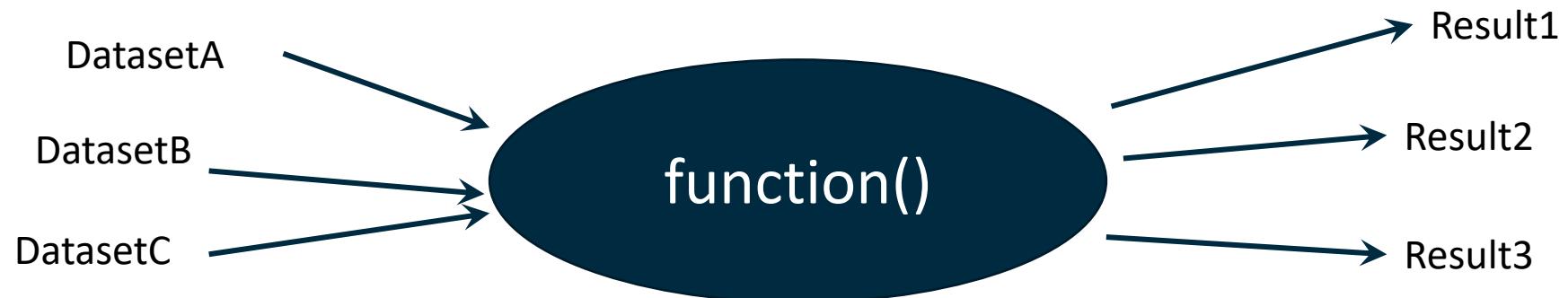
Working with buttons for modern scientific calculations?

- ◆ The best of calculators => 10s of buttons for specific tasks.
- ◆ Modern science needs many times more than 10s.



Functions are to R what buttons are to calculators.

- ◆ R can perform 100s of 1000s of tasks.
- ◆ Tasks are performed by using functions.
 - ◆ Examples: `sum()`, `prod()`, `mean()`, `t.test()`.



There are functions for all sorts of things.

- ◆ Example to read a table saved in a file:
 - ◆ `read.table()`
 - ◆ Usage: `read.table("filename")`



There are functions for all sorts of things.

- ◆ Example to find complementary DNA sequence.
 - ◆ `complement()`
 - ◆ Usage: `complement(sequence)`

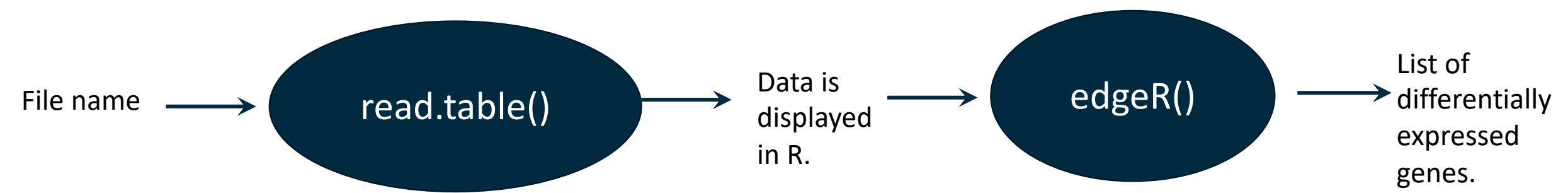


There are functions for all sorts of things.

- ◆ Example to install a package in R.
 - ◆ `install.packages()`
 - ◆ Usage: `install.packages("ggplot2")`



Directing output of one function as input to another.



Library is a collection of functions.

