Introduction to R

Jennifer Latessa
Zhiyuan Yao
Richard Johansen
Nov 26th 2019

University of CINCINNATI

Workshop Agenda

Workshop Expectations
Why R?
Understanding Data
Data Visualizations in R
Work in R Studio

Workshop Agenda

Workshop Expectations

What is R?

Understanding Data

Data Visualizations in R

Work in R Studio

Expectations

- Prerequisites:
  - R and R studio installed
- Goal:
  - Explore R and R studio environment
  - Basic functions of R
  - Import data
  - Download packages
  - Simple modeling
  - Visualize data
  - Export data and visualization
- Caveat
  - This is only an introduction to the R and R studio environment. Learning a new programming language, like a spoken language, takes years of practice. So we cannot cover everything but we will highlight many common features that most of you will use most of the time.

https://www.visumcx.com/single-post/2017/03/02/The-Value-Of-Setting-Customer-Expectations

Workshop Agenda

Workshop Expectations

What is R?

Understanding Data

Data Visualizations in R

Work in R Studio

What is R?

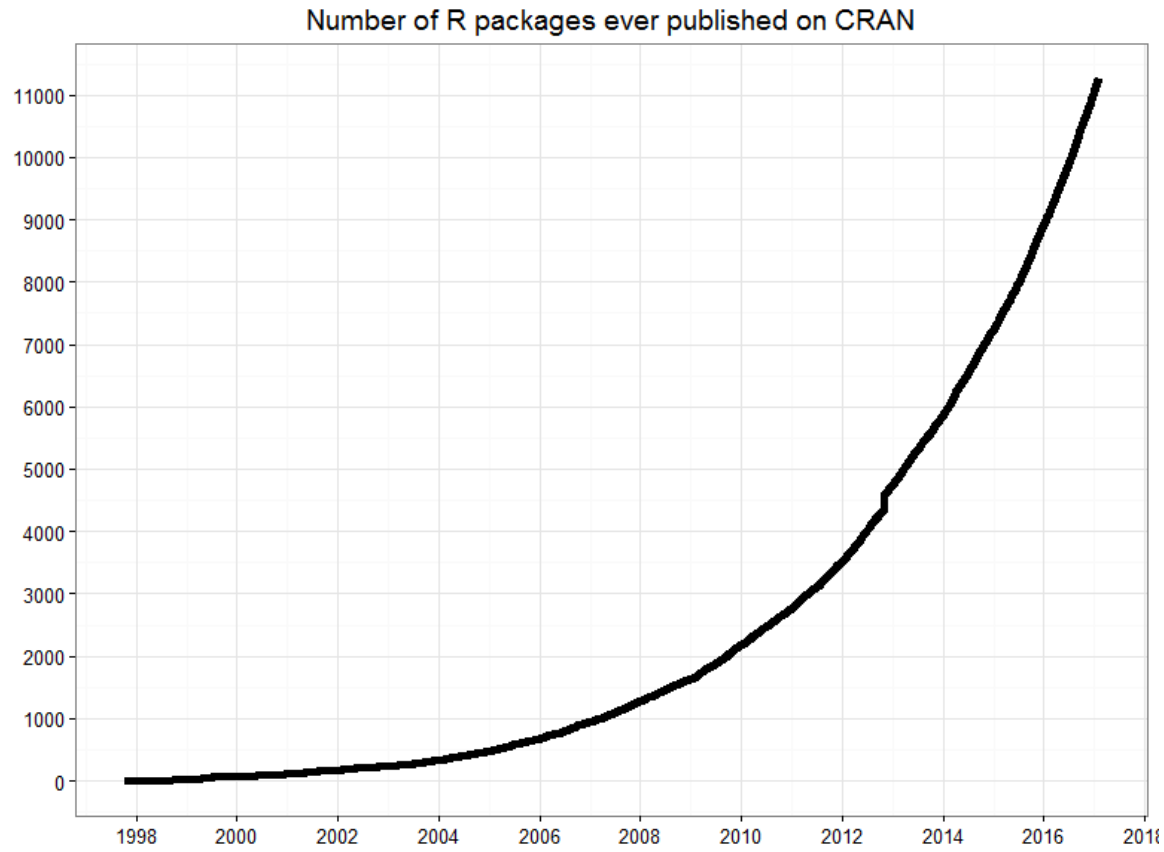
- R is a programming language specifically designed for statistical analysis and graphics
- R is free & open-source used in many academic and industry disciplines
- R Studio is the Interactive User Interface compatible with R
- R and R Studio can be downloaded here:
  - http://cran.r-project.org/
  - https://www.rstudio.com/products/RStudio/#Desktop

Why R?

