

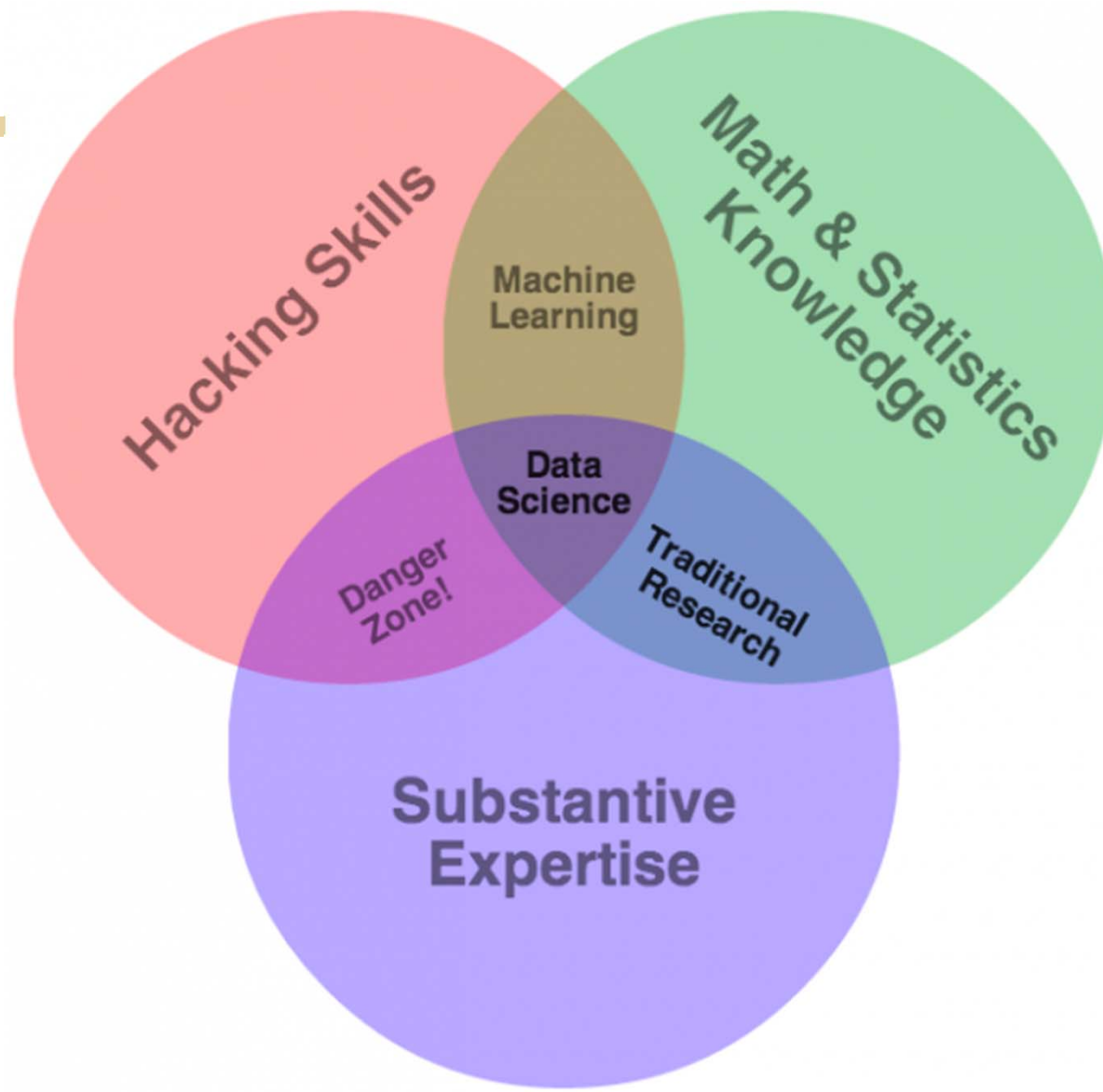
UNIVERSITY *of* WASHINGTON

Data Science UW

Methods for Data Analysis

Introduction and Data Exploration
Lecture 1
Nick McClure





W

Course Purpose

- > This course focuses on essential concepts
- > We are building foundations for your data science skills
- > Course Objectives:
 - Learn methods to explore and understand data.
 - Understand the core concepts of statistics.
 - Understand and implement various statistical procedures in R.
 - Describe and interpret analytical results from common statistical methods.
 - Expand R programming skills to be able to write/test/log code from scratch.
 - Work with structured and unstructured data.
- > See syllabus for more information:
 - <https://canvas.uw.edu/courses/1087732/pages/course-abbrev-course-syllabus>



Course Requirements and Grading

This course will be graded by attendance, homework, and an individual project.

- > Attendance: You MUST attend at least 8 out of 10 classes. **This is a non-negotiable UW requirement.**
- > Homework must be completed by the start of the next class. (Assigned weeks 1-8).
 - Returned as a 0,1, or 2.
 - > 0 = Not done or a major part wrong/missing.
 - > 1 = Completed, but missing or got wrong 1 or 2 parts.
 - > 2 = Completed with at most minor issues. Demonstrates full understanding of subject.
- > Individual Project: Due at the start of the last class.
 - Counts as 8 points.



