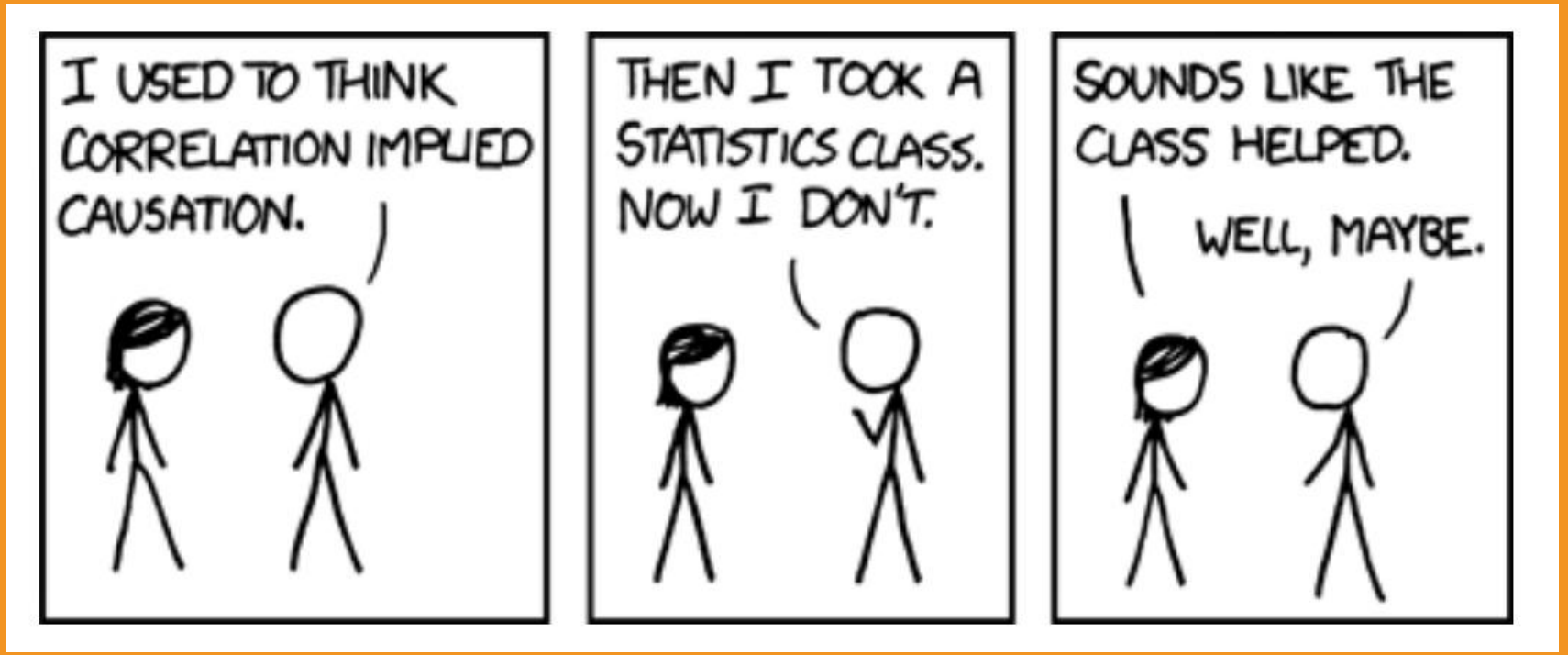
Week 3: Statistics and Exploratory Data Analysis

SESSION 1: WHAT IS DATA?

STARFISH SCHOOL 2021

Introduction to Basics in Statistics



https://xkcd.com/552/

Types of Data

Types of Data

Quantitative

- Continuous
  - Real or complex numbers
- Discrete
  - integers

Categorical

- Nominal
  - e.g., categories A, B, C, or I, II, III
- Ordinal
  - Ordering matters, e.g., a *Likert Scale* used in a survey: 1,2,3,4,5

Types of Data

Quantitative

- Continuous
  - Real or complex numbers
- Discrete
  - integers

mass of star
flux

planets
photon counts

Categorical

- Nominal
  - e.g., categories A, B, C, or I, II, III
- Ordinal
  - Ordering matters, e.g., a *Likert Scale* used in a survey: 1,2,3,4,5

stellar type
galaxy type

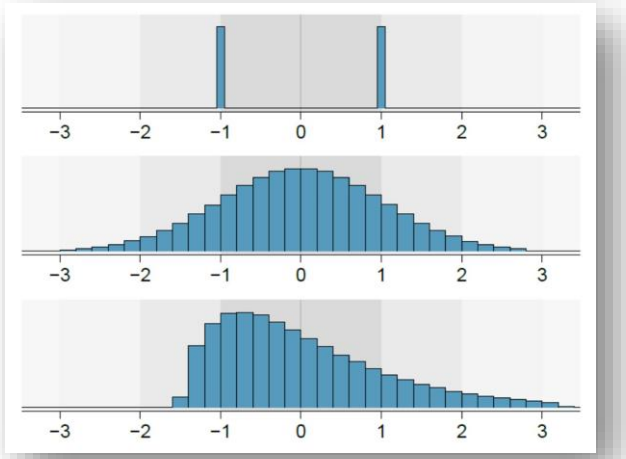
What astronomy examples can you think for each type?

Distributions

What exactly is a distribution?

A distribution…

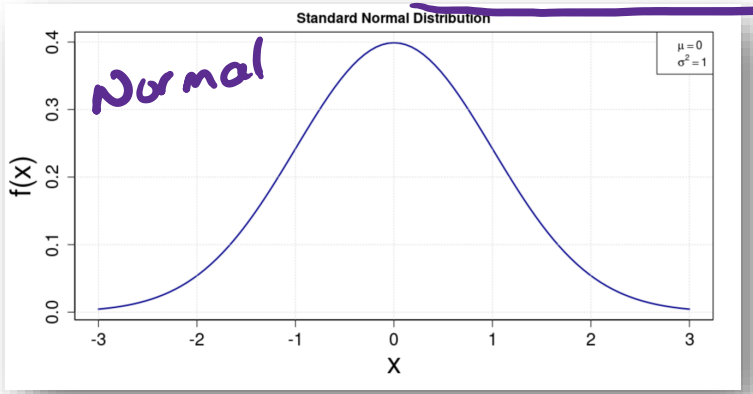
- Tells you the frequency or relative frequency of each possible value/event, or of some data that was collected
- Could be empirical or analytic
- Can be useful for modelling a population of objects
- Is often a foundation of statistical reasoning
- Can be continuous or discrete
- That is analytic has parameters that define its shape
- Can be univariate or multivariate

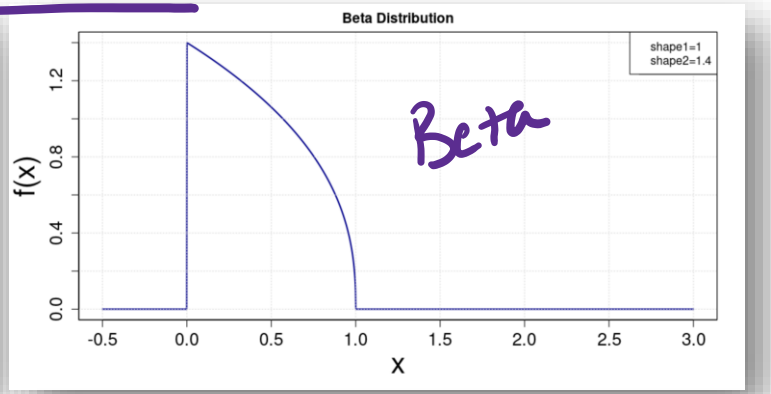


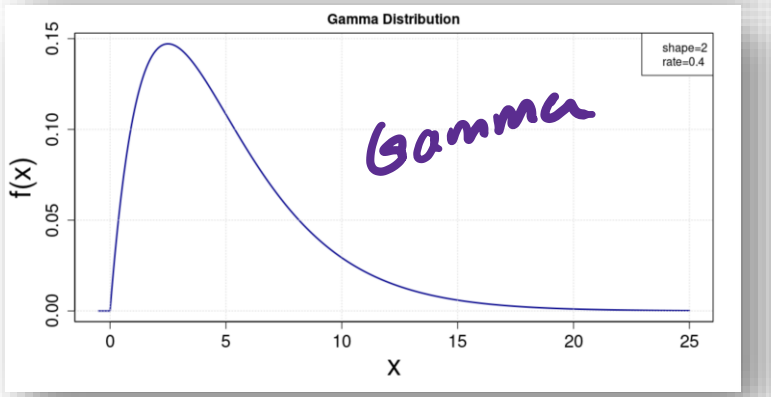
Example histograms (figure from Open Intro Statistics 4th ed.)

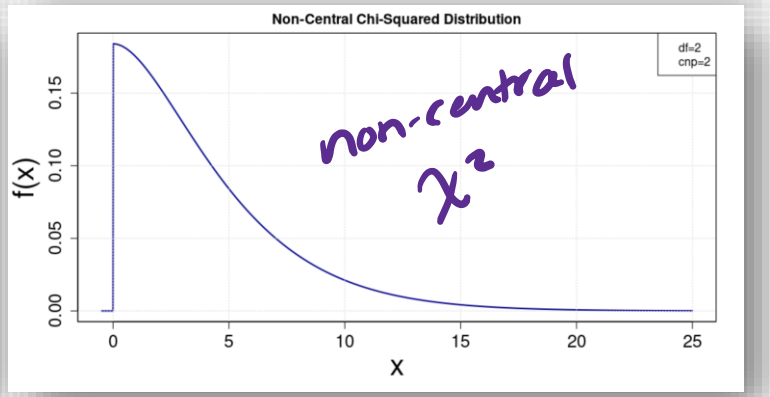
Some analytic probability distributions (plotted by me)

univariate





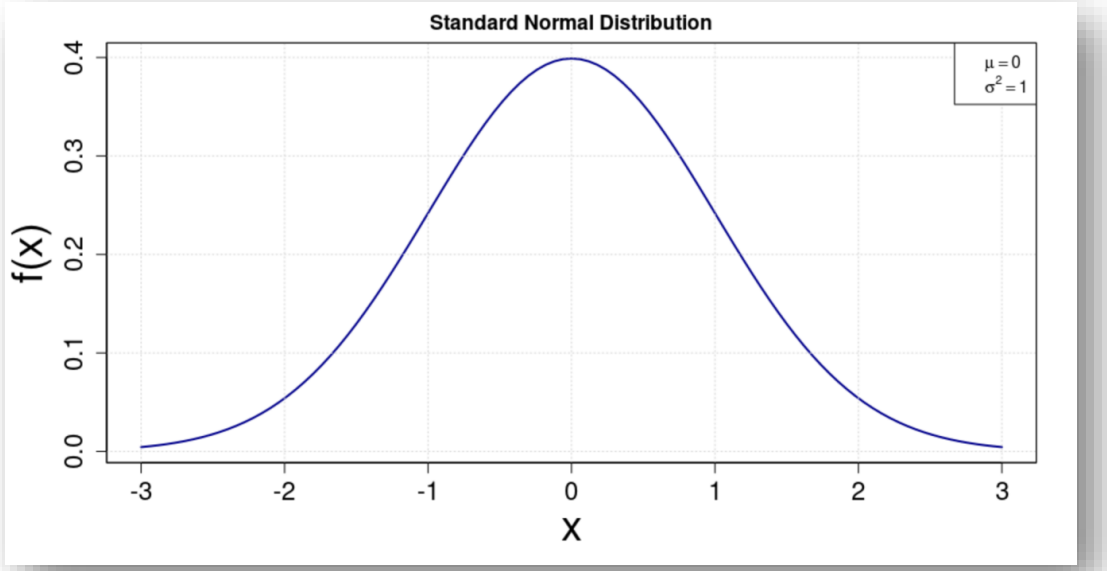




Probability Distributions

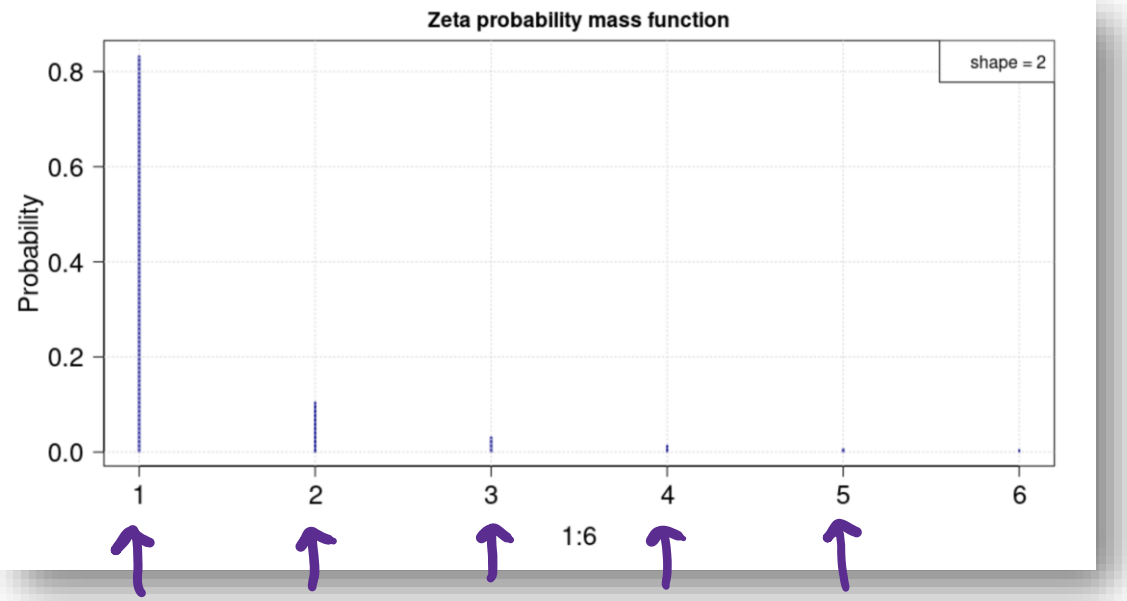
Continuous quantities

probability density function (pdf)