- Pros:
  - Reproducibility
    - Explicit documentation of steps
  - Versatile
    - Run on any operating system
    - Integrates with other software, languages, and data extensions
      - Python, Java, SAS, SPSS, Excel
  - Free and Open-Source w/ large active community
    - 10K+ packages CRAN, Twitter, GitHub, etc.
  - Comprehensive
    - Eliminates the need for multiple software
      - GIS, Excel, ENVI, etc.
  - Computationally Robust
    - Fast and allows for high level analysis
- Cons:
  - Steep Learning Curve
    - Requires a significant time investment especially starting with little to no coding experience
  - Limited “Point & Click”
    - R Commander or Radiant?
  - Open-Source
    - Rely on creators to follow coding etiquette
    - Version control and instability

Why R?

- Increasing in Popularity (especially in academia)
- User contributions significantly increased over the last decade



http://blog.revolutionanalytics.com/2014/01/in-data-scientist-survey-r-is-the-most-used-tool-other-than-databases.html

http://blog.revolutionanalytics.com/2016/03/16-years-of-r-history.html

Workshop Agenda

Workshop Expectations
What is R?
Understanding Data
Data Visualizations in R
Work in R Studio

What is Data?

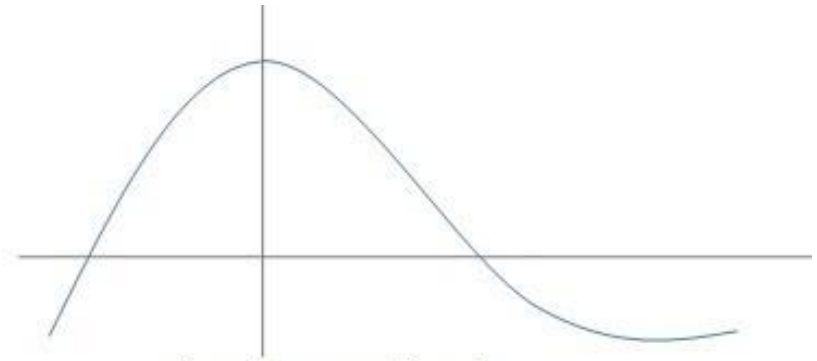
- Data is a collection of **objects** defined by **attributes**

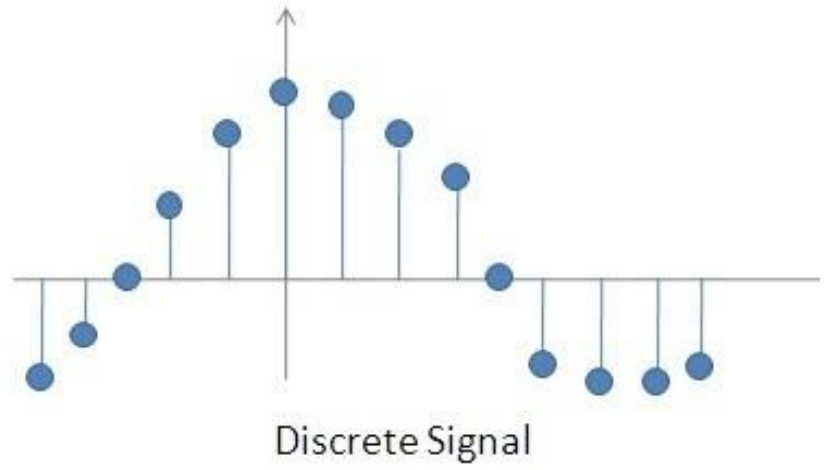
- An **attribute** is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.
  - Synonyms: ***Columns*****,** variables, fields, characteristics, features, etc.

- An **object** is the phenomena being described or evaluated
  - Examples: Bob/Sarah, house, substance, etc.
  - Synonyms: ***Rows***, records, points, cases, samples, instances, etc.