Course Requirements and Grading

There is a total of 24 possible points. (16 pts for hmk + 8 project)

- > Must get 18 total points to pass.
- > All homework assignments must use good R coding technique
- > The individual project must be production level code.



Office Hours and Contact Information

> Contact me at:

- stephen.elston@quantia.com

> When I'm *usually* available:

- Off/on for simple things during work. (M-F 8am-5pm PST)
- Sunday various afternoon/evening times.

Emergency contact: 402-980-3192

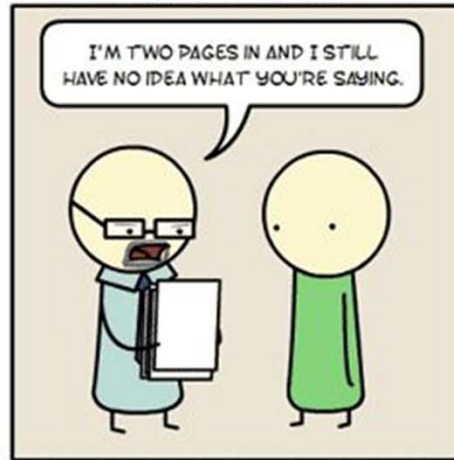


Review

PYTHON



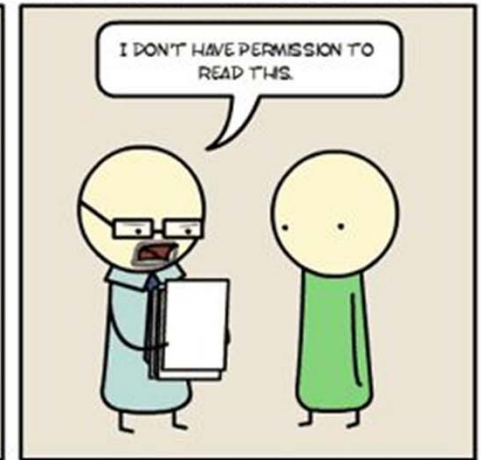
JAVA



C++



UNIX SHELL



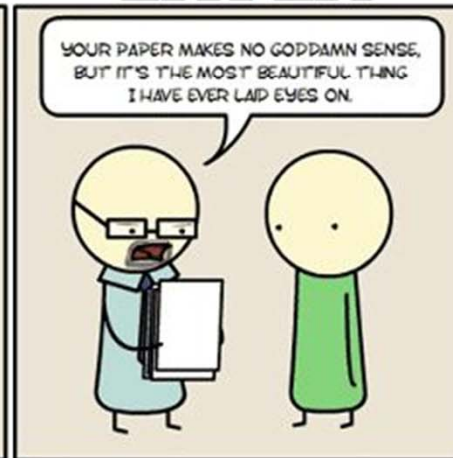
ASSEMBLY



C



LATEX



HTML



SQL Review

- > SQL is the 'Lingua Franca' of data access
- > SQL (to know):
 - *Create tables*
 - *Drop tables*
 - Select, where, groupby
 - Joins (Inner, outer, right, left)
 - Temp tables
 - Coalesce, Cast, Case



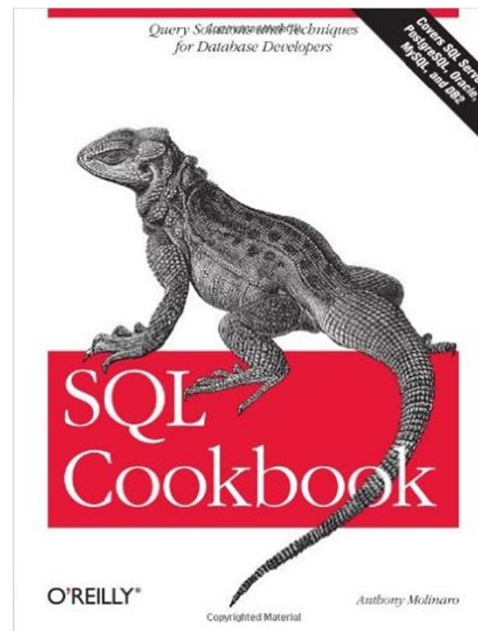
SQL Resources

SQL Tutorial and Resources

<http://www.w3schools.com/sql/>

Querying with Transact SQL Course, Graeme Malcom

<https://www.edx.org/course/querying-transact-sql-microsoft-dat201x-3>



Prepare for R Demos

> Install R

<https://cran.r-project.org/>

-or-

<https://mran.revolutionanalytics.com/download/>

> Install RStudio

<https://www.rstudio.com/products/rstudio/download/>



GitHub

- > Code, data and slides for this course are in a GitHub repository

<https://github.com/StephenElston/DataScience350>

- > Install GitHub for desk top

<https://help.github.com/desktop/guides/getting-started/installing-github-desktop/>

- Or, just download the zip files



R Review

> R resources:

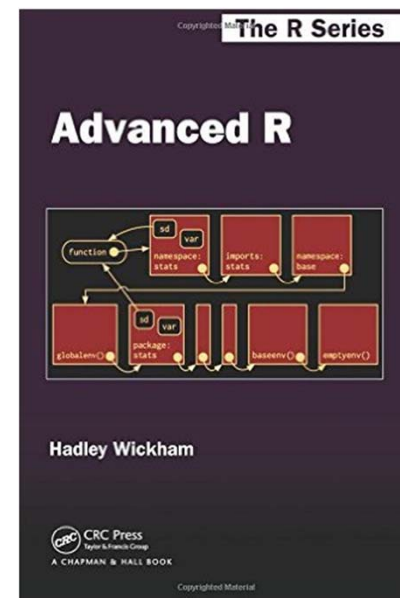
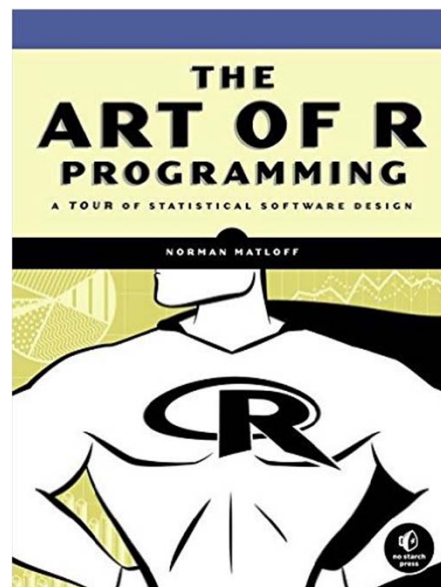
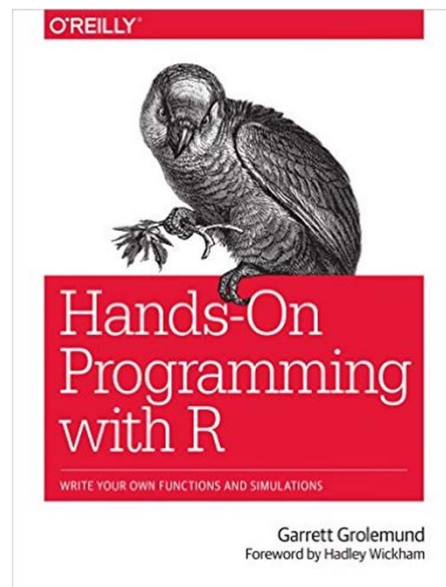
- R page:
 - > <http://www.r-project.org/other-docs.html>
- Stackoverflow:
 - > <http://www.stackoverflow.com>
- ‘Little’ R intro:
 - > <http://cran.r-project.org/doc/contrib/Rossiter-RIntro-ITC.pdf>
- Quick R:
 - > <http://statmethods.net/>
- There are many tutorials available online, e.g.,
 - > <http://cyclismo.org/tutorial/R/>
- Google’s Style Guide:
 - > <http://google-styleguide.googlecode.com/svn/trunk/google-r-style.html>



More R Resources

R Inferno, Pat Burns

http://www.burns-stat.com/pages/Tutor/R_inferno.pdf



Statistics Review

> Familiar Concepts:

- Discrete vs. Continuous Distributions
- Probability
- Statistics
- $y = mx + b$ vs $\bar{Y} = \mathbf{M} \cdot \bar{X} + \mathbf{B}$

> These concepts are the focus of this course.



Counting Review

> Factorials

- Count # ways to order N things = $N!$

> Permutations

- Count # of ways to **order** R things from N things = $N!/(N-R)!$
- Ordering matters
- $P(N,R)$

> Combinations

- Count # of ways to **group** R things from N things = $N!/(R!(N-R)!)$
- Ordering doesn't matter
- $C(N,R)$ or $\binom{N}{R}$

> We will talk about this in depth next class.



Data Distributions (Discrete)

> Discrete Distribution Properties

- Sum of probability of all possible events must equal 1.
- Probability of event equal to value of distribution at point.
- All values strictly in range 0-1.



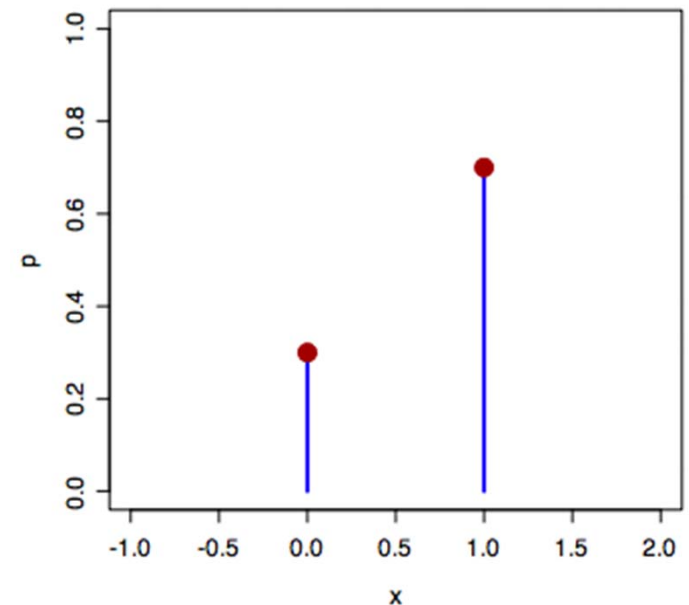
Data Distributions (Discrete)