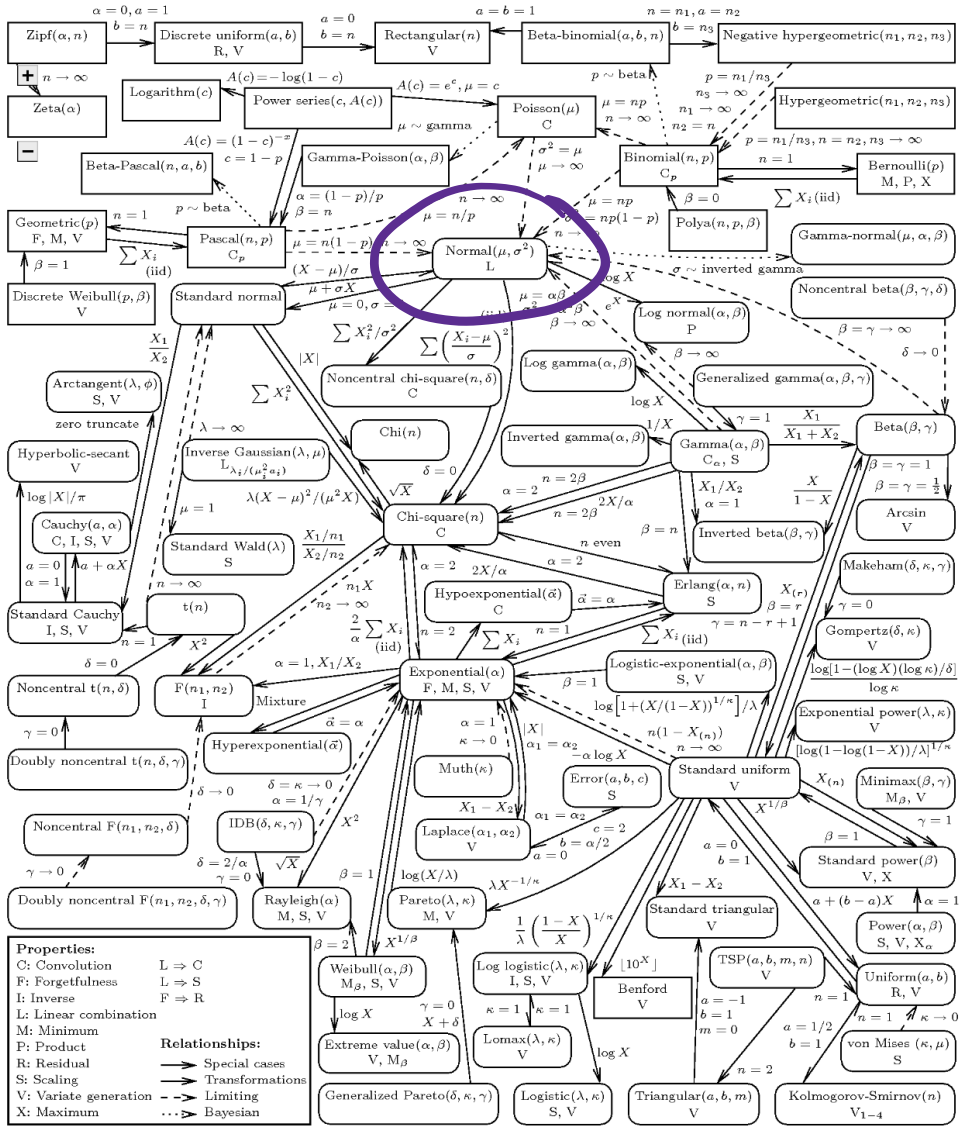


Discrete quantities

Probability mass function (pmf)

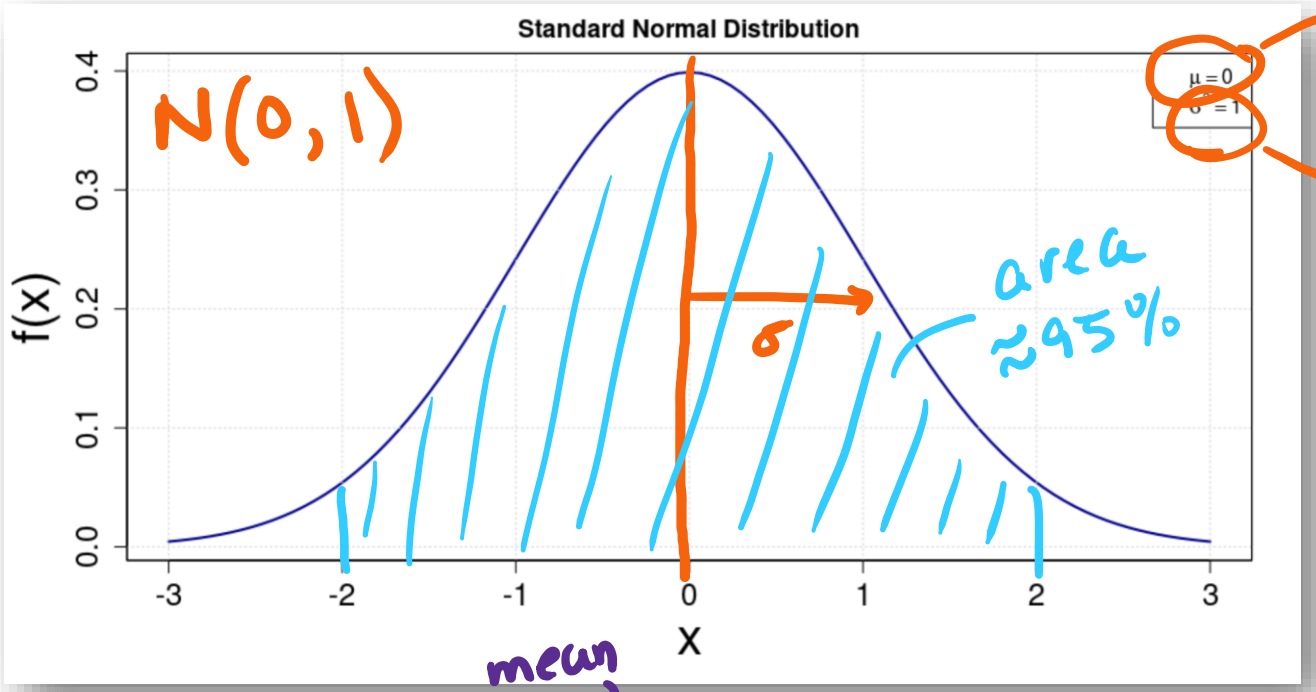


There are many univariate distributions!



http://www.math.wm.edu/~leemis/chart/UDR/UDR.html

The Normal Distribution



mean

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

probability distribution function

variance

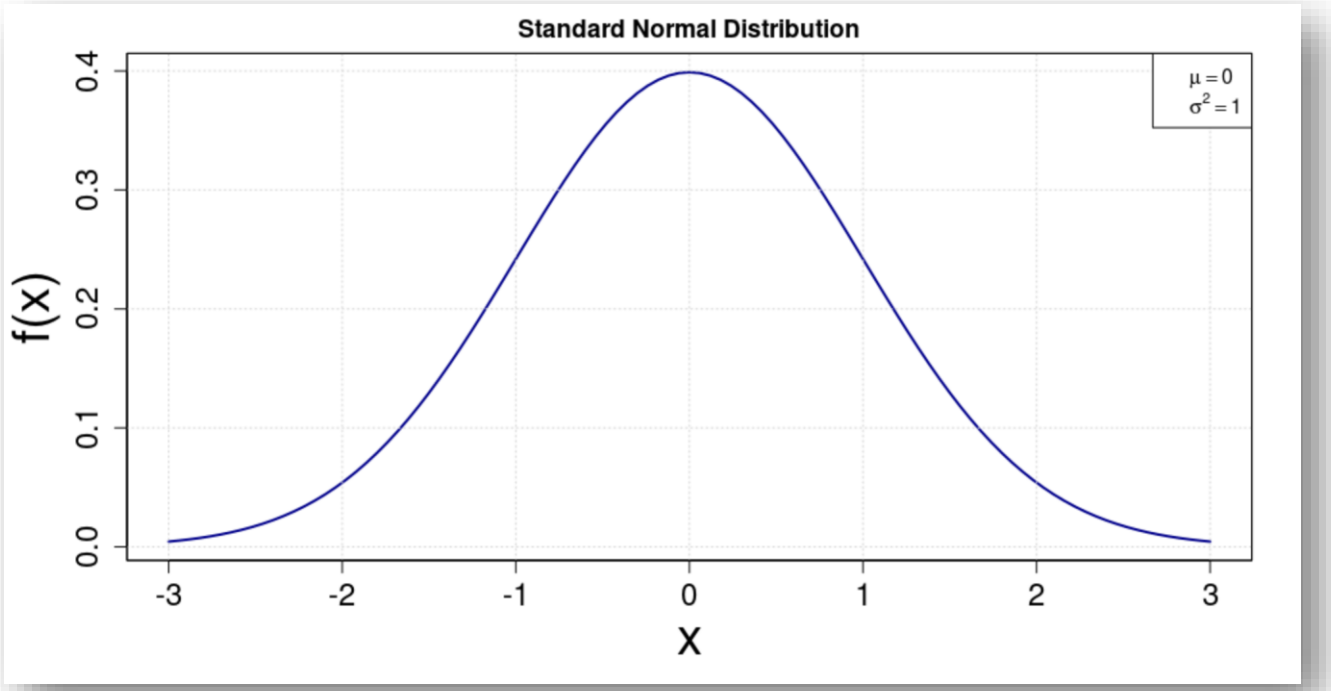
$N(\mu, \sigma)$

σ = standard deviation

$\sigma = \sqrt{\sigma^2}$

The Normal Distribution

$N(\mu, \sigma^2)$ ← variance

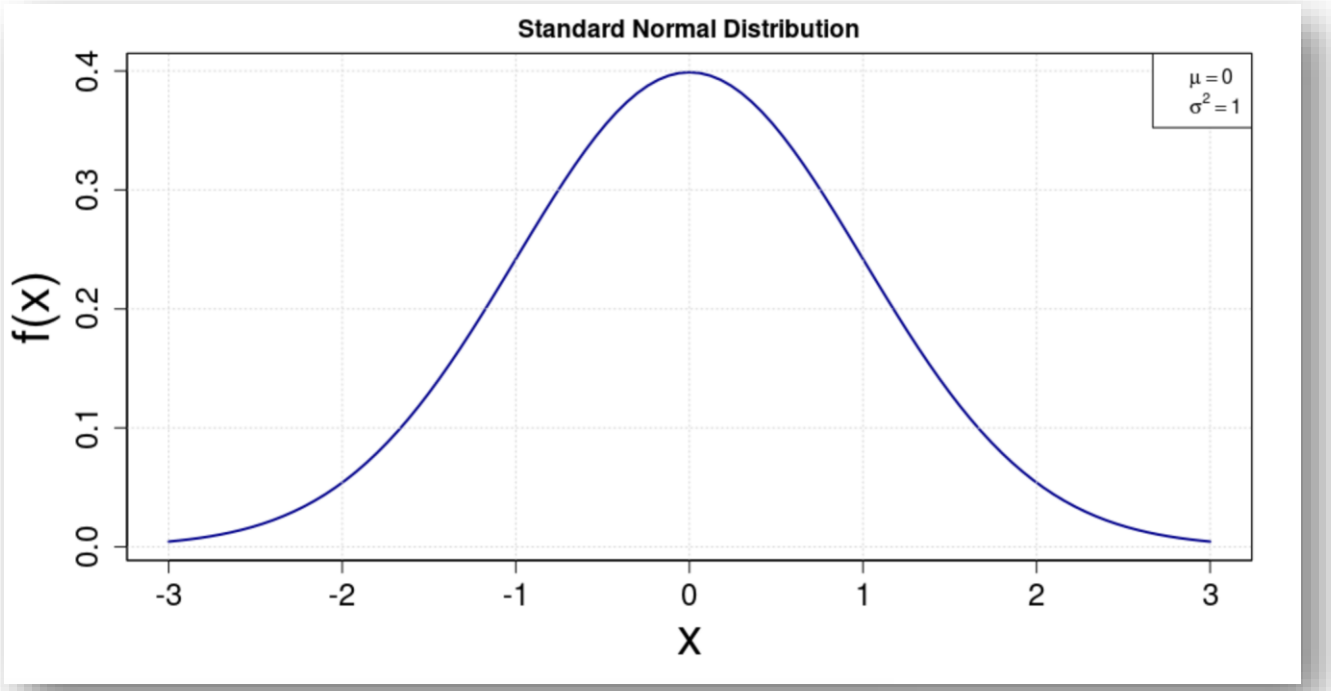


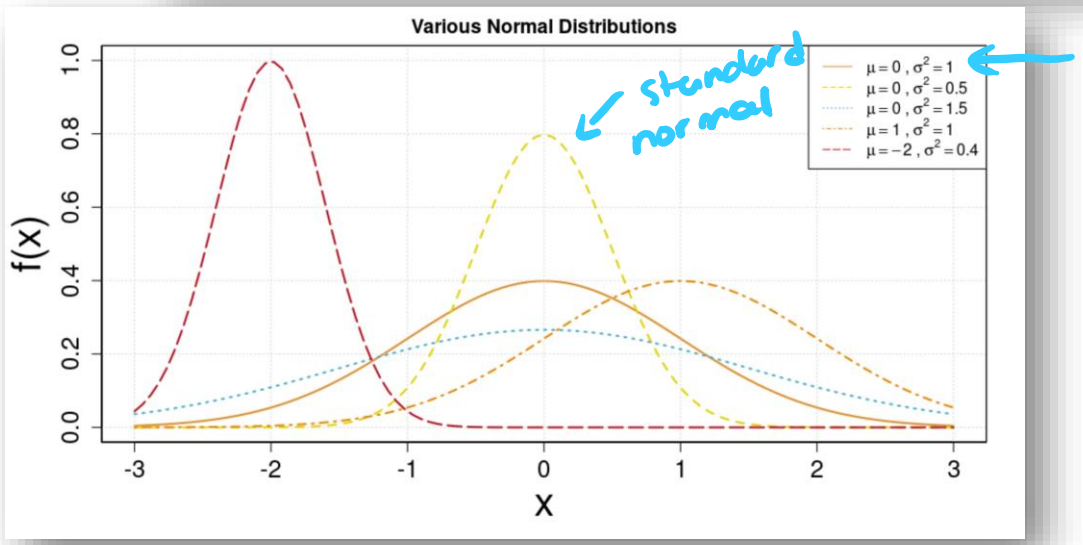
$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{(x-\mu)^2}{2\sigma^2}}$$

$N(\mu, \sigma)$ ← more common
s.d.