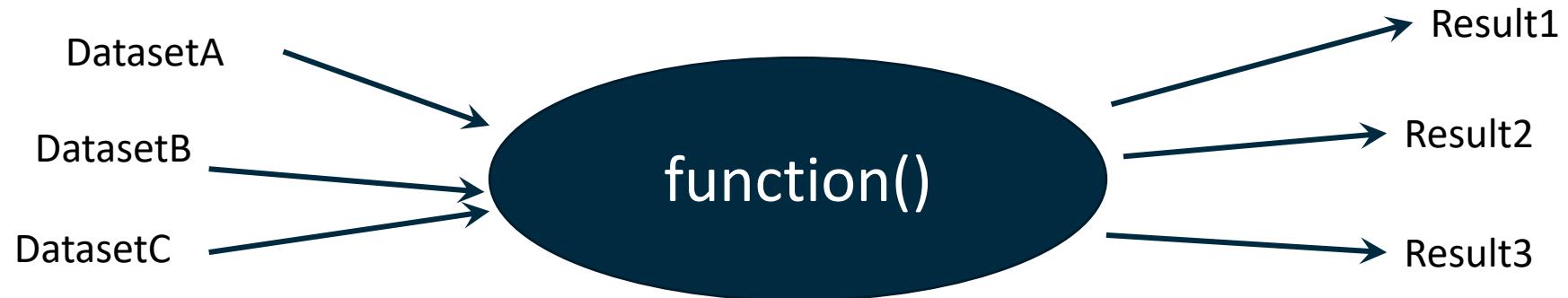
- ◆ R works with > 100000 functions.
- ◆ More being added everyday.
- ◆ If the software were to load (i.e., make available) all of them at startup, it will take a while to launch.
- ◆ Solution: Bundle related functions together in a *library*.
 - ◆ Example: edgeR.
- ◆ Load functions only when needed. (:= Install apps on smartphone as needed)

Functions may need lots of information

- ◆ Calculators may take only numbers.
- ◆ R functions take variety of things. Numbers, characters, etc.
- ◆ Example:
 - ◆ Reading a file. Inputs required: File name and address on computer, formatting of file, type of file, etc.
 - ◆ Writing a file. Inputs required:
 - ◆ what to write?
 - ◆ where to write?
 - ◆ how to format the file?
 - ◆ Include column names, row names?
 - ◆ separate with comma?

ggplot2: Package with functions for plotting.

- ◆ R can perform 100s of 1000s of tasks.
- ◆ Tasks are performed by using functions.
 - ◆ Examples: `sum()`, `prod()`, `mean()`, `t.test()`.



ggplot2: Package with functions for plotting.

- ◆ Plotting layer-by-layer.
- ◆ Function names:
 - ◆ qplot
 - ◆ ggplot



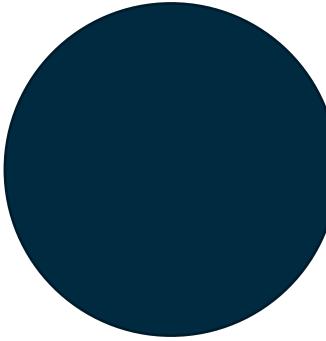
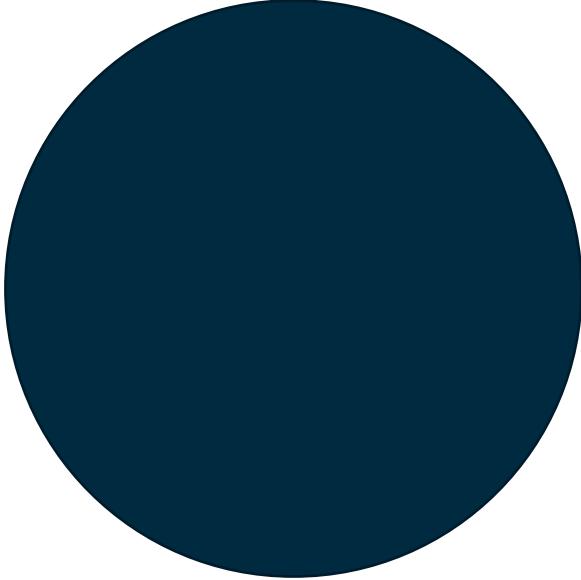
Hands-on: Part 1

Explore *Iris* data and save a publication-ready figure.

- ◆ Data: “iris.csv”
 - ◆ Iris is a plant genus.
 - ◆ Species of this genus may be identified based on the dimensions of petals and sepals of its flowers.
- ◆ Result: “Iris.pdf”
- ◆ Script: “1. iris.R”