> Bernoulli (1 event, e.g.: coin flip)

$$P(x) = \begin{cases} p & \text{if } x = 1 \\ (1 - p) & \text{if } x = 0 \end{cases}$$

$$P(x) = p^x (1 - p)^{(1-x)} \quad x \in \{0,1\}$$

- Mean = p
- Variance = $p(1-p)$



Data Distributions (Discrete)

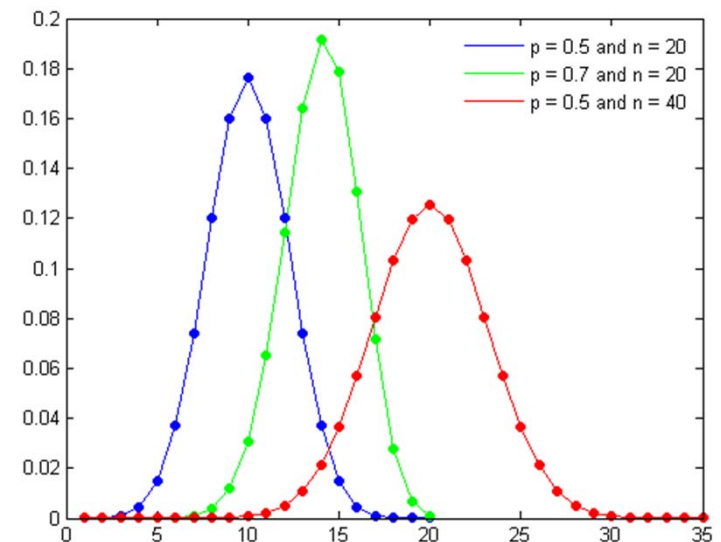
> Binomial (Multiple Bernoulli's Events)

- Multiple Independent events = Product of Bernoulli Probabilities

$$P(x|N, p) = \binom{N}{x} p^x (1 - p)^{(N-x)}$$

- Mean = np
- Variance = $np(1-p)$

Note: for larger n , we approximate this by a normal distribution.



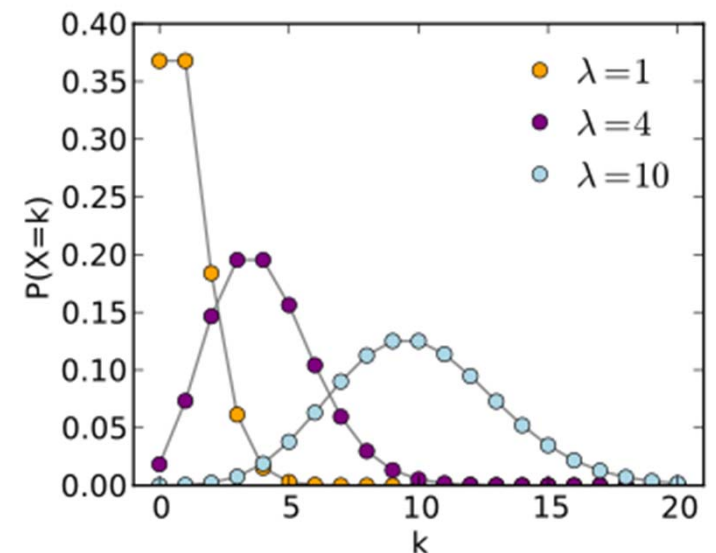
Data Distributions (Discrete)

> Poisson (Count of number of events in a time span)

$$P(x|\lambda) = \frac{\lambda^x}{x!} e^{-\lambda}$$

- Mean = λ
- Variance = λ

Interpret as the rate of occurrence of an event is equal to lambda in a finite period of time.





R Demo



Discrete distributions



Data Distributions (Continuous)

> Continuous Distribution Properties

- Area under the curve must be equal to 1.
- Probability a range of values of an event equal to AREA under curve.
- No negative values.
- Probability of a single, exact value is 0.



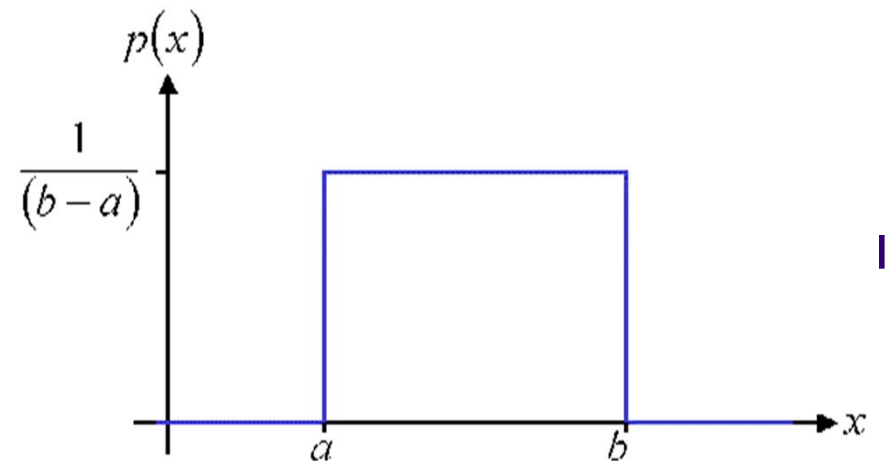
Data Distributions (Continuous)