The Normal Distribution

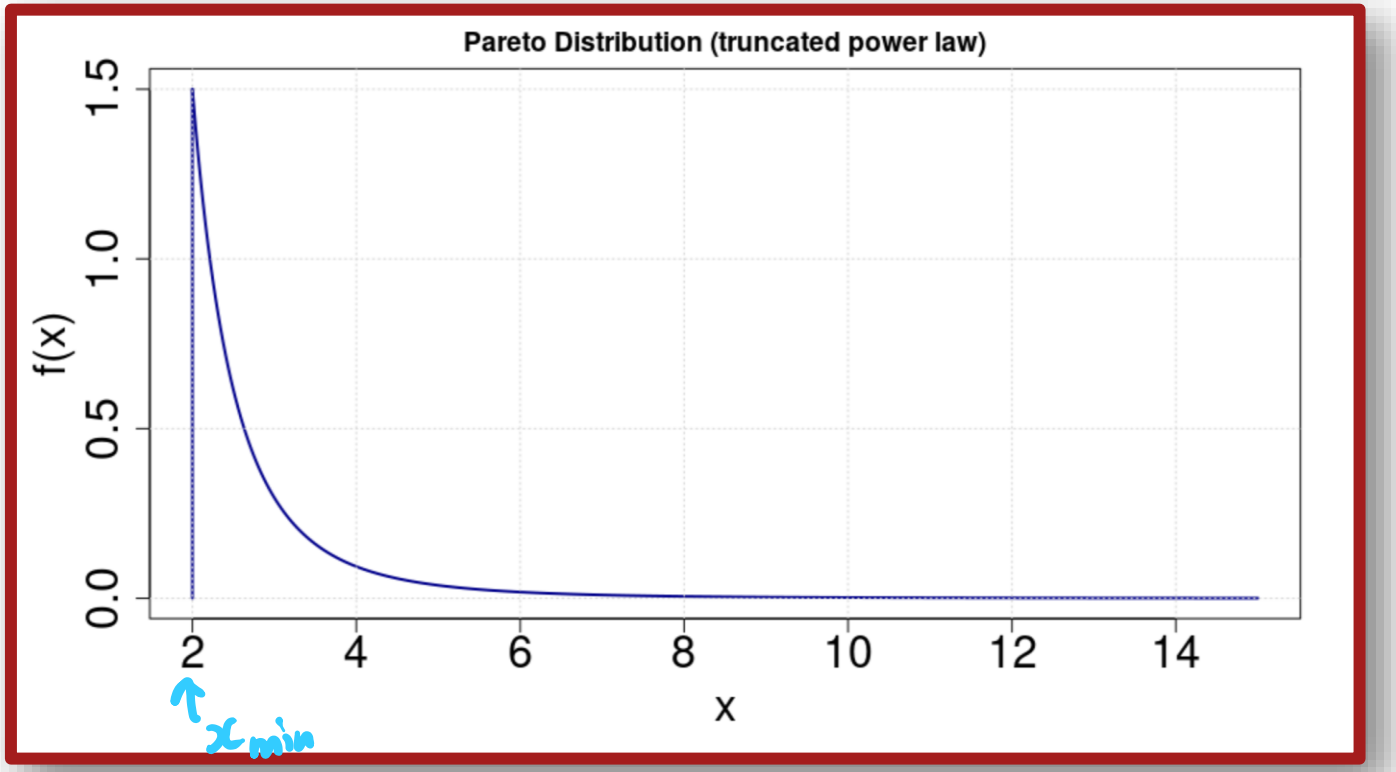
The mean and variance are all you need to plot the Normal





Example:

Pareto distribution (truncated power-law)

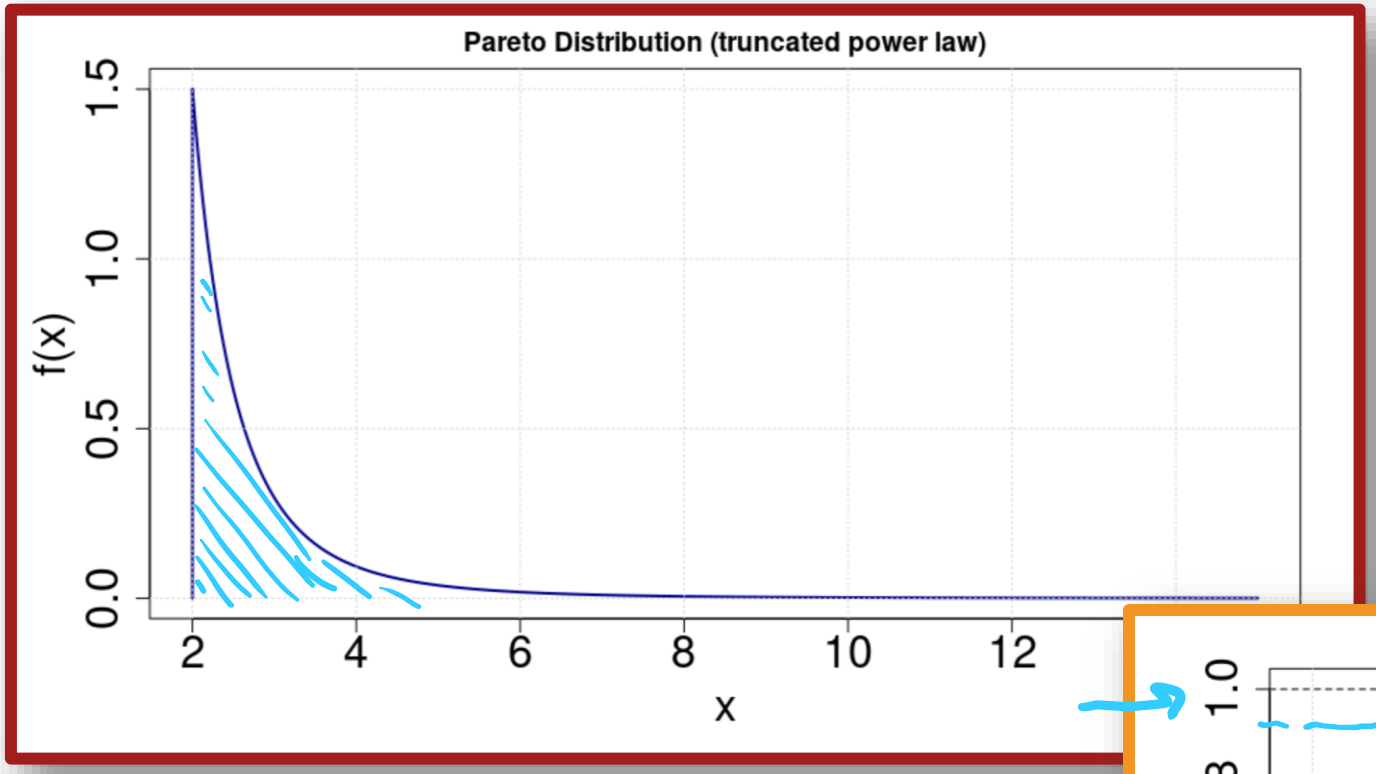


Probability distribution function (pdf)

$$f(x) = \frac{\alpha x_{\min}^{\alpha}}{x^{\alpha+1}}$$

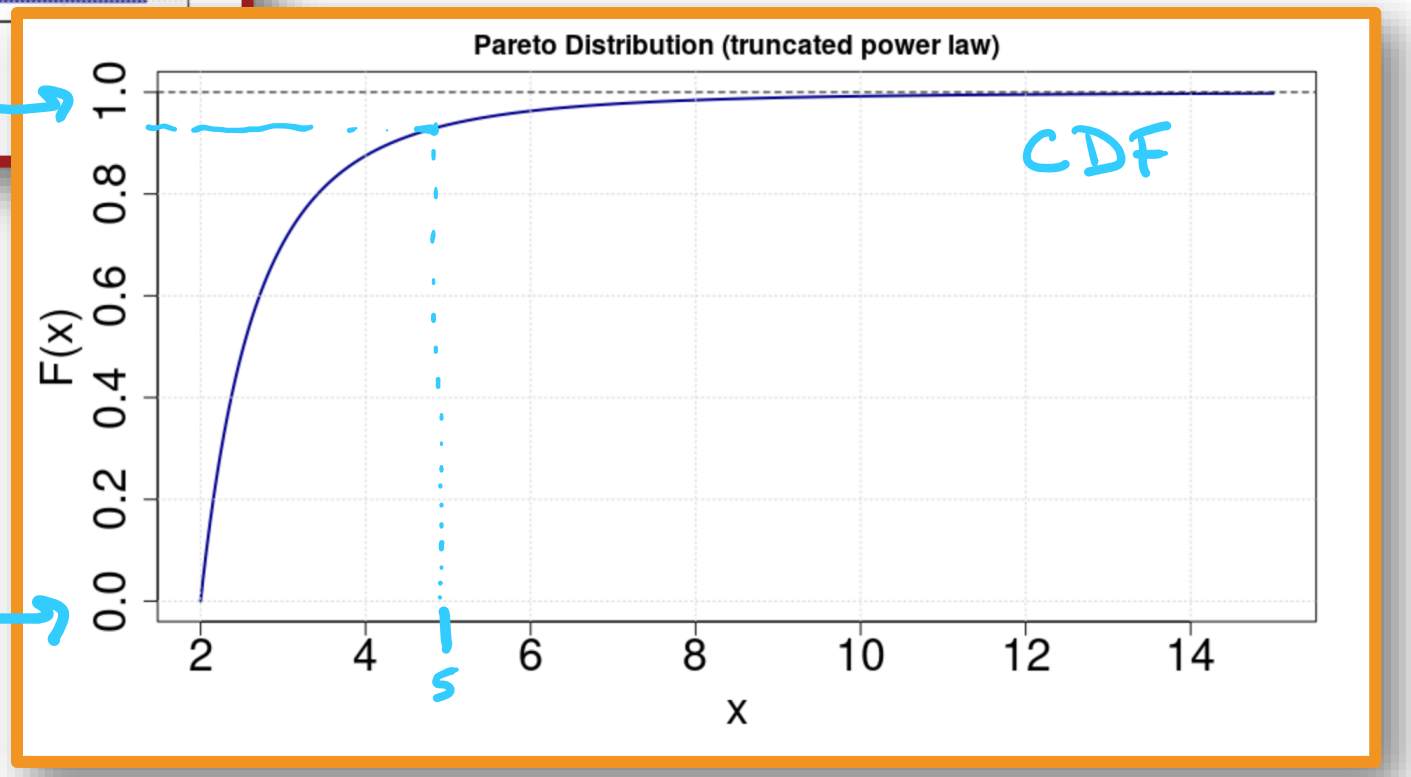
two parameters: α, x_{min}

Example:
Pareto distribution (truncated power-law)



$$F(x) = P(X \leq x) = 1 - \left(\frac{x_{\min}}{x}\right)^{\alpha}$$

Cumulative distribution function (cdf)



Probability distribution function (pdf)

$$f(x) = \frac{\alpha x_{\min}^{\alpha}}{x^{\alpha+1}}$$

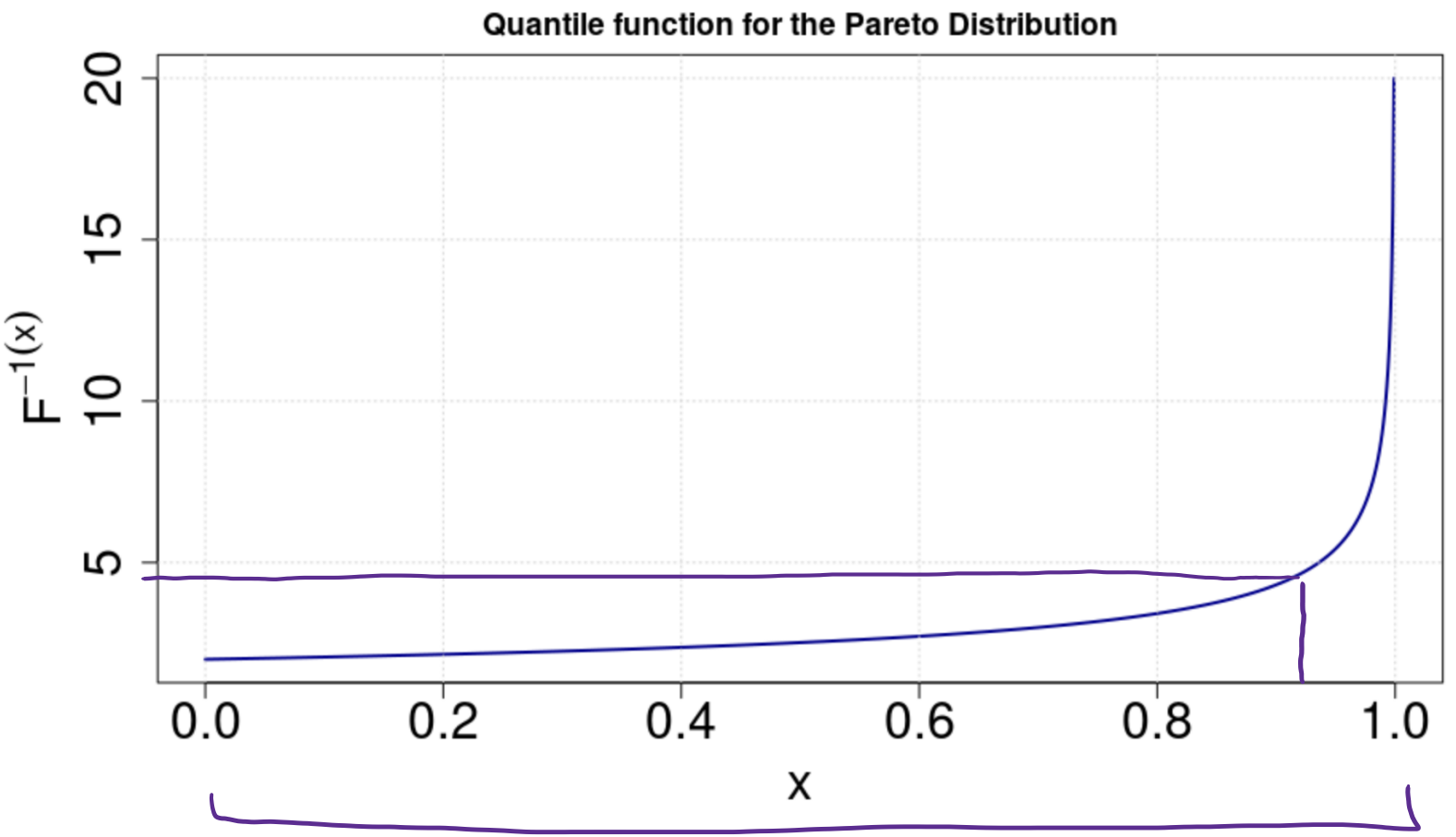
$F(x) = \int f(x)\,dx$

CDF PDF

Quantile Function

→ $F^{-1}(x)$

- This is the inverse of the cumulative distribution function (cdf)