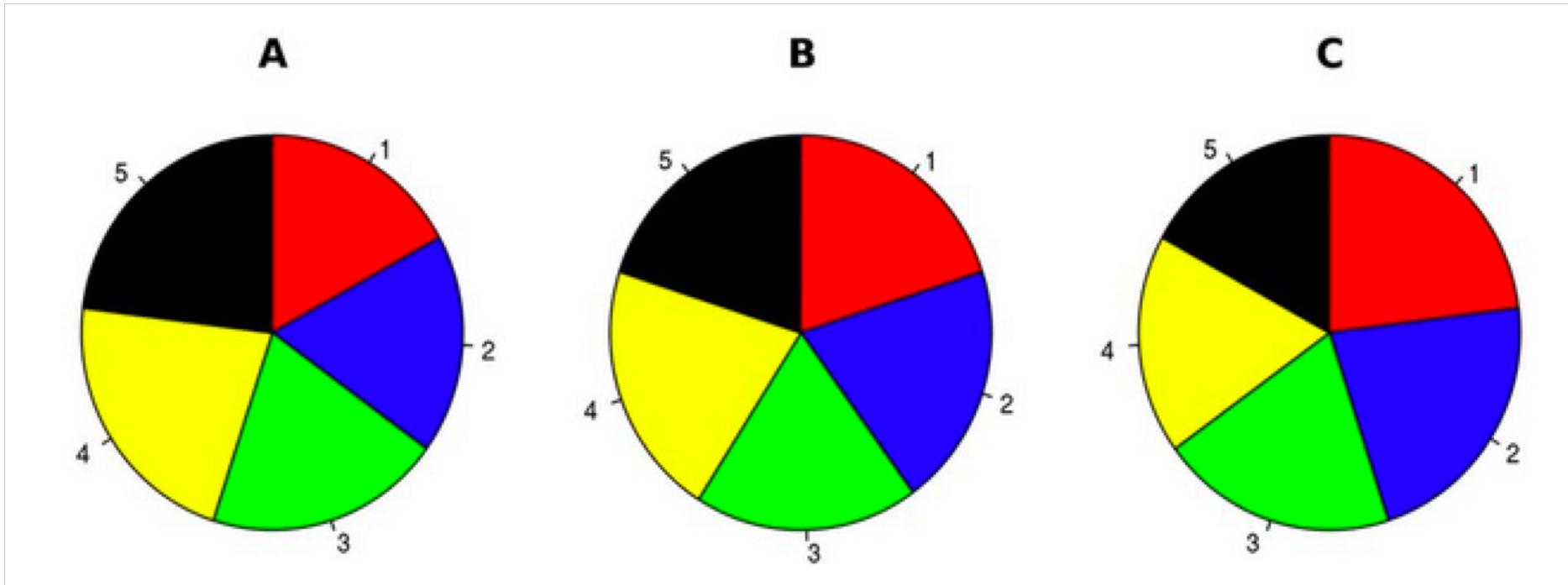
Additional resources

- ◆ Q&A: <https://stackoverflow.com/>
- ◆ More on arranging figures for publications.
 - ◆ <https://cran.r-project.org/web/packages/egg/vignettes/Ecosystem.html>