Attribute Classification

- Discrete Attribute – has a countable set of values
  - Examples: zip codes, number of words,
  - Typically represented as **integers**
- Continuous Attribute - has an infinite set of numbers and potential divisions
  - Example: temperature, height, weight
  - Typically represented as floating point (**decimal**)

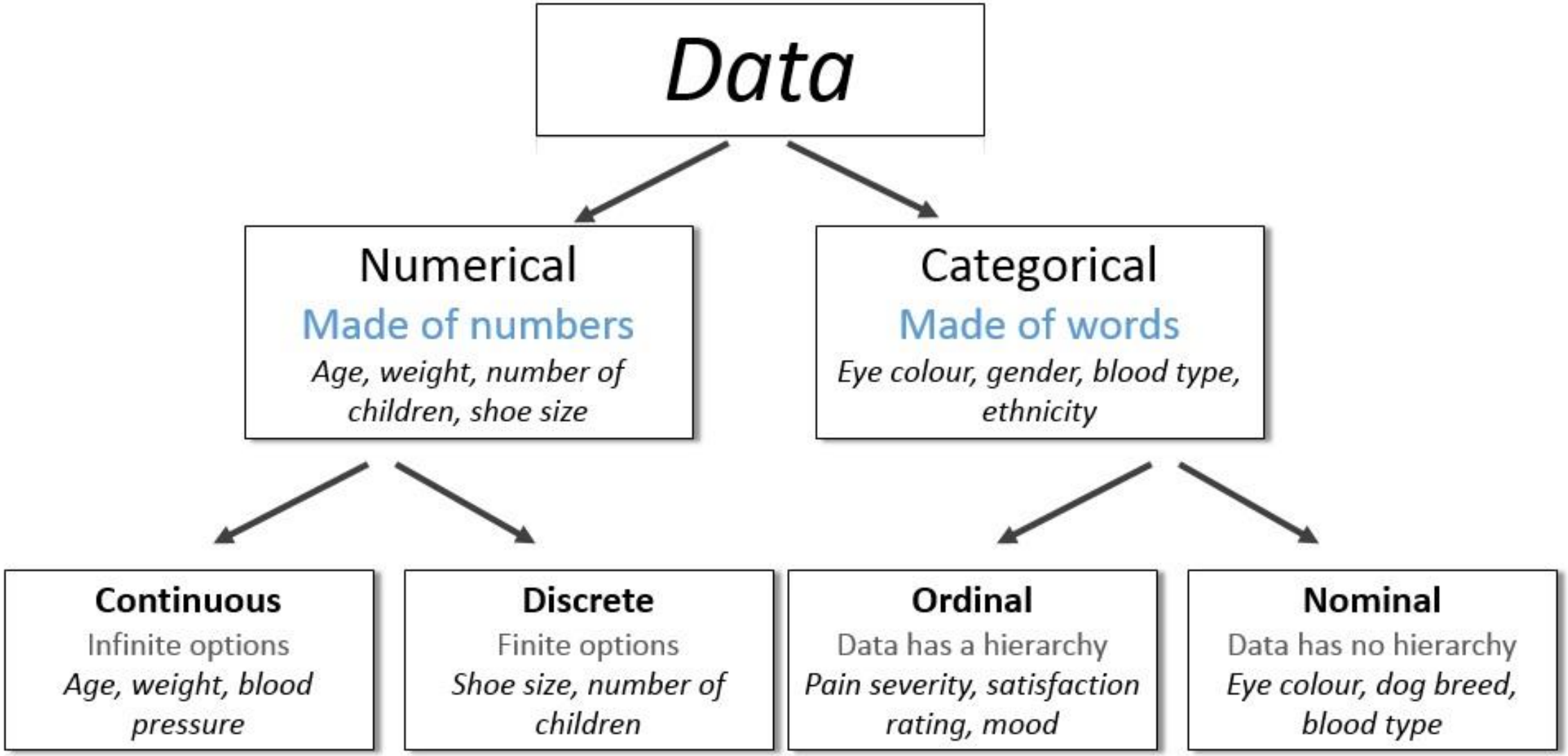




https://qph.fs.quoracdn.net/main-qimg-7badb966d5ff6063ddb515737011ed1b-c

Important Attribute Classes

- Categorical
  - Nominal - Data that can be counted, but not aggregated or ordered
    - Examples: Eye Color, Zip Code, Music Genre
  - Ordinal - Data that can be counted and ordered, but not aggregated.
    - Examples: Grades, Clothing Size, Positions (in a race)
- Numerical
  - Interval (metrics) - The difference in values are constant and meaningful
    - Examples: The difference between a temperature of 100°F and 90°F is the same difference as between 90°F and 80°F.
  - Ratio - An interval scale with an absolute zero
    - Examples: Income, Height, Weight