CDF for Pareto

$$F(x) = 1 - \left(\frac{x_{min}}{x}\right)^{\alpha}$$

can invert this (solve for x) to get $F^{-1}(x)$

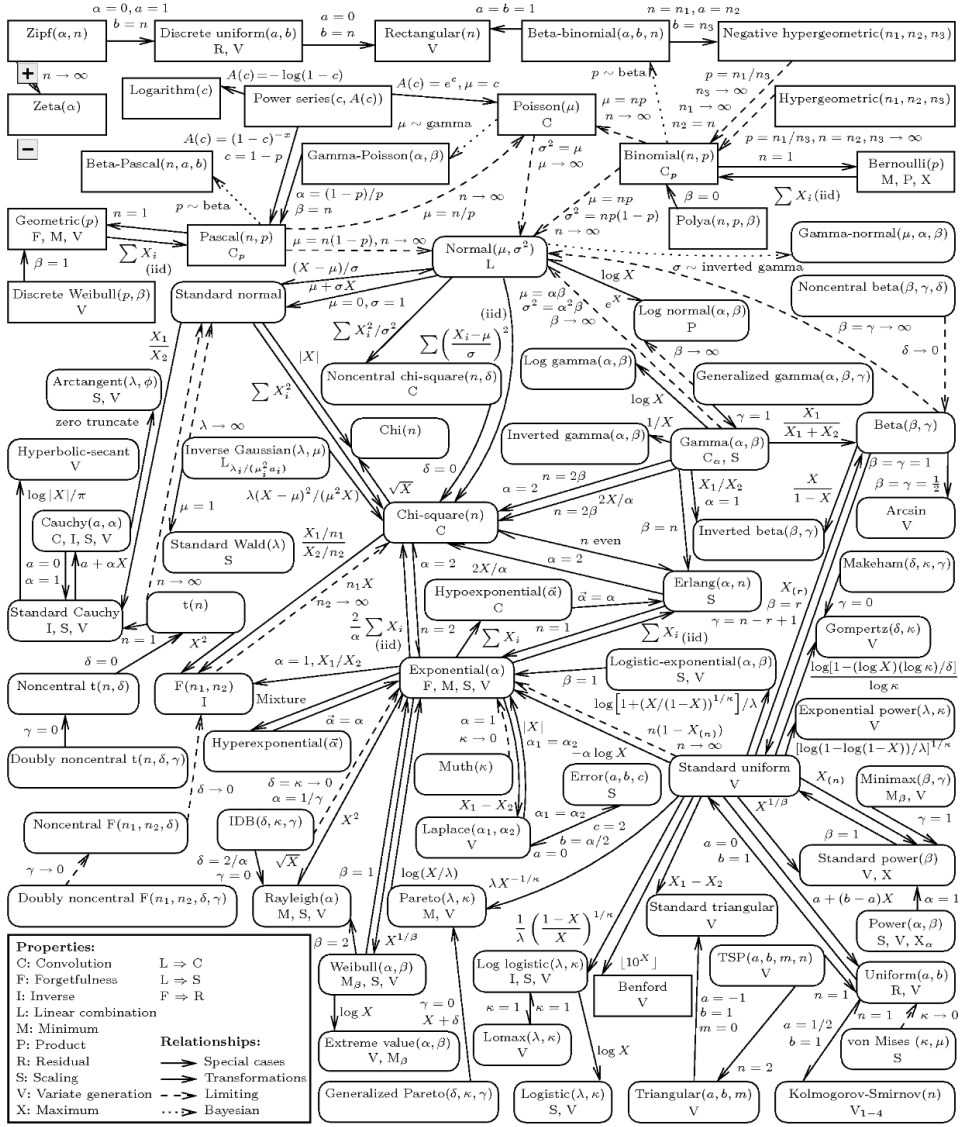
draw random #'s between 0 and 1

HOT TIP

This chart has a pdf file for every distribution! Check out the link below

There are many univariate distributions!

Demo Time!
Look at the data



http://www.math.wm.edu/~leemis/chart/UDR/UDR.html

Random Variables

Random Variable

- A random variable X is a *function* that maps an *outcome* to a *real number*
  - e.g., Let's say we decide to flip a coin repeatedly, and each time we flip it we record whether we get heads or tails with a 1 or a 0 respectively.

- In other words, X is a **function**. Little x represents the data --- *realizations* of that random variable.

$$X \begin{cases} 1 & \text{heads} \\ 0 & \text{tails} \end{cases}$$

$x \rightarrow$ a particular value/ outcome

A Random Variable follows a distribution

The standard statistics notation to show what distribution a random variable follows is:

$$X \sim N(\mu, \sigma^2)$$

distributed as

For example, we might assume that are data x (e.g. the photon counts from a star) follows a Poisson distribution

$$X \sim Pois(\lambda)$$

x_1
x_2
x_3
x_4
$\vdots$

$$X = \begin{cases} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{cases}$$

$$X = \begin{cases} 2 \longrightarrow \text{success} \rightarrow 1 \\ \text{not } 2\ (1,3,4,5,6) \rightarrow \text{failure} \rightarrow 0 \end{cases}$$

$X \sim$ Bernoulli trial

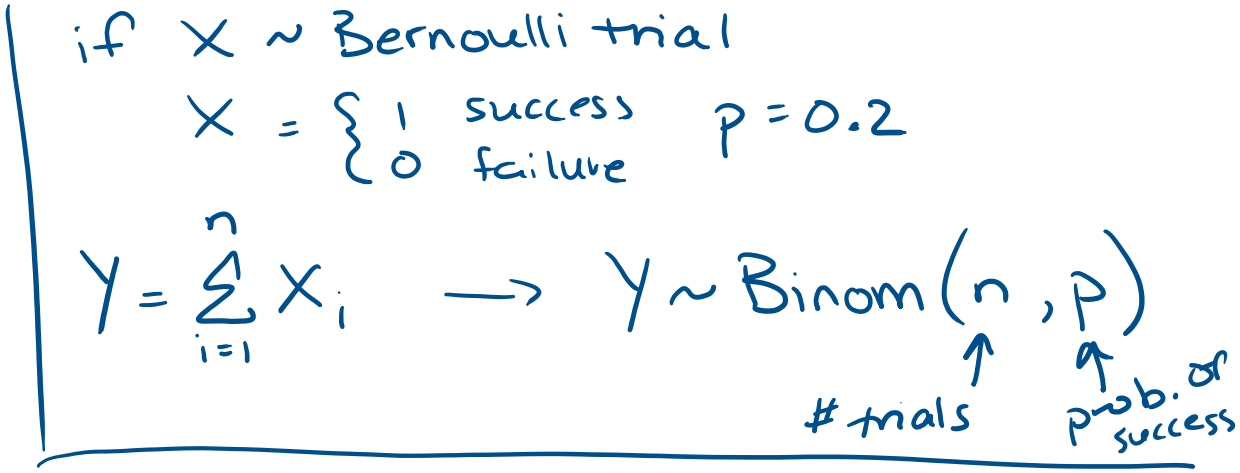
A Random Variable follows a distribution

The standard statistics notation to show what distribution a random variable follows is:

$$X \sim N(\mu, \sigma^2)$$

For example, we might assume that are data x (e.g. the photon counts from a star) follows a Poisson distribution

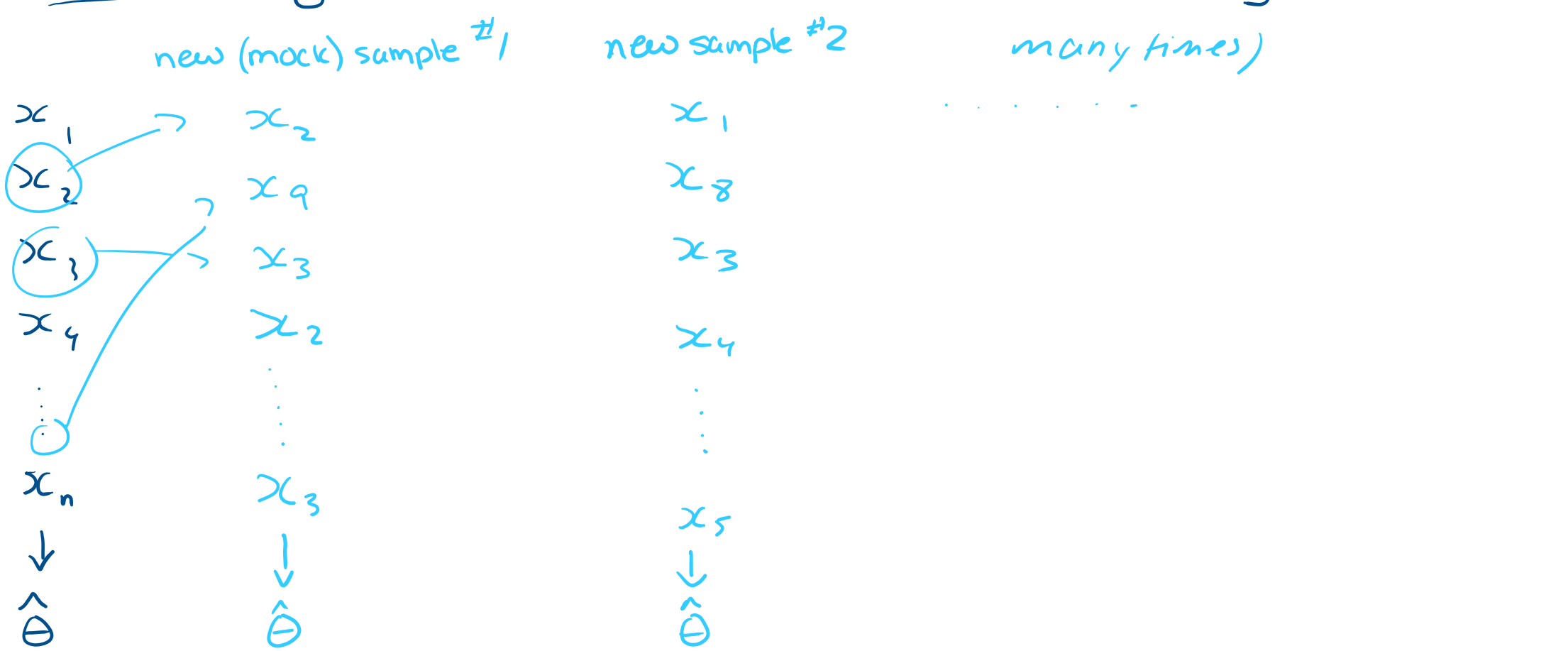
$$X \sim Pois(\lambda)$$



HOT TIP

The Poisson distribution is often used to describe *counts* of events in an interval of time or space. It is a *discrete probability distribution.*

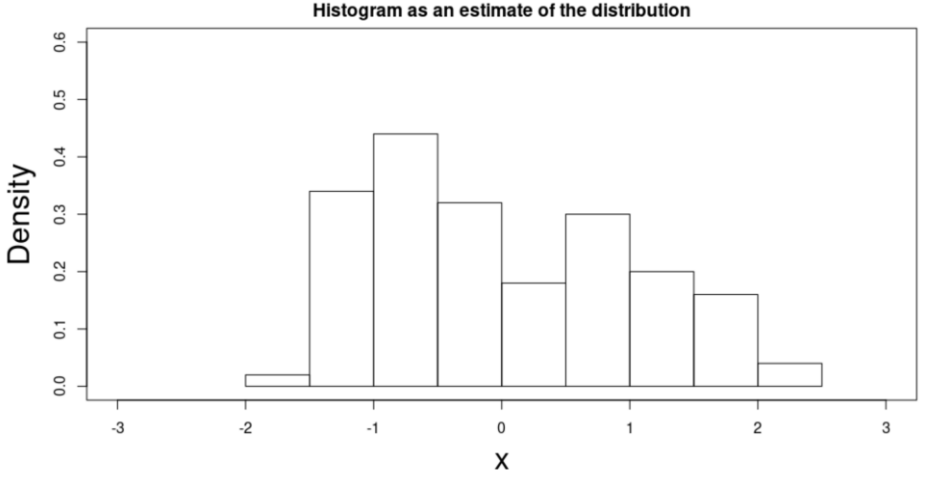
Bootstrapping —> when you don't know the underlying distribution