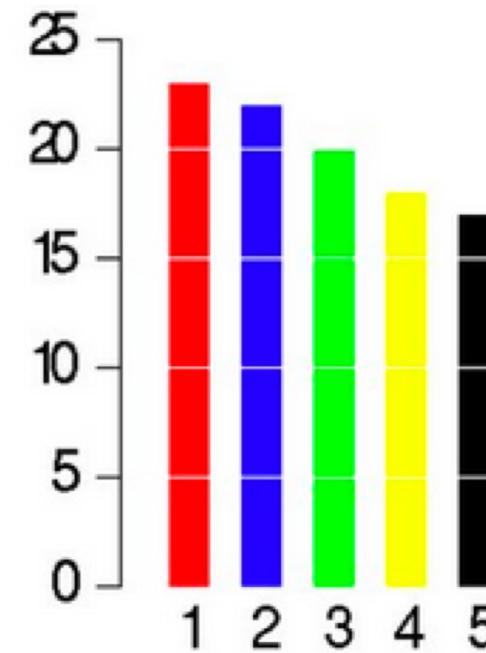
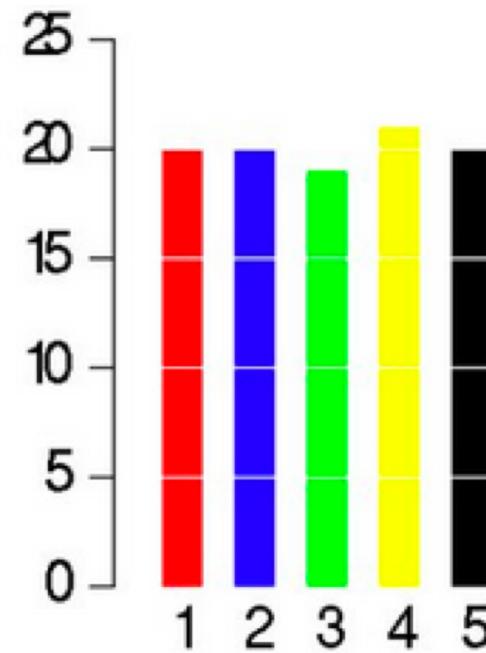
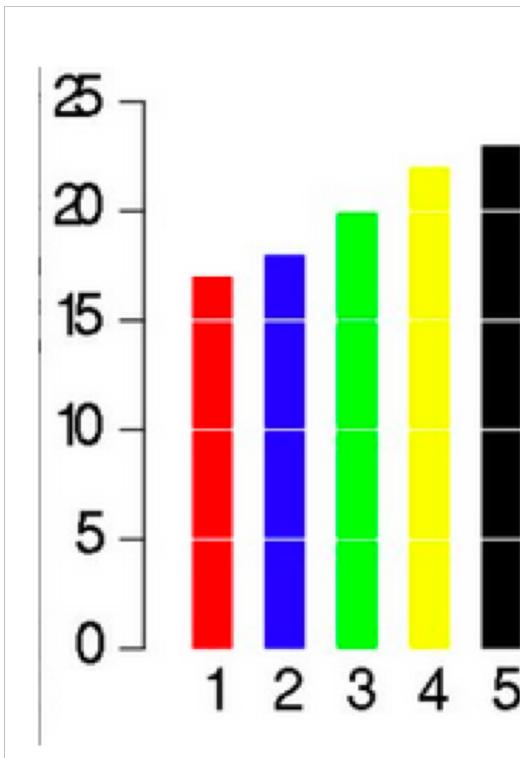


Notes on geometrical
attributes of data.

Which color occupies third largest area?



We are better at comparing lengths than areas.



“The purpose of data visualization is insight, not pictures.”

- ◆ See link in description.

Brief digression to heatmaps

Save a heatmap for *counts* of genes.

- ◆ Data: “norm_counts.txt”
- ◆ Results: “Heatmap.pdf” and “Heatmap.tiff”
- ◆ Script: “5_Heatmap.R”

Hands-on: Part 2

Plot and annotate nucleotide-resolution data.

- ◆ Data: “nucleotide_resolution.RData”
 - ◆ One value for six replicates of each nucleotide of > 10000 genes.
 - ◆ Need to plot specific regions of 10 genes and annotate motifs.
 - ◆ ...
- ◆ Result: “Nucleotide_resolution.pdf”
- ◆ Script: “2. nucleotide_resolution.R”

Hands-on: Part 3

Flip book type animation for characteristics of Hill equation.

- ◆ Data: To generate using Hill equation.
 - ◆ Hill equation gives reaction velocity in terms of substrate concentration.
 - ◆ [https://en.wikipedia.org/wiki/Hill_equation_\(biochemistry\)](https://en.wikipedia.org/wiki/Hill_equation_(biochemistry))
- ◆ Result: “Hill_equation.pdf”
- ◆ Script: “3. Hill_equation_static.R”

Interactive visualization for characteristics of Hill equation using Shiny.

- ◆ Data: To generate using Hill equation.
 - ◆ Hill equation gives reaction velocity in terms of substrate concentration.
 - ◆ [https://en.wikipedia.org/wiki/Hill_equation_\(biochemistry\)](https://en.wikipedia.org/wiki/Hill_equation_(biochemistry))
- ◆ Result: Interactive application is launched.
- ◆ Script: “4. Hill_equation_interactive.R”

Interactive visualization with Shiny.

- ♦ Need to define user interface (UI).
- ♦ Need to serve data/figures to UI and communicate input values.