http://survivestatistics.com/variables/

Primitive Data Types
(Computer Language Data Types)

- Boolean:
  - True (T) or False (F)
- Char:
  - Characters and Strings – “A”, “Beta”, “There are different data types!”
- Int:
  - Integers – (1, 2, 100)
- Float/Double:
  - Decimal – (0.1, 0.2, 0.1352)

Data types in R

- R has a wide variety of data types including scalars, vectors (numerical, character, logical), matrices, data frames, and lists.

- A scalar **data structure** is the most basic data type that holds only a single atomic value at a time. Using scalars, more complex data types can be constructed.

R has 5 basic atomic classes
- logical (e.g., TRUE, FALSE)
- integer (e.g,, 2L, as.integer(3))
- numeric (real or decimal) (e.g, 2, 2.0, pi)
- complex (e.g, 1 + 0i, 1 + 4i)
- character (e.g, "a", "swc")

Data types in R

- Vector

```
a <- c(1,2,5.3,6,-2,4) # numeric vector
b <- c("one","two","three") # character vector
c <- c(TRUE,TRUE,TRUE,FALSE,TRUE,FALSE) #logical vector
```

Refer to elements of a vector using subscripts.

```
a[c(2,4)] # 2nd and 4th elements of vector
```

Data types in R

- Matrices

```
mymatrix <- matrix(vector, nrow=r, ncol=c, byrow=FALSE,
  dimnames=list(char_vector_rownames, char_vector_colnames))
```

byrow=TRUE indicates that the matrix should be filled by rows. byrow=FALSE indicates that the matrix should be filled by columns (the default). dimnames provides optional labels for the columns and rows.

```
# generates 5 x 4 numeric matrix
y<-matrix(1:20, nrow=5,ncol=4)

# another example
cells <- c(1,26,24,68)
rnames <- c("R1", "R2")
cnames <- c("C1", "C2")
mymatrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
  dimnames=list(rnames, cnames))
```

Data types in R

```
x[,4] # 4th column of matrix
x[3,] # 3rd row of matrix
x[2:4,1:3] # rows 2,3,4 of columns 1,2,3
```

- Arrays

Arrays are similar to matrices but can have more than two dimensions. See help(array) for details.

Data types in R

- Data Frames

A data frame is more general than a matrix, in that different columns can have different modes (numeric, character, factor, etc.). This is similar to SAS and SPSS datasets.

```
d <- c(1,2,3,4)
e <- c("red", "white", "red", NA)
f <- c(TRUE,TRUE,TRUE,FALSE)
mydata <- data.frame(d,e,f)
names(mydata) <- c("ID","Color","Passed") # variable names
```

There are a variety of ways to identify the elements of a data frame.

```
myframe[3:5] # columns 3,4,5 of data frame
myframe[c("ID","Age")] # columns ID and Age from data frame
myframe$X1 # variable x1 in the data frame
```

Data types in R

- Lists

An ordered collection of objects (components). A list allows you to gather a variety of (possibly unrelated) objects under one name.

```
# example of a list with 4 components -
# a string, a numeric vector, a matrix, and a scaler
w <- list(name="Fred", mynumbers=a, mymatrix=y, age=5.3)

# example of a list containing two lists
v <- c(list1,list2)
```

Identify elements of a list using the [[]] convention.

```
mylist[[2]] # 2nd component of the list
mylist[["mynumbers"]] # component named mynumbers in list
```