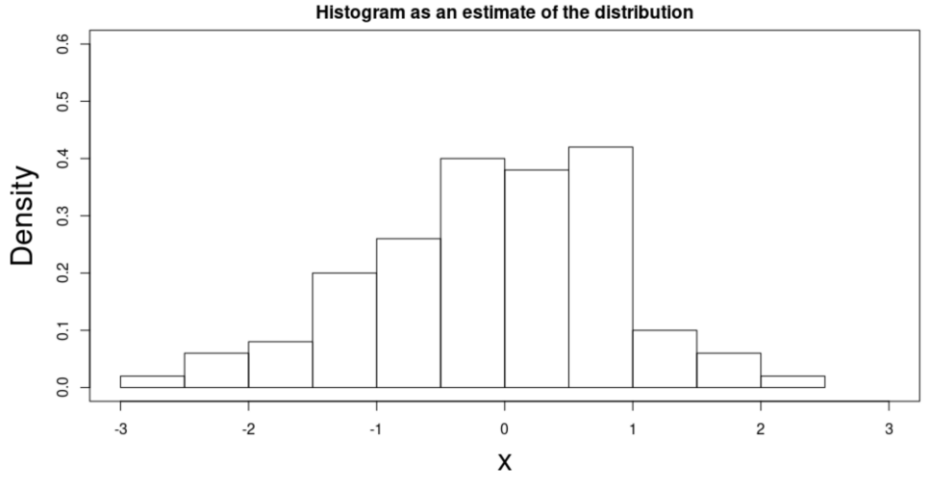


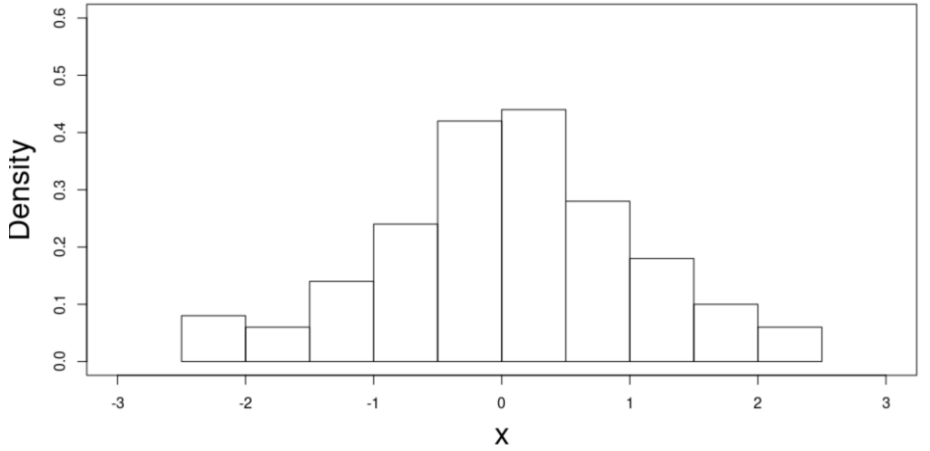
Randomness in Data

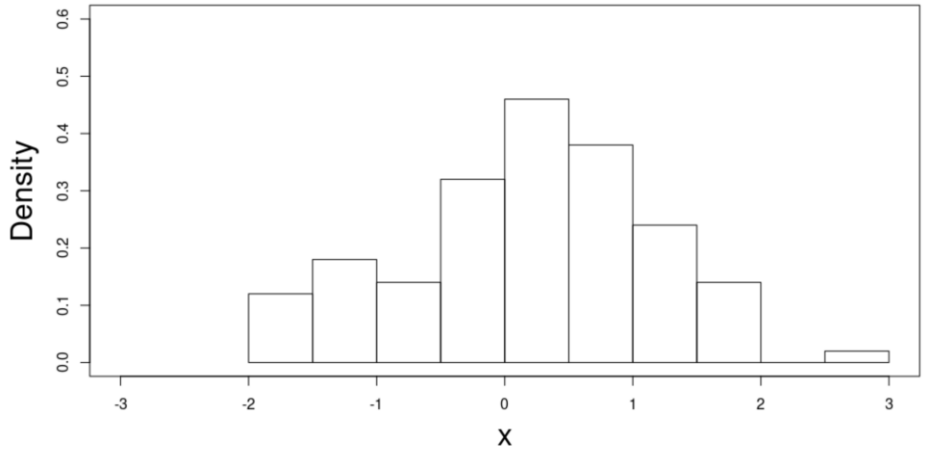
Randomness in Data

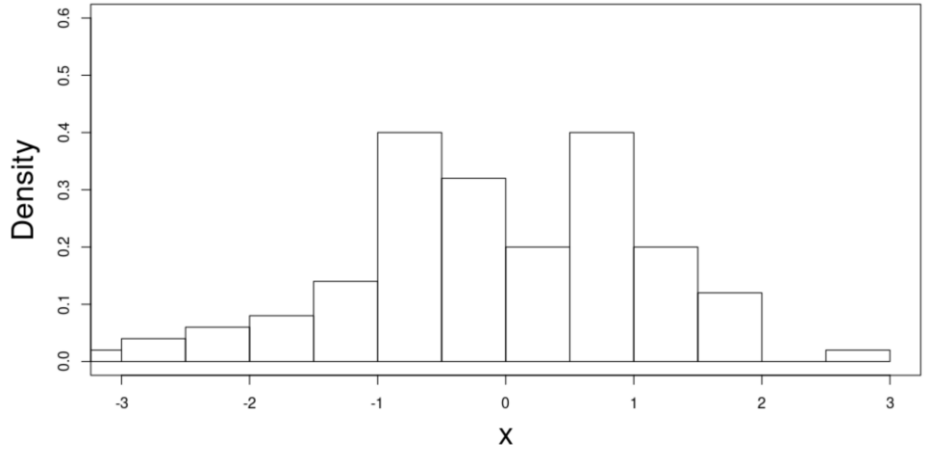
- All these histograms were generated from 100 draws from a standard normal







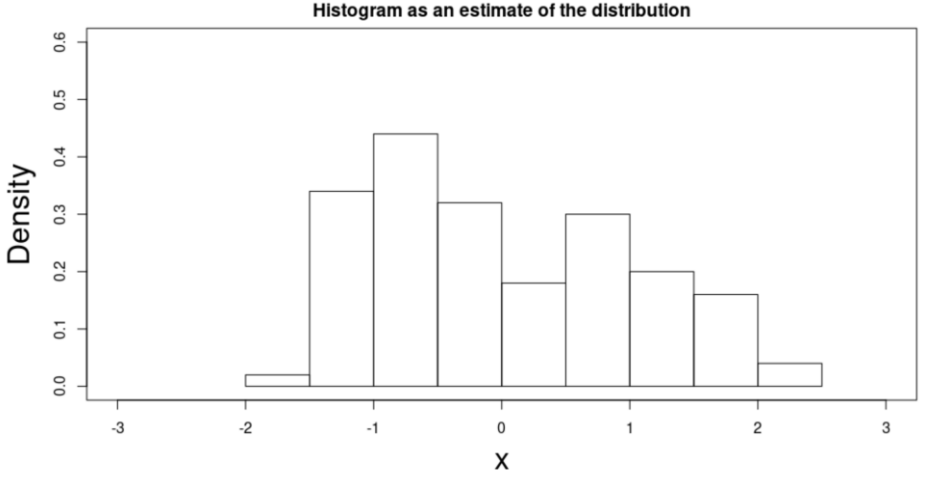


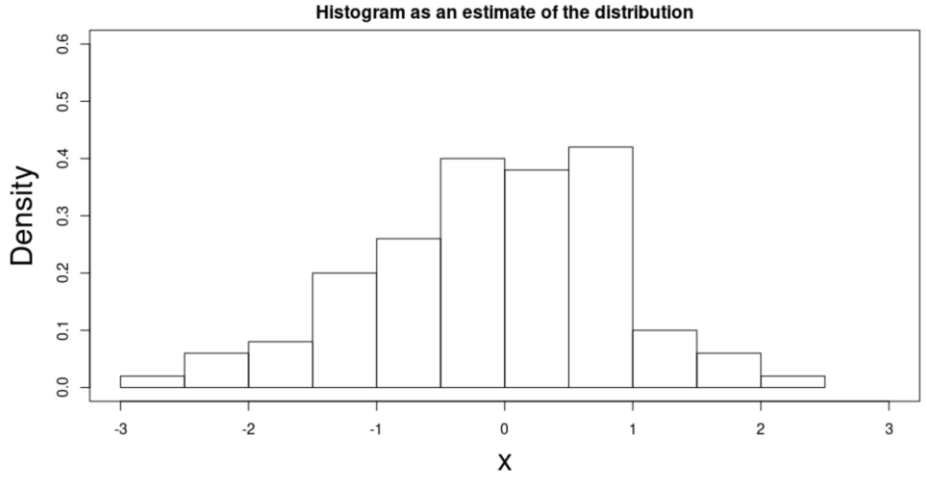


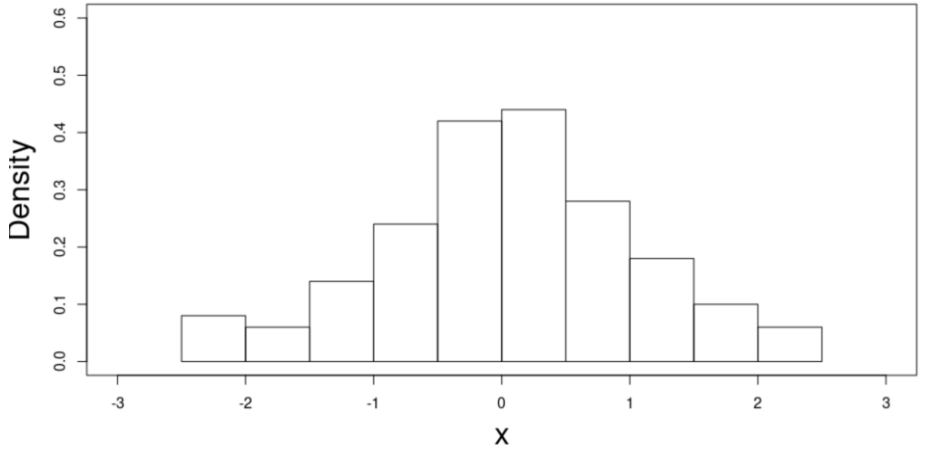
Randomness in Data

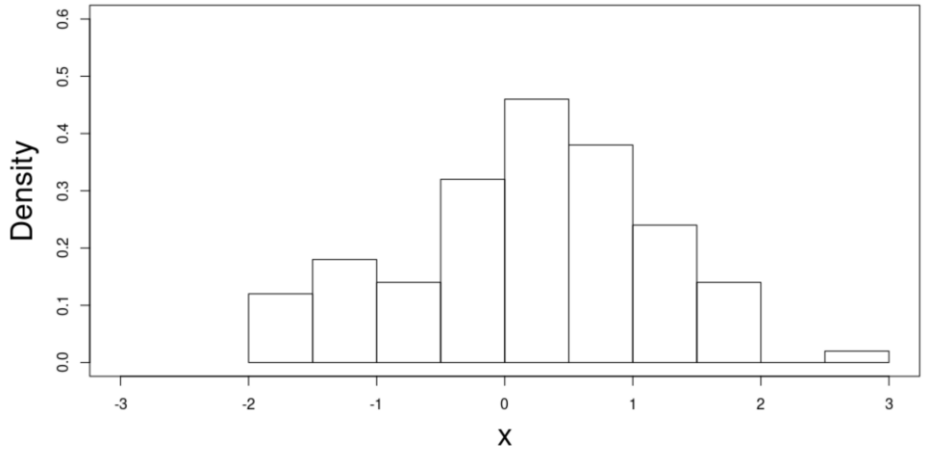
- All these histograms were generated from 100 draws from a standard normal

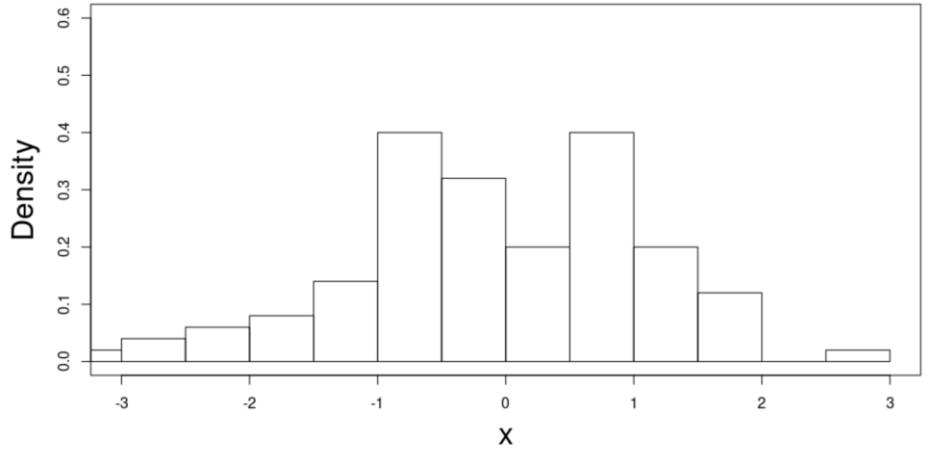
From the data, we can try to *estimate* the true distribution. We can also try to estimate the underlying parameters of the distribution. These estimates are **random variables.**