Beyond ggplot2:

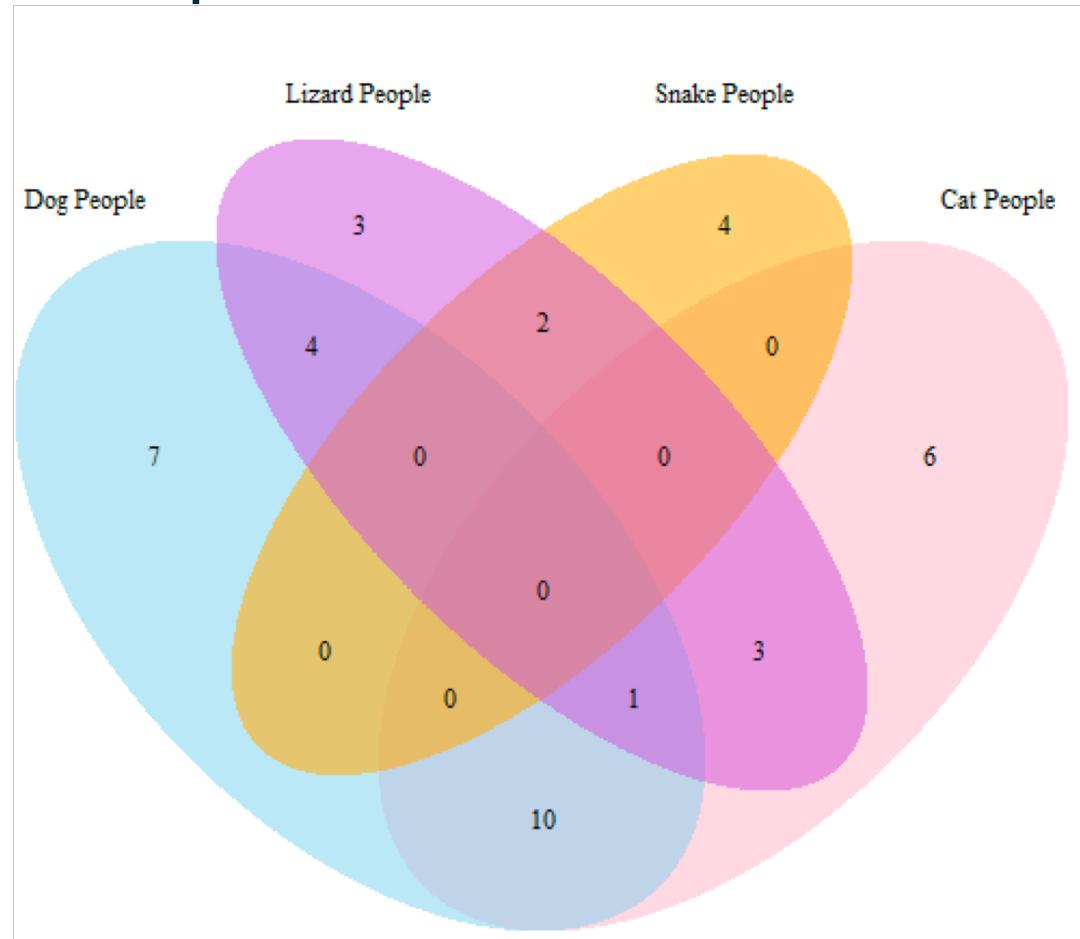
1. Heatmap.

- ◆ Data: “norm_counts.txt”
- ◆ Results: “Heatmap.pdf” and “Heatmap.tiff”
- ◆ Script: “5_Heatmap.R”

Beyond ggplot2:

2. Venn diagram with multiple sets.

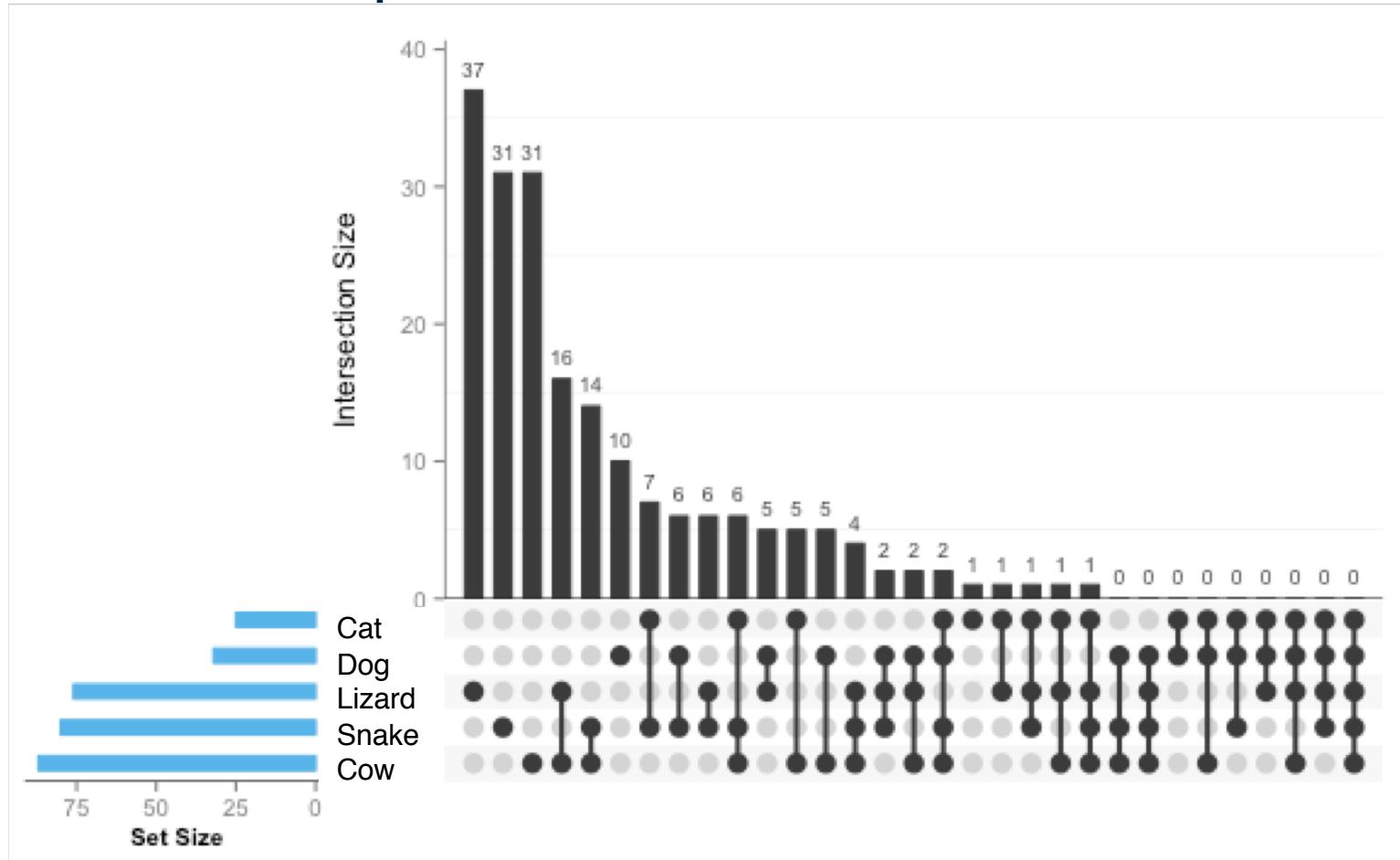
- ◆ See link in description.



Beyond ggplot2:

2. Venn diagrams with multiple sets (alternative).

- ◆ See link in description.



Beyond ggplot2:

3. Select color palettes using RColorBrewer.

- ◆ See link in description.



Your feedback is important to us!

- ◆ <https://bioinformatics-course-feedback.questionpro.com/>
- ◆ ~5 min.

Additional resources

R topics covered

- ◆ Hands on
 - ◆ Plotting with ggplot2.
 - ◆ Writing loops.
 - ◆ Saving figures in high resolution.
 - ◆ Plotting heatmaps.
- ◆ Time-permitting
 - ◆ Interactive data visualization with Shiny.

Examples of high-resolution graphics

- ◆ <https://www.r-graph-gallery.com/>

Interactive visualization with Shiny

- ◆ <https://shiny.rstudio.com/gallery/>

Conclusion

- ◆ Need good grammar for effective communication.
 - ◆ Grammar of Graphics for plotting data.
- ◆ Exploring large-scale data enabled by automated plotting.
 - ◆ Integrate information from multiple sources on same plot.
- ◆ Scripts for high-resolution graphics.
 - ◆ Reproducible research.
- ◆ Fine control over all of plotting area.
 - ◆ Easily change/reproduce figures with alternate themes/annotations.





GLADSTONE INSTITUTES