Data types in R

- **Factors**

Tell R that a variable is nominal by making it a factor. The factor stores the nominal values as a vector of integers in the range [1... k] (where k is the number of unique values in the nominal variable), and an internal vector of character strings (the original values) mapped to these integers.

```
# variable gender with 20 "male" entries and
# 30 "female" entries
gender <- c(rep("male",20), rep("female", 30))
gender <- factor(gender)
# stores gender as 20 1s and 30 2s and associates
# 1=female, 2=male internally (alphabetically)
# R now treats gender as a nominal variable
summary(gender)
```

An ordered factor is used to represent an **ordinal variable**.

```
# variable rating coded as "large", "medium", "small'
rating <- ordered(rating)
# recodes rating to 1,2,3 and associates
# 1=large, 2=medium, 3=small internally
# R now treats rating as ordinal
```

Data types in R

- Other useful functions

- length(object) # number of elements or components
- str(object) # structure of an object
- class(object) # class or type of an object
- names(object) # names

- c(object,object,...) # combine objects into a vector
- cbind(object, object, ...) # combine objects as columns
- rbind(object, object, ...) # combine objects as rows

- object # prints the object

- ls() # list current objects
- rm(object) # delete an object

- newobject <- edit(object) # edit copy and save as newobject
- fix(object) # edit in place

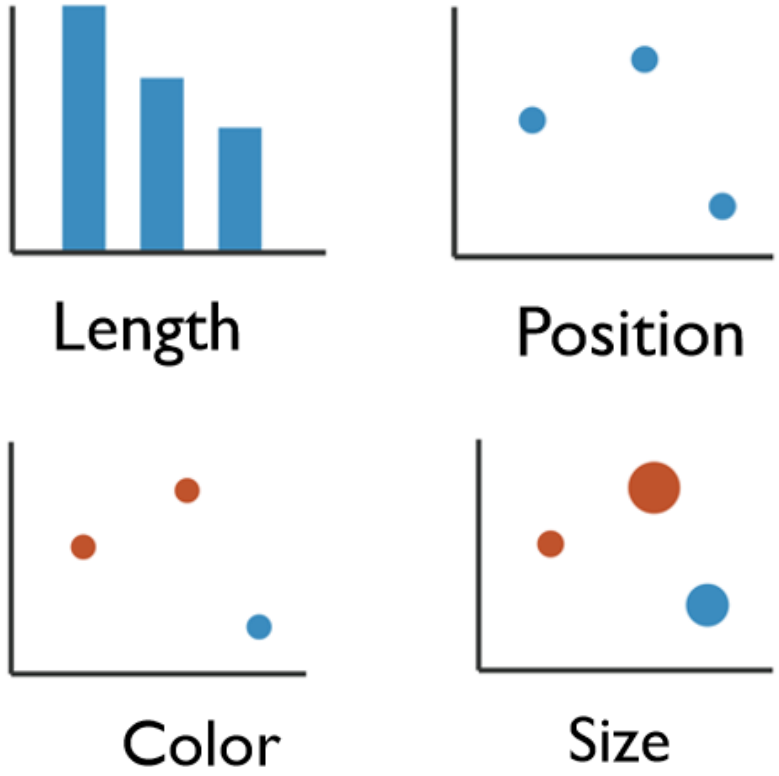
https://www.statmethods.net/input/datatypes.html