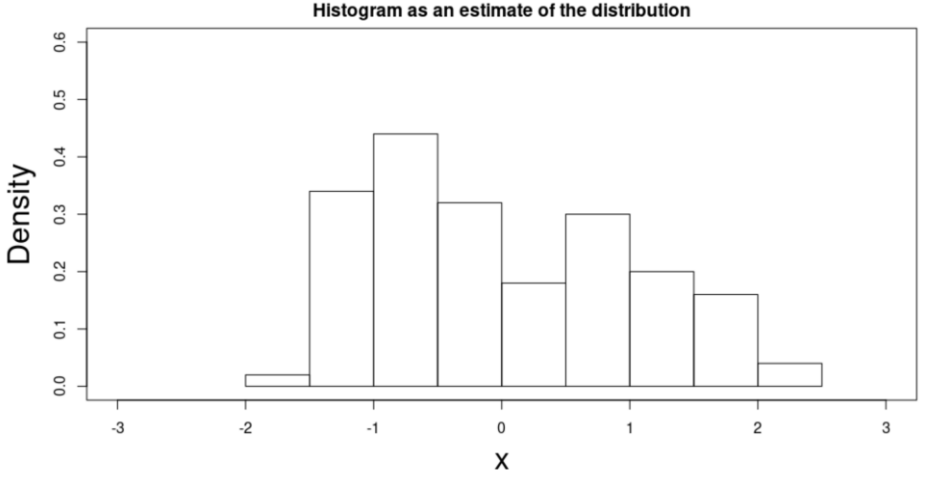


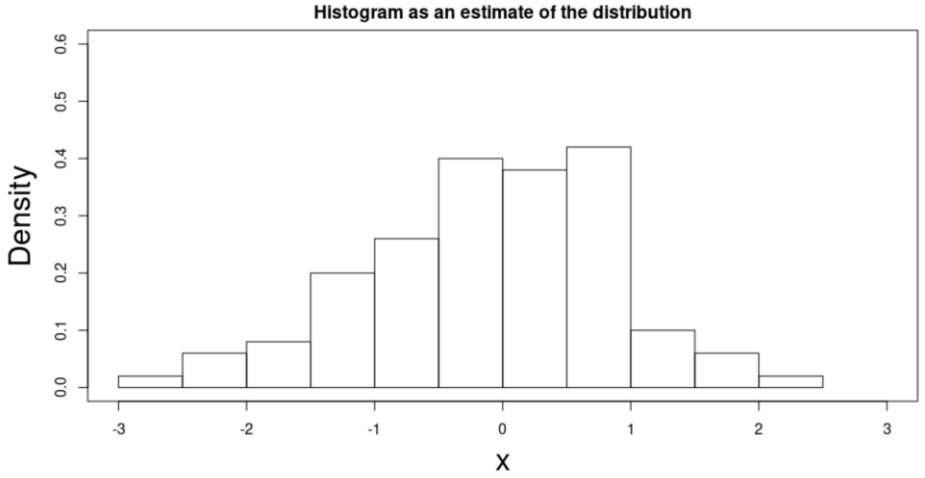


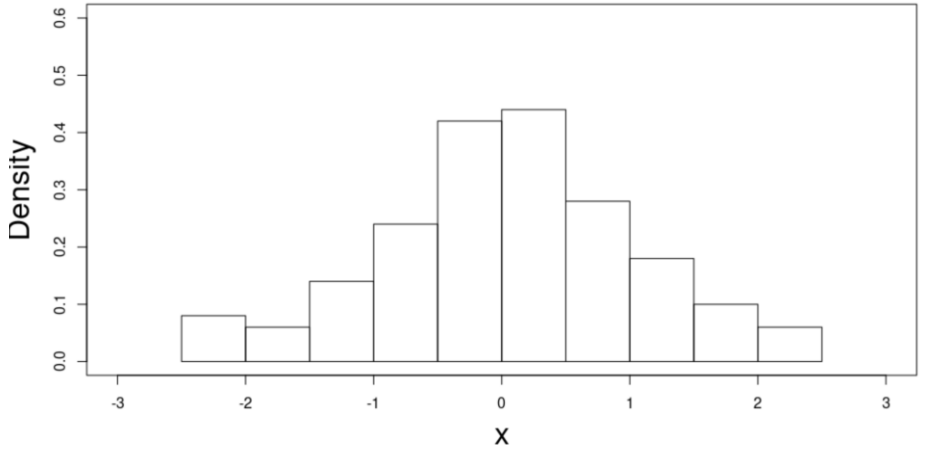
Randomness in Data

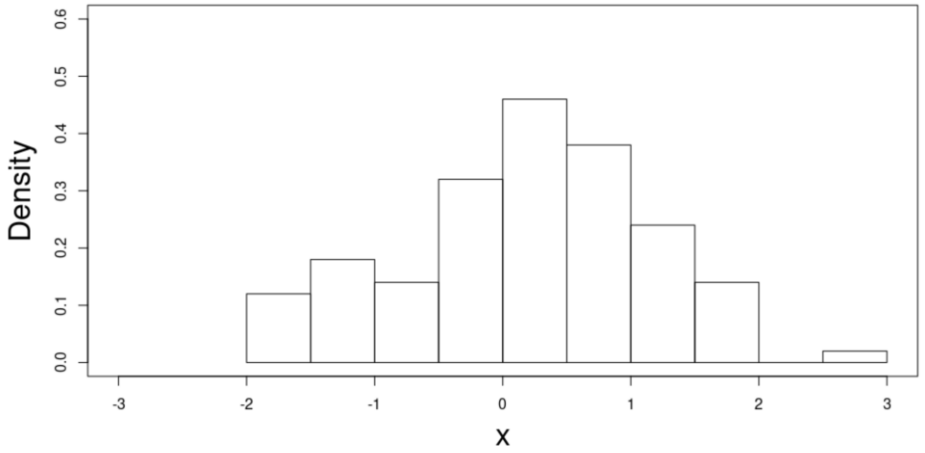
- All these histograms were generated from 100 draws from a standard normal

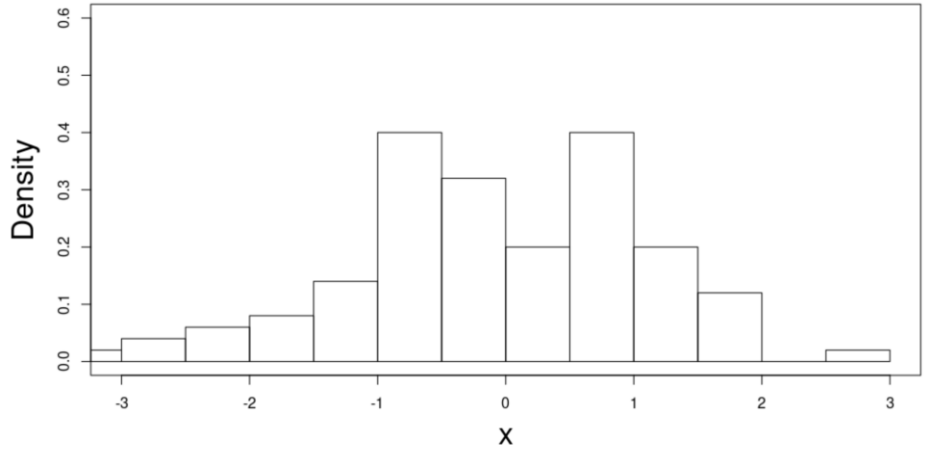
From the data, we can try to *estimate* the true distribution. We can also try to estimate the underlying parameters of the distribution. These estimates are **random variables.**











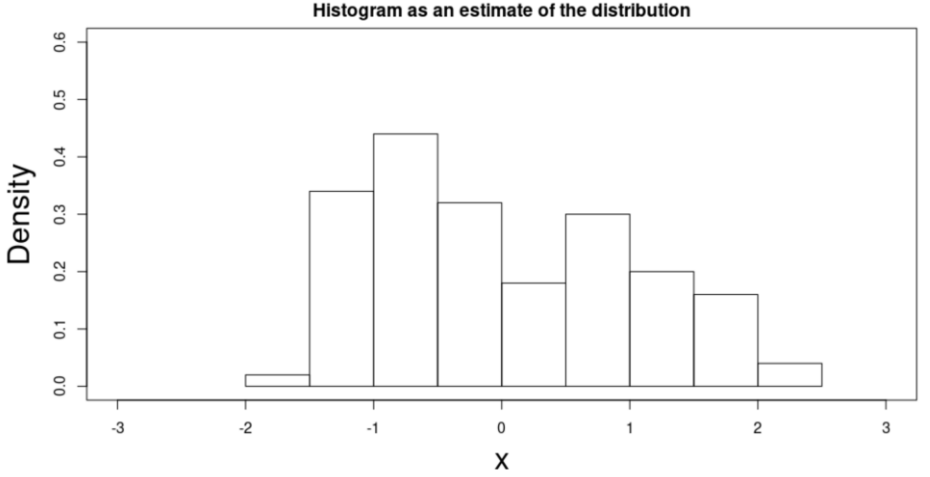
Demo Time!
R-Studio

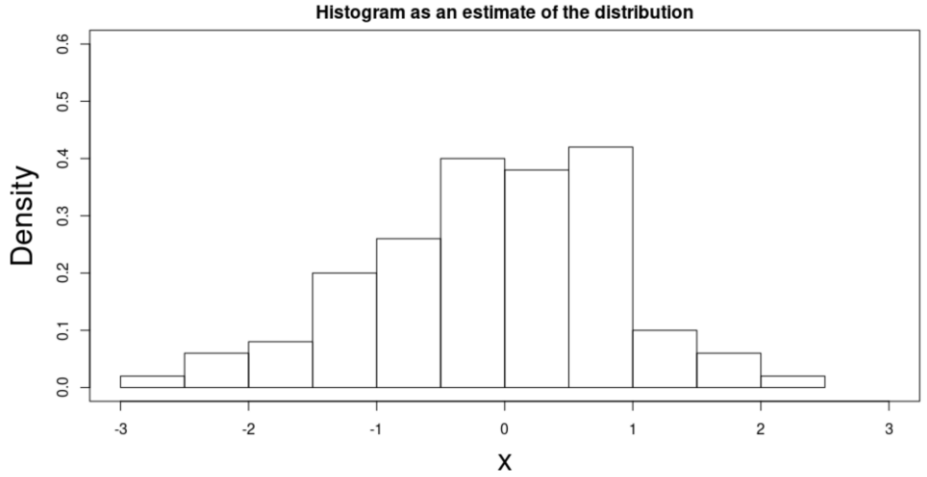
Randomness in Data

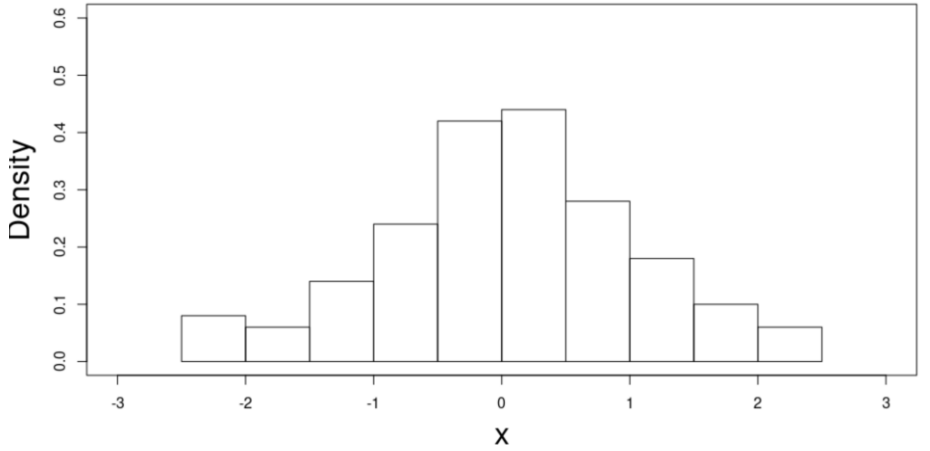
- All these histograms were generated from 100 draws from a standard normal

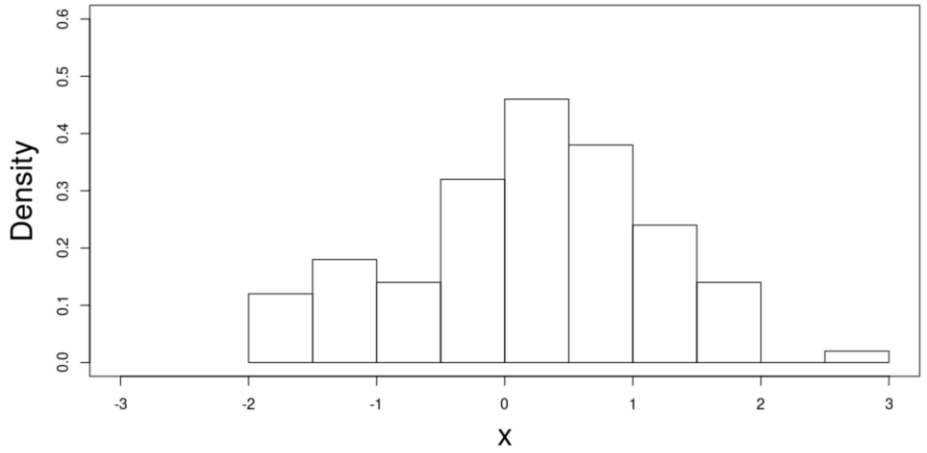
COOL CATCH

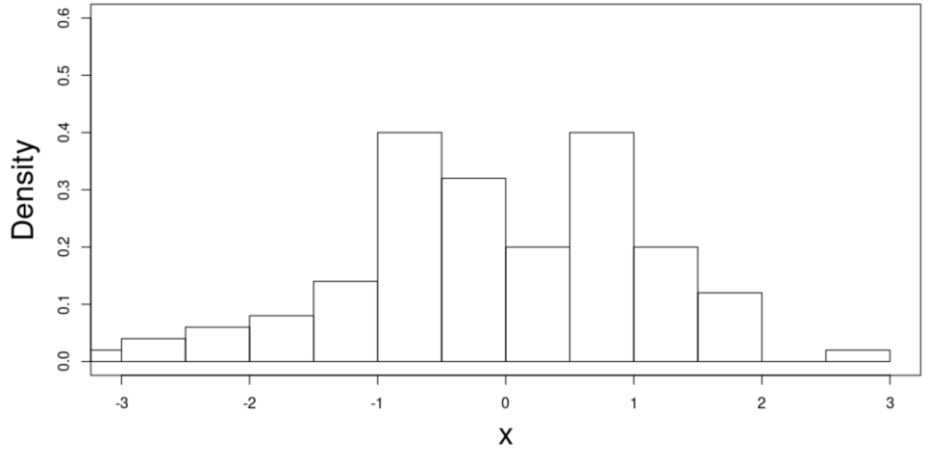
Human eyes like to look for patterns/trends. Don't mistake randomness for a signal.