> **Uniform** (flat, bounded)

$$P(x) = \begin{cases} \frac{1}{(b-a)} & \text{if } a \leq x \leq b \\ 0 & \text{if } x < a \text{ or } x > b \end{cases}$$

> Used for parameter priors. (future discussion)

- Mean = $(a+b)/2$
- Variance = $(1/12)(b-a)^2$



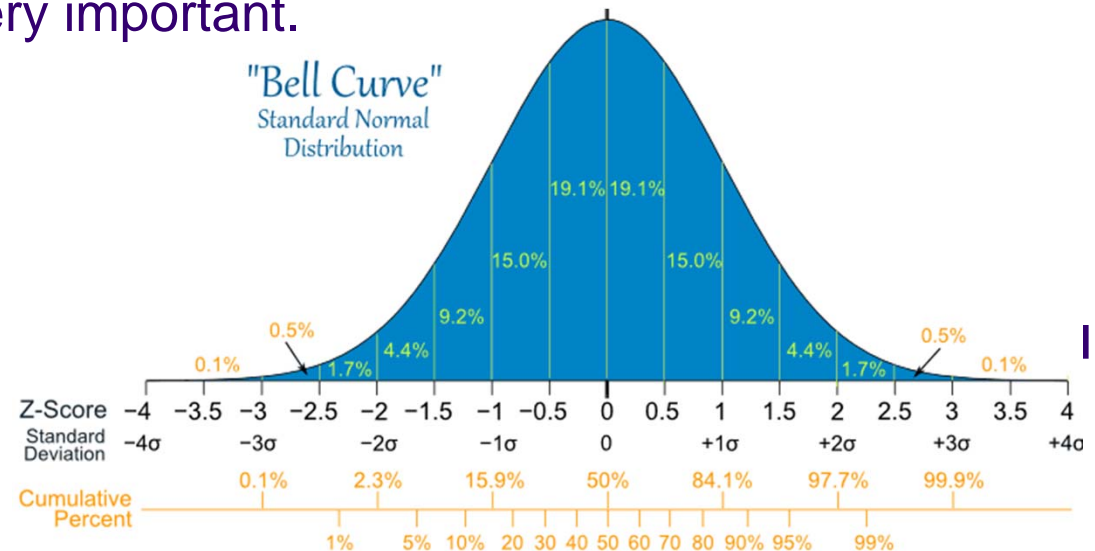
Data Distributions (Continuous)

> Normal (Gaussian) distribution

- Most common and occurs naturally.
- Defined by a mean and variance only. (standard = $N(0,1)$)

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Has very nice properties.
- Tests for normality are very important.



Data Distributions (Continuous)

- > Student's T (normal for small samples)
 - Important for hypothesis testing smaller sample sizes.
 - Used for:
 - > Testing of mean value when st. dev. is unknown.
 - > Testing difference between two distribution means.
 - Looks very similar to the normal distribution.





R Demo



Continuous distributions



R review and summary statistics

- > Purpose: To gain a clear understanding of your data.
 - How large is it?
 - What columns are of interest?
 - Missing data?
 - Outliers?



R Vectors, Arrays and Lists

- > Vectors have one dimension, one data type
- > Vectors are the atomic object in R
- > R is optimized for vector operations
- > Array has multiple dimensions, all data of one type
- > Matrix is a 2D array
- > Lists are comprised of other R object
- > Elements of a list can be of any type and dimension



Data frames

- > Rectangular tables: cannot be ragged
- > List with special attributes
- > Each column is vector of single type
- > Columns can have names
- > Rows can have names
- > Subsetting operator []
- > Column operator \$



Functional Programming with R

Definition from Wikipedia

In computer science, **functional programming** is a programming paradigm—a style of building the structure and elements of computer programs—that treats computation as the evaluation of **mathematical functions** and **avoids changing-state** and mutable data. It is a declarative programming paradigm, which means **programming is done with expressions or declarations**^l instead of statements. In functional code, the **output value of a function depends only on the arguments** that are input to the function, so calling a function f twice with the same value for an argument x will produce the same result $f(x)$ each time. **Eliminating side effects**, i.e. changes in state that do not depend on the function inputs, can make it much easier to understand and predict the behavior of a program, which is one of the key motivations for the development of functional programming.



Functional programming in R

- > R is a functional language
- > Expressions and function are objects
- > Functions can be named or anonymous
- > Use functional operators rather than loops



R programming practice

- > Vectorize for performance
- > Use functions; **avoid repetitive code!**
- > Make use of functional operators
- > Comment your code
- > Names should mean something





R Demo

Basic data wrangling and functional programming



Data Exploration (Descriptive Statistics)

- > Purpose: To gain a clear understanding of your data.
 - How large is it?
 - What columns are of interest?
 - Missing data?
 - Outliers?