Workshop Agenda

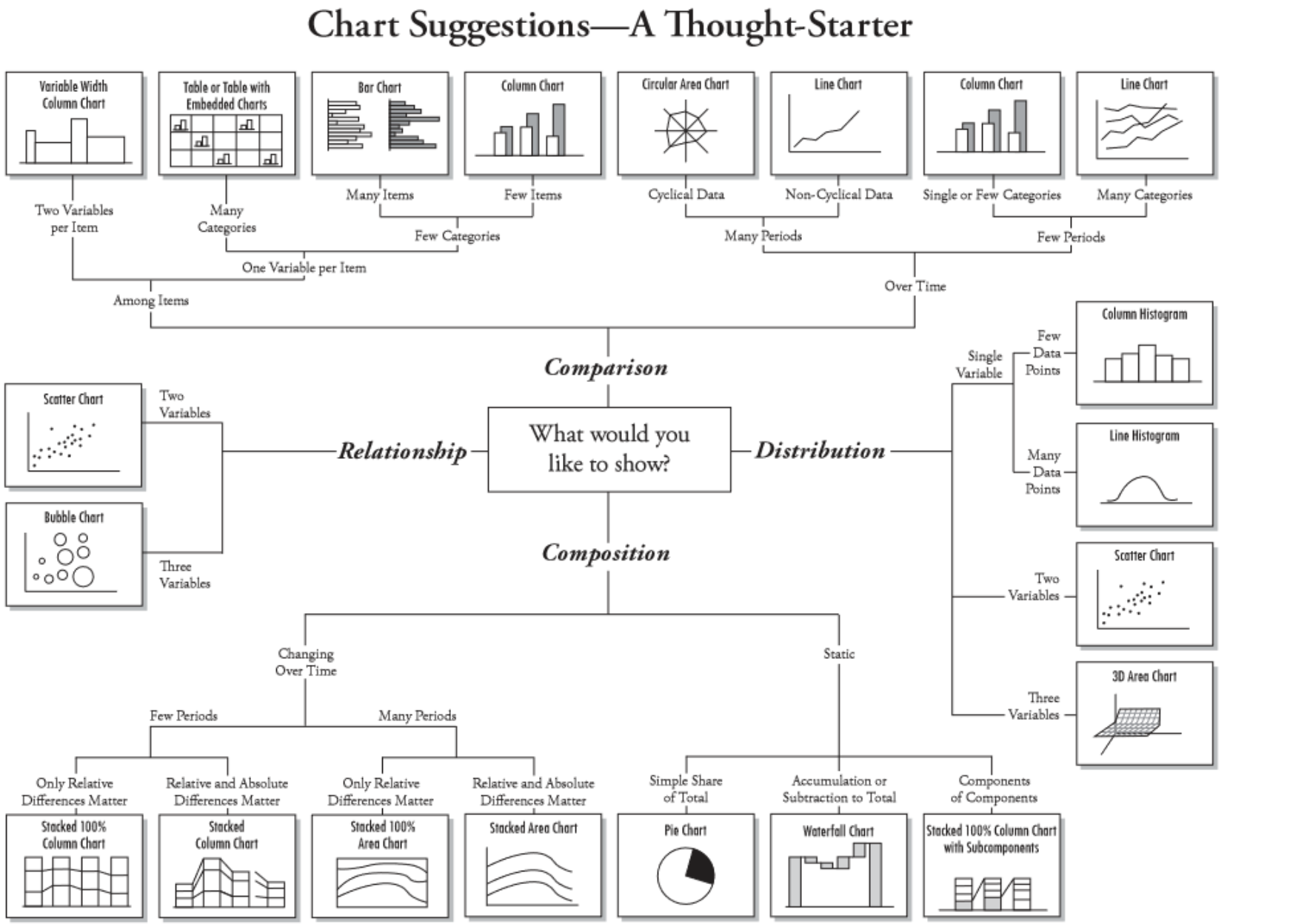
Workshop Expectations
What is R?
Understanding Data
Data Visualizations in R
Work in R Studio

Enhancing Visualizations

- 1 Dimensional Data
  - Length
- 2 Dimensional Data
  - Position
- 2+ Dimensional Data
  - Position
  - Color Hue/Saturation
  - Size
  - Shape



Four Basic Chart Types



http://extremepresentation.typepad.com/files/choosing-a-good-chart-09.pdf

Grammar of Graphics

Originated by Leland Wilkinson, simplified by Hadley Wickham and others.

- Data – The raw materials of your visualization
- Layers – What you actually see on the plot (plots,lines,etc.)
- Scales – Maps the data to grpahical output
- Coordinates – The visualization's perspective (normally a grid)
- Faceting – Provides details into the data (analoguous to pivot tables)
- Themes – Control the details of the display (color scheme, fonts)