Randomness in Data

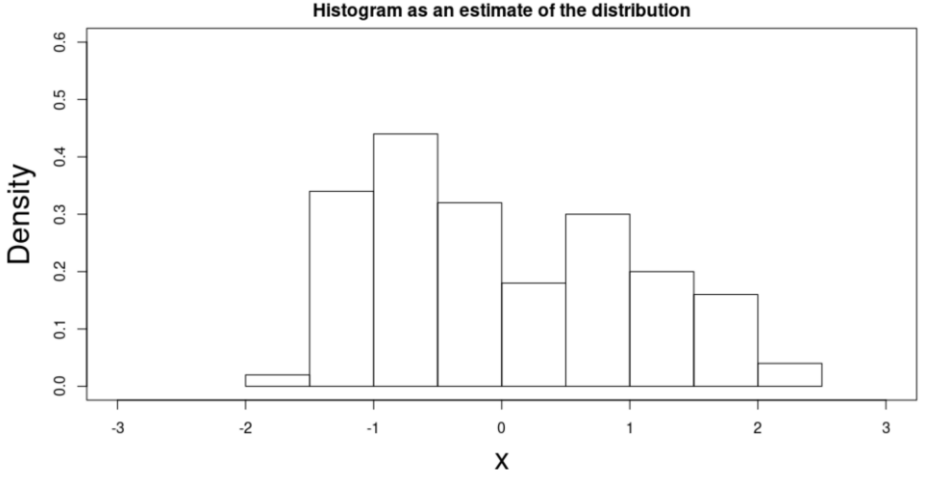
- All these histograms were generated from 100 draws from a standard normal

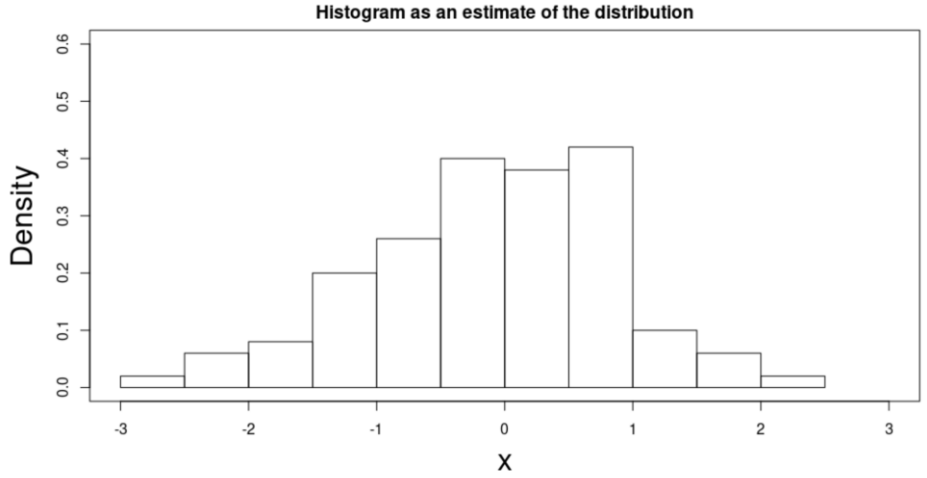
COOL CATCH

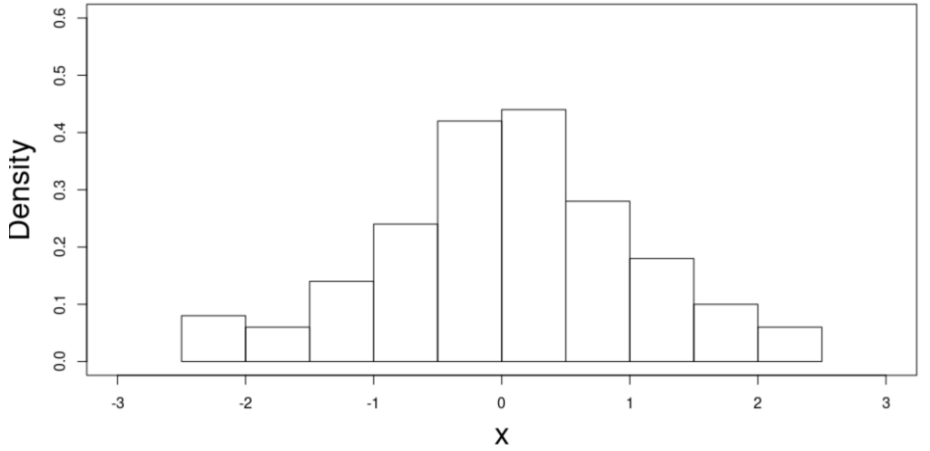
Human eyes like to look for patterns/trends. Don't mistake randomness for a signal.

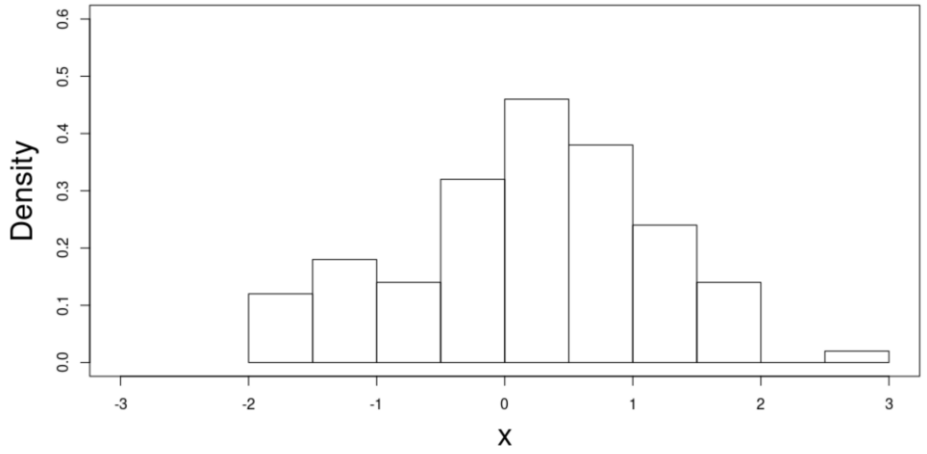
HOT TIP

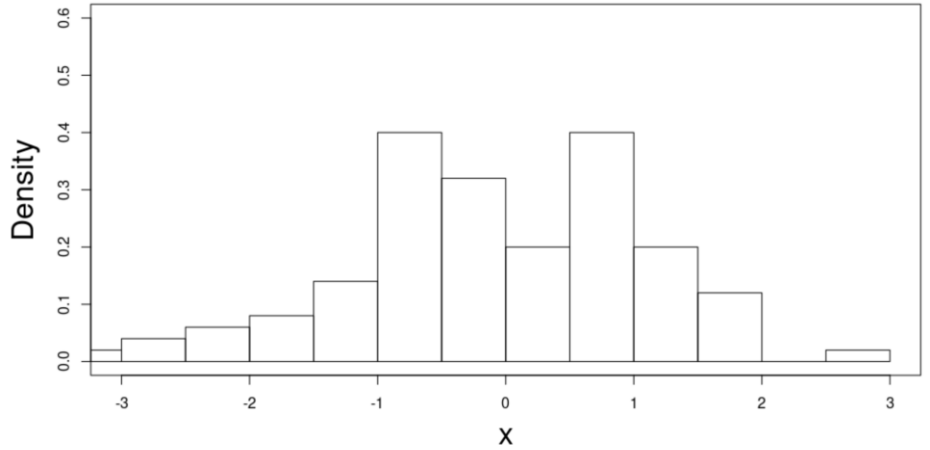
Sometimes we want to *generate* randomness to create mock data that looks "real"











Sampling from a distribution

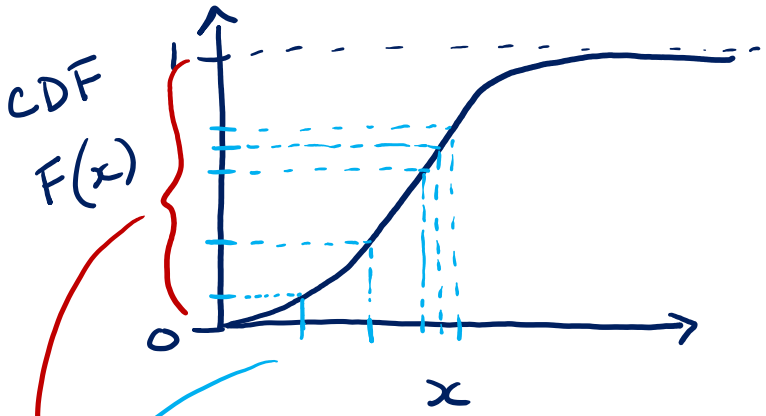
Sampling from a distribution (two basic approaches)

Inverse cdf Method

inverse CDF

$F^{-1}(x)$

- First choice if the inverse cdf is tractable



draw random values between 0 and 1

drawing from $U(0,1)$, and pass to $F^{-1}(x)$

"uniform distribution"

a bunch of x values that have f(x) → Pareto distribution α x_{min}

Accept/Reject Algorithm

- Useful when you can't write down the inverse cdf

y_1^*, $f(x^*)$

y_2^*, $f(x_2^*)$

BREAK!

Estimates of Distributions

Visualizing Empirical Distributions

- Histograms (in frequency or relative frequency)
- Boxplots
- Kernel Density Estimators
- Empirical cumulative distribution functions (ecdfs)
- Bar charts, stacked bar chart, mosaic plots, contingency tables, ...

Visualizing Empirical Distributions

- Histograms (in frequency or relative frequency)
- Boxplots
- Kernel Density Estimators
- Empirical cumulative distribution functions (ecdfs)
- Bar charts, stacked bar chart, mosaic plots, contingency tables, ...

Estimating Parameters of Distributions

- Method of moments
- Maximum Likelihood Estimators
- Bayesian inference

Box Plots

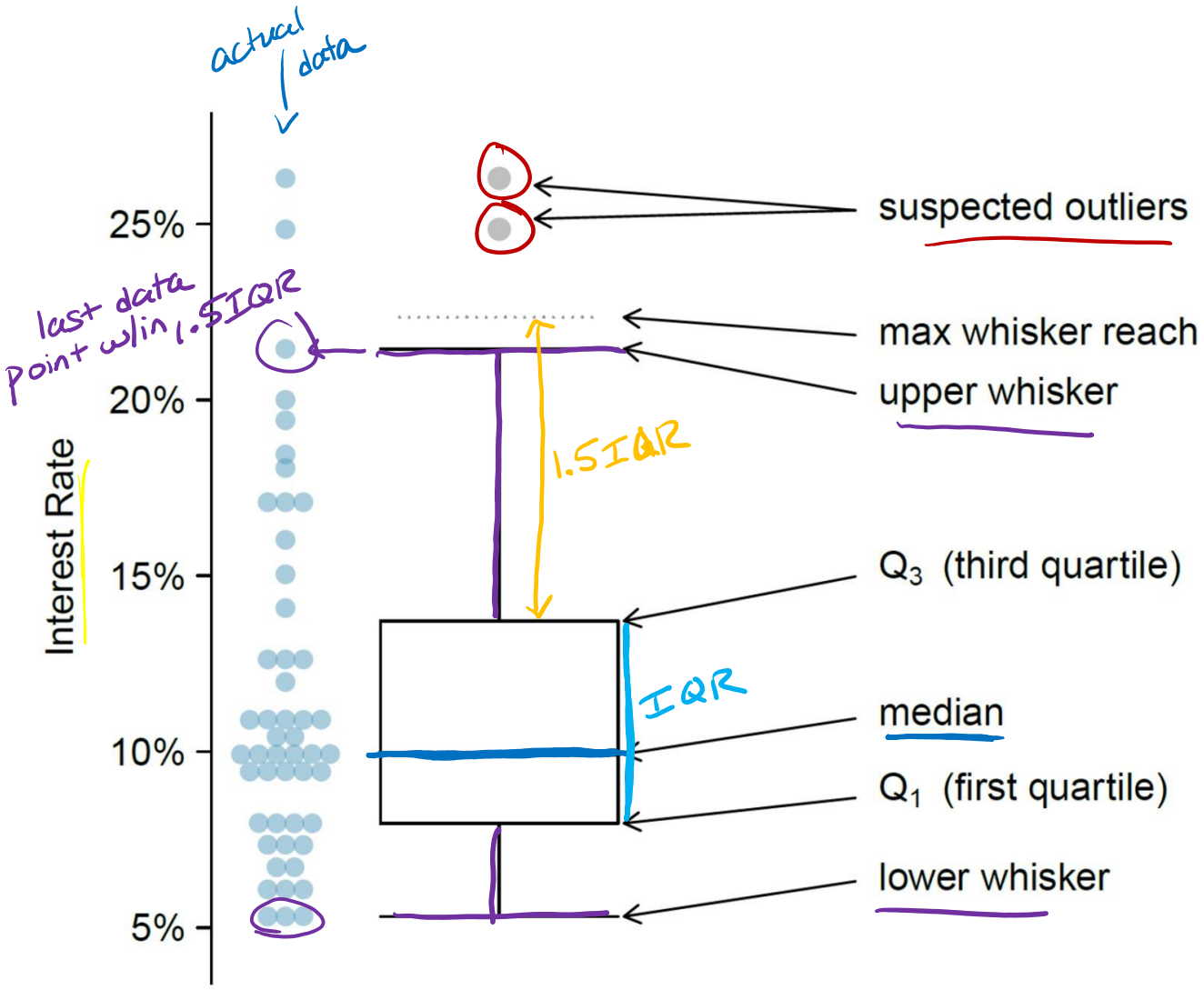
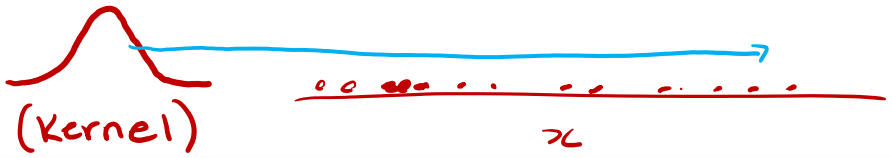


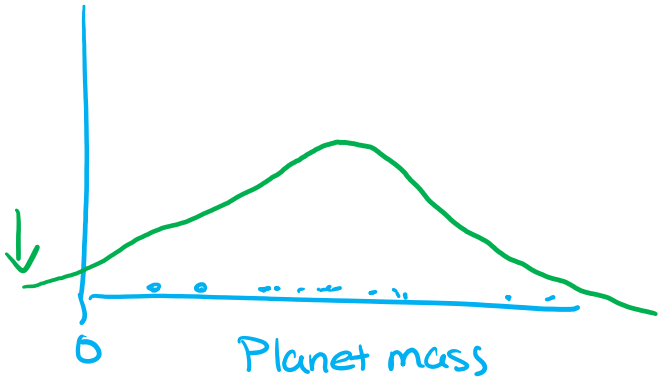
Figure 2.10, OpenIntro (4th ed.)