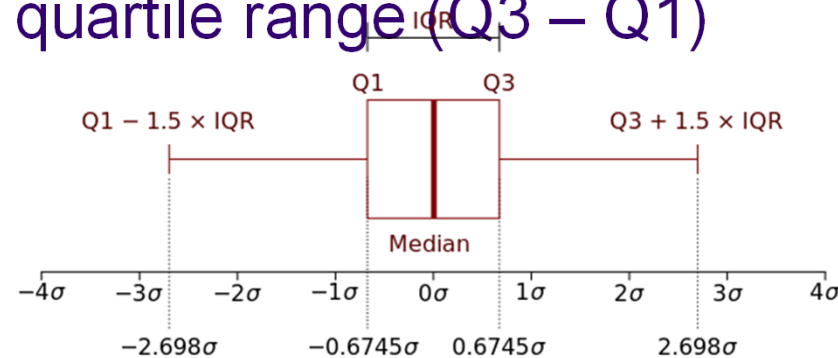
Numerical Exploration

- > `str()`: structure of the data frame
- > `summary()`: summary of each of the columns
- > `head()` / `tail()`: top / bottom of data frame
- > `table()`: frequency table



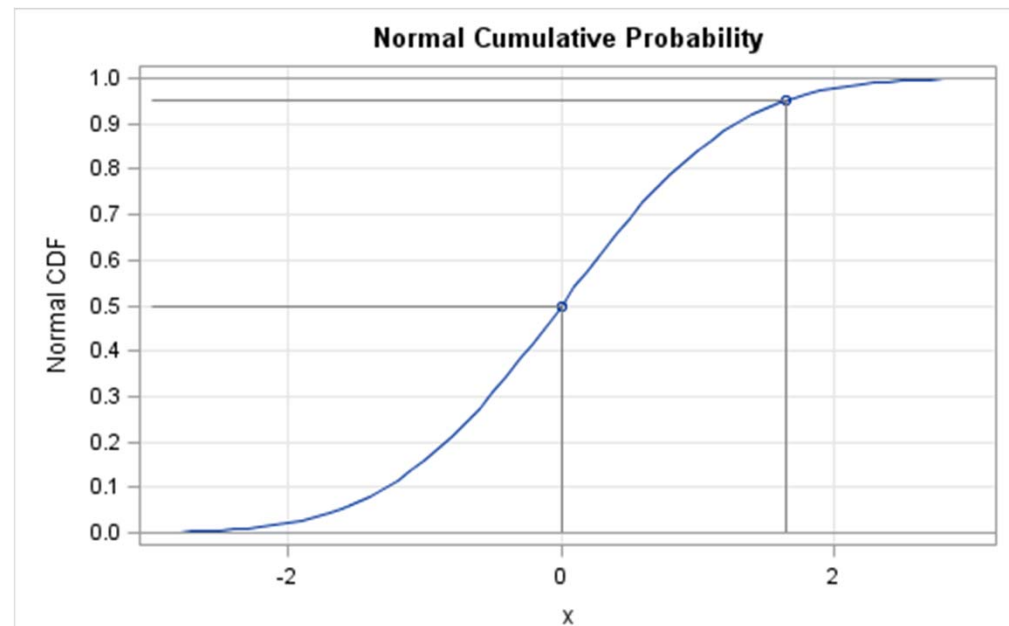
Numerical Exploration

- > Quartile == $\frac{1}{4}$ values
- > IQR(): inner quartile range ($Q3 - Q1$)



Numerical Exploration

- > `quantile()`: quantiles of numerical vectors
 - Quantiles are inverse values of the CDF (cumulative distribution function).
 - Standard Normal: (shown in figure)
 - > $\text{Quantile}(0.5) = 0$, means at $x=0$, 50% of the distribution lies to the left. (This is also the median)
 - > $\text{Quantile}(0.95) = 1.65$



Numerical Exploration

> Relationships:

- `cov()`: covariances

$$\text{cov}(x, y) = E((x - \mu_x)(y - \mu_y))$$

- Interpretation: Expected value of the differences between x and y and their corresponding mean.
- E.g. if x is above its mean when y is also above its mean, then they will have a high covariance.
- Not bounded.



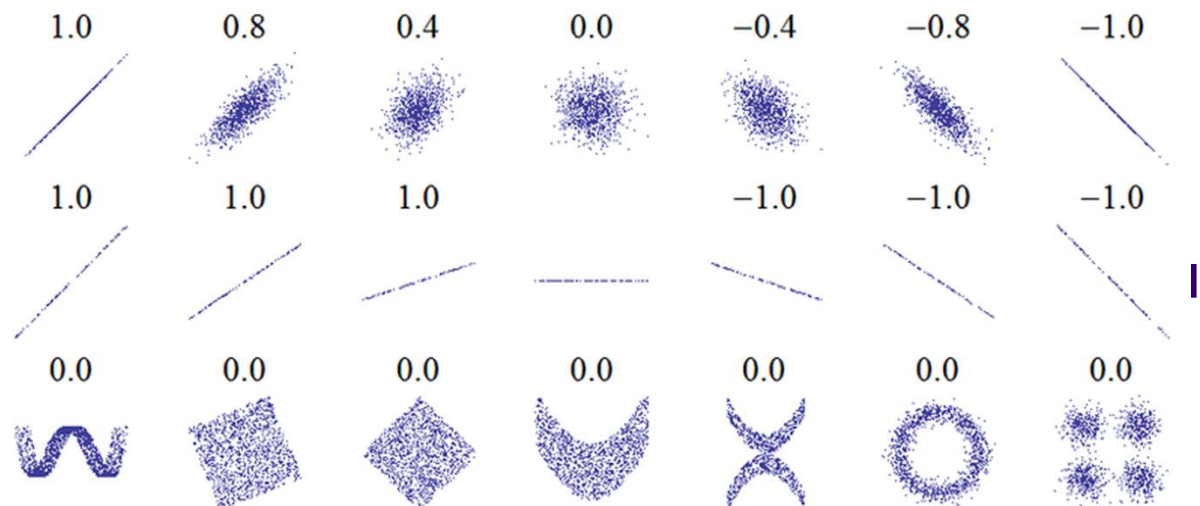
Numerical Exploration

> Relationships:

- `cor()`: correlations (pearsons)

$$\text{cor}(x, y) = \frac{E((x - \mu_x)(y - \mu_y))}{\sigma_x \sigma_y}$$

- Bounded between 0 and 1.
- **Does not indicate causation!!**



Distribution Transformations

- > The purpose of transforming a variable is to make it easier to distinguish between values.
 - Most commonly we are looking to transform a distribution to be normal.
- > Common Transformations
 - Log-based:
 - > $\text{Log}(x)$, $\log(x+1)$, $\log(x - \min(x) + 1)$
 - N-th Root based:
 - > $X^{(1/n)}$
 - Any combination you can think of (remembering math rules).
- > We will cover normality tests in a later class.





R Demo



Summary statistics



Introduction to dplyr

Data Wrangling Cheat Sheet:

<https://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf>

- > Data scientists spend most of their time on data munging or data wrangling
- > dplyr package provides a regular grammar for most data wrangling
- > Optimized for fast operations on data frames
- > Chain verbs (operators) for fast operations



Col1	Col2	Col3
2012	14	45
2013	13	76
2013	34	65
2014	23	47

```
library(dplyr)
df <- read.csv('data.csv',
header = TRUE,
stringsAsFactors = FALSE)
```

Col1	Col2	Col3
2012	12	45
2013	32	65
2013	34	65
2014	23	47

```
df <- filter(df, Col1 ==
2013)
```

Col1	Col3	Col3
2012	44	45
2013	76	76
2013	64	65
2014	43	47

```
df <- select(df, Col1, Col3)
```

Col1	Col2		Col3	Col4	
2012	14	14	45	45	59
2013	13	13	76	76	89
2013	34	34	65	65	99
2014	23	23	47	47	70

```
df <- mutate(df, Col4 = Col2 +  
Col3)
```

Other useful dplyr verbs include:

```
df <- group_by(df, Col1)
```

```
df <- distinct(df, Col1)
```

```
df <- arrange(df, Col1)
```

```
df <- slice(df, 10:15)
```

```
df <- sample_frac(df, 0.5)
```

```
df <- sample_n(df, 500)
```

```
df <- summarize(df, m1 =  
mean(Col1))
```

Col1	Col2		Col3	Col4	
2012	13	14	76	45	89
2013	34	13	65	76	99
2013		34		65	
2014		23		47	

```
df <- df %>%
  filter(Col1 == 2013) %>%
  mutate(Col4 = Col2 + Col3)

iris %>% group_by(Species) %>%
  summarise(...)
```

Simpsons Paradox

- > Slicing up data in different ways can create different results.
- > Generally arises in context of larger dataset with latent variable
- > <http://vudlab.com/simpsons/>
- > **R Demo with dplyr**



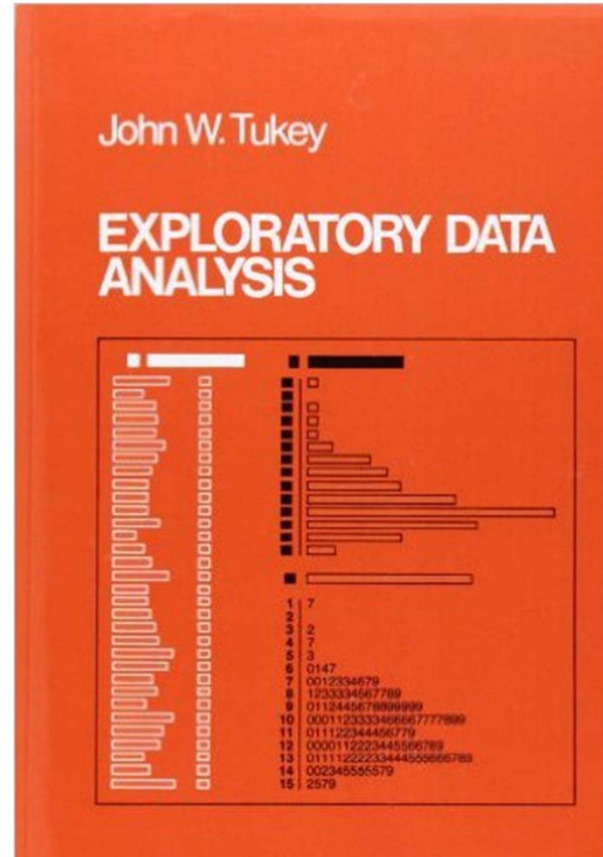
Exploratory data analysis

- > Explore the data with visualization
- > Understand the relationships in the data
- > Use multiple views of data
- > Aesthetics for multiple dimensions
- > Conditioning to project multiple dimensions