http://vita.had.co.nz/papers/layered-grammar.pdf

Grammar of Graphics

Describe all the non-data ink

Plotting space for the data

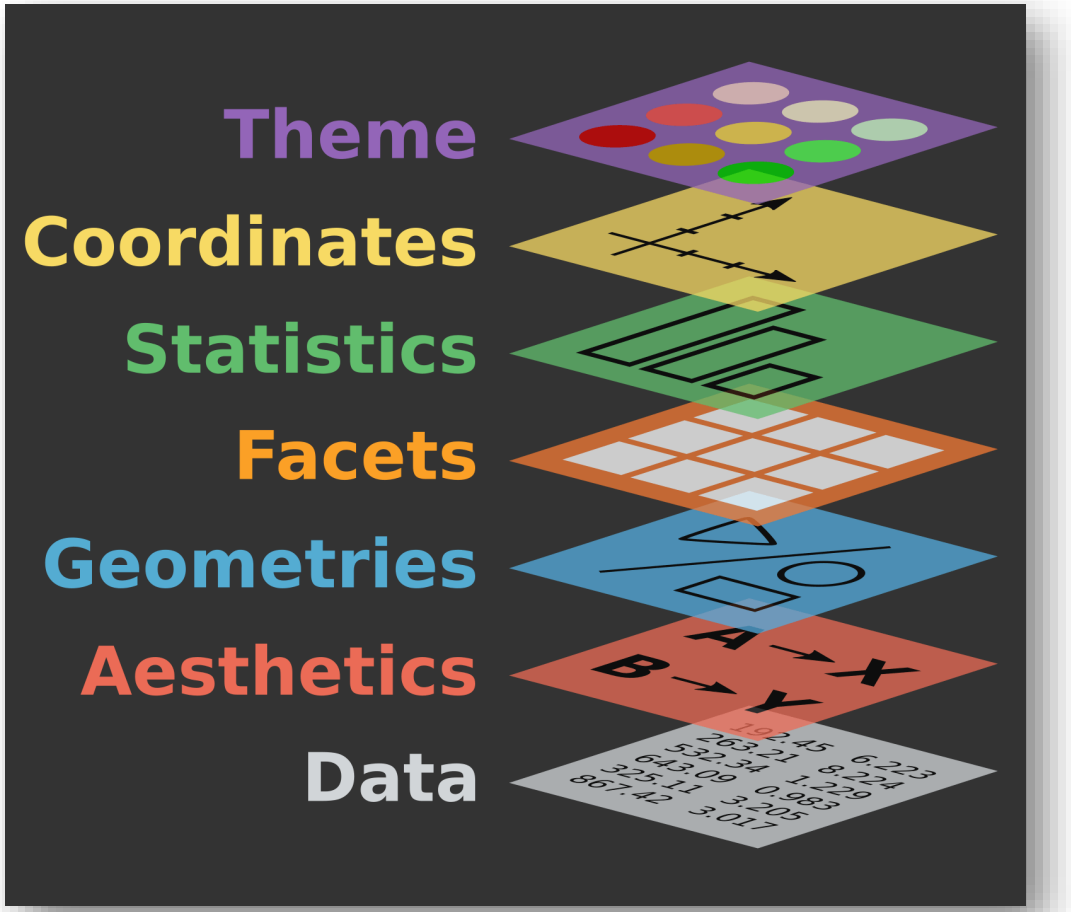
Statistical models & summaries

Rows and columns of sub-plots

Shapes used to represent the data

Scales onto which data is mapped

The actual variables to be plotted



The Basics

- Data – The raw materials of your visualization
- Aesthetics – The mapping of your data to the visualization
  - X-axis is age
  - Y-axis is survival
- Layers – Any visualization requires at least one layer and in ggplot2 these are typically the geoms.
  - Example a barchart is geom_bar()

Want to continue learning R?

Academic Courses:

- CS 2005C – Introduction to Programming for Informatics with Python and R
- BE 8083 – Data Analysis with R and SAS
- BANA 5143 – Statistical Computing
- PH 7011 – Statistical Computation and Software

Lynda (Many free courses):

- Learning R
- Data Wrangling with R
- R Statistics Essential Training

Library Workshops:

- https://webapps2.uc.edu/ce/facdev/Workshops

Online Help/Resources

- Google!
- www.YouTube.com
- www.stackoverflow.com
- www.r-bloggers.com
- Twitter: #rstats

Contact Research & Data Services West

Email: AskData@uc.edu

Web: https://libraries.uc.edu/research-teaching-support/research-data-services.html

Visit: 240 Braunstein Hall (Geology-Math-Physics Library)

Consultations: walk-ins during open consultation hours or appointment by email

Twitter: @DataVizJohansen

University of CINCINNATI