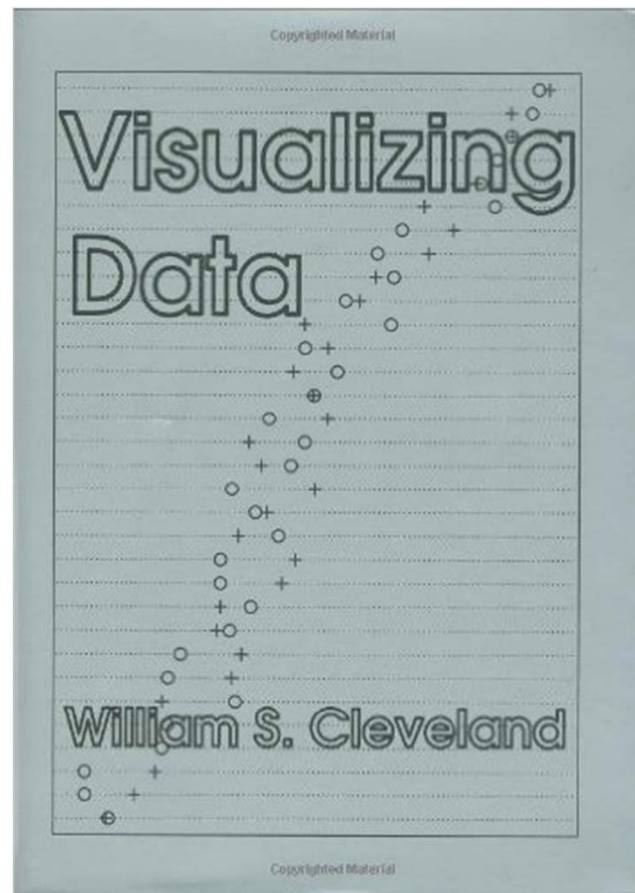
Seminal Book

John Tukey, Exploratory Data Analysis, 1977, Addison-Westley



Seminal Book

Visualizing Data, William S. Cleveland



W

Views of data

- > Data contains complex relationships
- > Explore data with multiple views
- > Views show different aspects of the relationships
- > Different plots highlight different relationships



Different plots for different views

- > Scatter
- > Scatter plot matrix
- > Line plots
- > Bar plots
- > Histograms
- > Box plots
- > Violin plots
- > Q-Q plots



Visualization Aesthetics

Aesthetics expand dimensionality of projection

- > Color
- > Shape
- > Size
- > Transparency
- > Aesthetics specific to plot type
- > **Don't over do it!**



Presenting charts

Charts must inform, not confuse!

- > Creating good visualization is iterative, and lots of work
- > Axis labels
- > Title
- > Legend
- > Large symbols and lines
- > Be sensitive to color blindness
- > No pie charts, please!
- > **Simplify, simplify, simplify!**

W

Grammar of Graphics: ggplot2

Grammar builds plot

> Import library

```
library(ggplot2)
```

> Define plot basics

```
p1 = ggplot(df, aes(x = col1, y = col2, by  
= col3)) +  
      geom_plotttype(aes(asthetics))
```

> Chain to add attributes

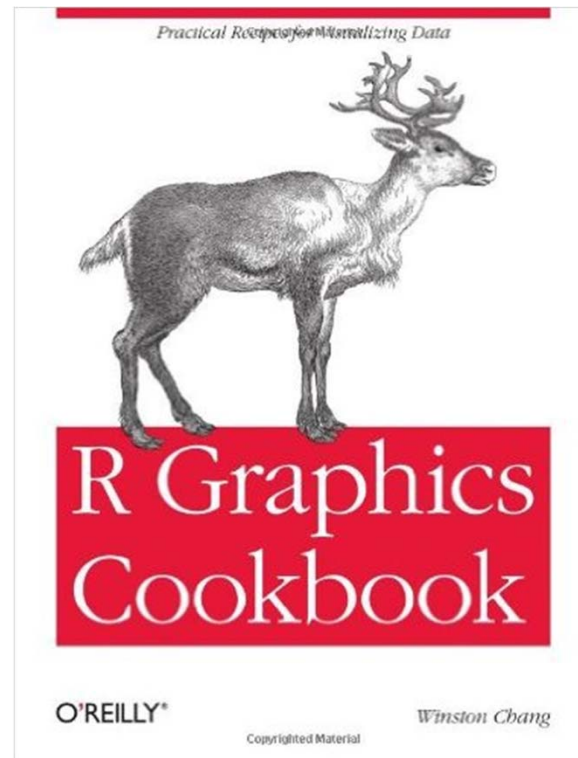
```
p1 + xlab('xlab') + ylab('ylab') +  
      ggtitle('The Title') +  
      other_attributes(...)
```

W

ggplot2 resources

ggplot2 cheat sheet

<https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf>



W



R Demo



Data Visualization



Assignment

Homework 1:

- Explore auto price data set.
- Write R program that shows/illustrates 3 key takeaways of your choosing from exploring the data.
- You should submit:
 - > **ONE R-script.**
 - > **One word document with 3 key points.** (example next page).



Example Takeaway

- > The price of automobiles is dependent on feature x . The levels of feature x separate the following specific types of autos. Specifically these include..... The charts illustrate this relationship. Examining the chart shows.....



Recommended Readings

- > An Introduction to Data Science, Chapters 3 and 9
- > Statistical Thinking for Programmers, Chapter 2