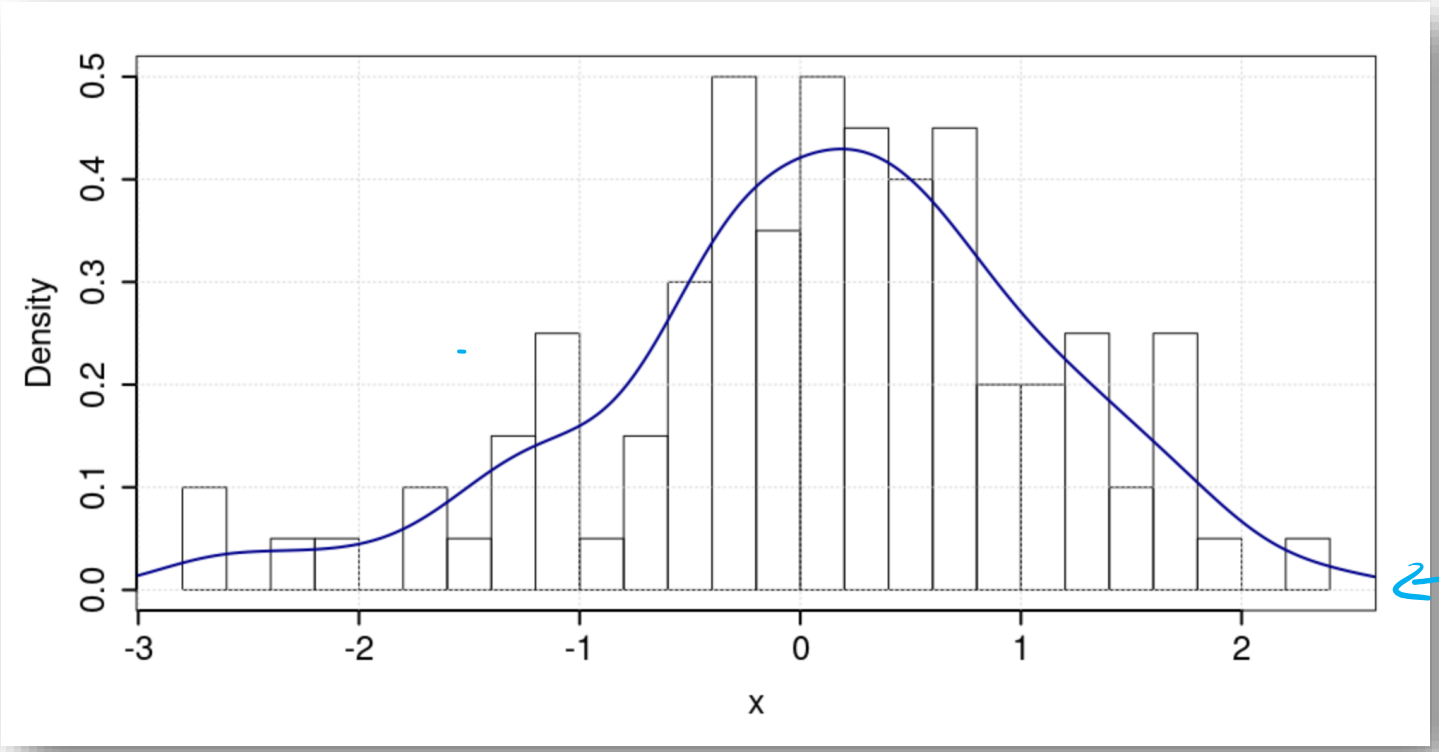
- Building a box plot:
  - Find the median first
  - Draw a rectangle that shows the interquartile range (IQR) → contains 50% of data
  - Extend the whiskers out to the furthest data point that is still within 1.5xIQR
  - Show the individual points that are outside the whiskers

Kernel Density Estimates

- Less sensitive to bin size, bin choice



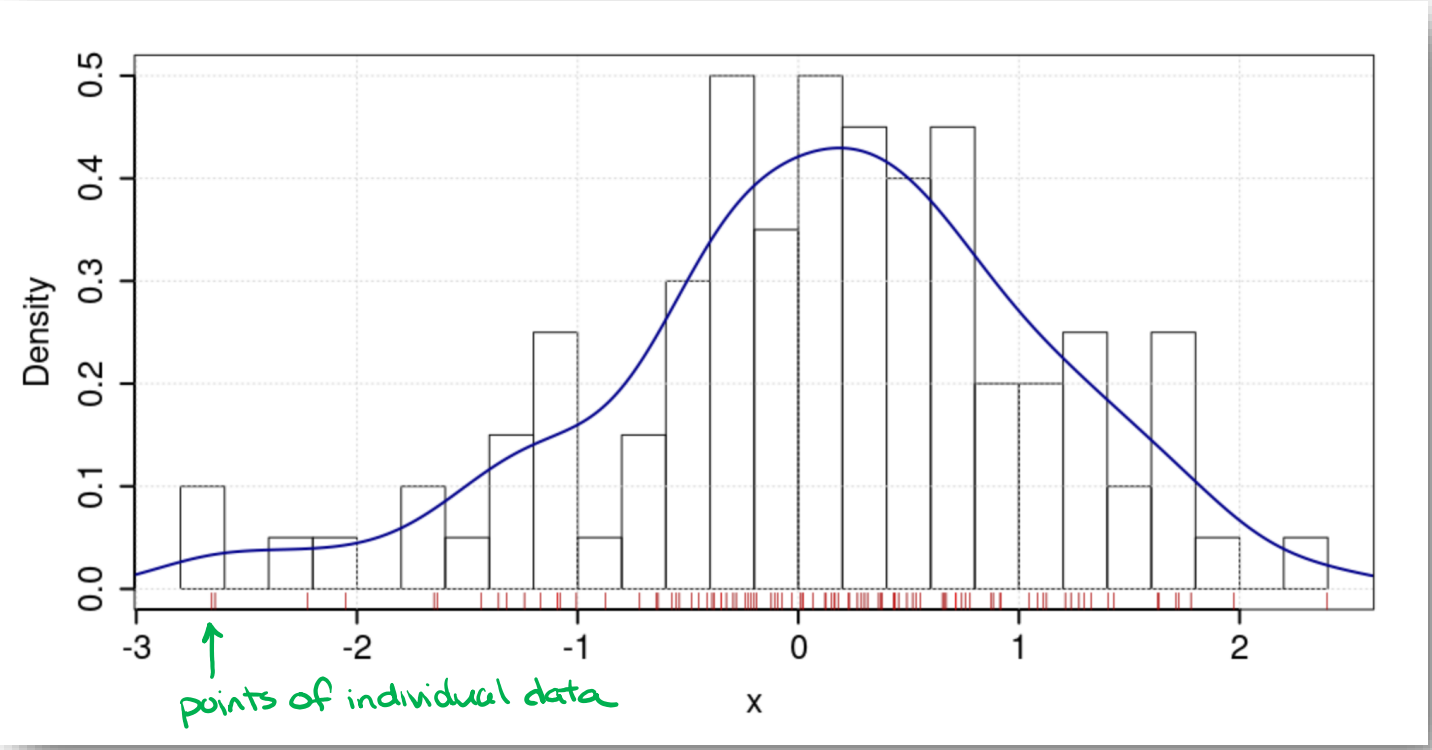




smooth estimate of distribution

Kernel Density Estimates

- Less sensitive to bin size, bin choice
- Helpful to add a "rug"



Kernel Density Estimates

- Less sensitive to bin size, bin choice
- Helpful to add a "rug"
- (really quick to plot in R)

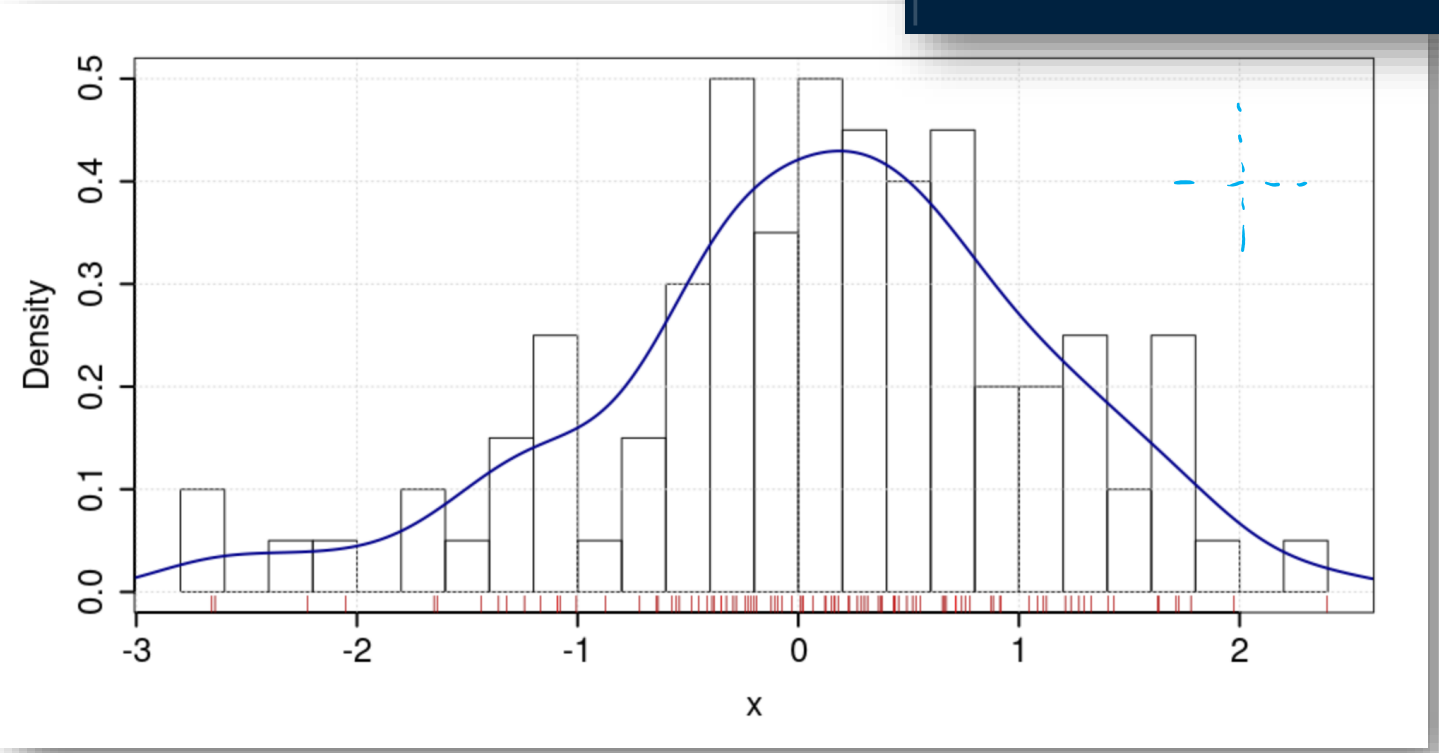
```
par(mar=c(5,5,2,2))
hist(x, breaks = 20, freq = FALSE, lwd=2, main="", cex.lab=1.5, cex.axis=1.5)
grid()
lines(density(x), col="darkblue", lwd=2)
box()
rug(x, col="red")
```

plots the KDE estimate

histogram

background grid

calculates KDE



HOT TIP

You do not have to import any modules or packages to make these kinds of plots in R. The functions are just there!

Summary Statistics

Five-Number Summary

You almost get all five from a boxplot.

- Minimum
- 1st quartile
- Median
- 3rd quartile
- Maximum

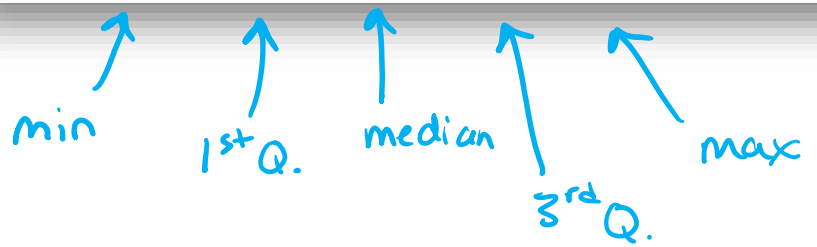
Five-Number Summary

You almost get all five from a boxplot.

- Minimum
- 1st quartile
- Median
- 3rd quartile
- Maximum

Fivenum function is in base R

```
> moons <- c(0, 0, 1, 2, 63, 61, 27, 13)
 > fivenum(moons)
[1]  0.0  0.5  7.5 44.0 63.0
```



Examples above from:
https://en.wikipedia.org/wiki/Five-number_summary

Five-Number Summary

You almost get all five from a boxplot.

- Minimum
- 1st quartile
- Median
- 3rd quartile
- Maximum

Fivenum function is in base R

```
> moons <- c(0, 0, 1, 2, 63, 61, 27, 13)
 > fivenum(moons)
[1]  0.0  0.5  7.5 44.0 63.0
```

Fivenum function must be written in Python

```
import numpy as np

def fivenum(data):
    """Five-number summary."""
    return np.percentile(data, [0, 25, 50, 75, 100], interpolation='midpoint')

moons = [0, 0, 1, 2, 63, 61, 27, 13]
print(fivenum(moons))
[  0.    0.5   7.5  44.   63. ]
```

Examples above from:
https://en.wikipedia.org/wiki/Five-number_summary

Five-Number Summary

You almost get all five from a boxplot.

- Minimum
- 1st quartile
- Median
- 3rd quartile
- Maximum

Fivenum function is in base R

```
> moons <- c(0, 0, 1, 2, 63, 61, 27, 13)
 > fivenum(moons)
[1]  0.0  0.5  7.5 44.0 63.0
```

Fivenum function must be written in Python

```
import numpy

def fivenum(c
    """Five-n
    return np

moons = [0, 0
print(fivenum
[  0.    0.5
```

HOT TIP

Want a quick way to see distribution details in python? Pip install “knowyourdata”:

```
In [3]: kyd(x)
```

Basic Statistics

Mean: -0.0007368 **Std Dev:** 0.9813

Min:	-3.161	**-99% CI:**	-2.53
1Q:	-0.6406	**-95% CI:**	-1.905
Median:	0.01018	**-68% CI:**	-0.9957
3Q:	0.6503	**+68% CI:**	0.9445
Max:	3.651	**+95% CI:**	1.938
		+99% CI:	2.493

Array Structure

Number of Dimensions:	1
Shape of Dimensions:	(2000,)
Array Data Type:	float64
Memory Size:	15.7KiB
Number of NaN:	0
Number of Inf:	0

Five-Number Summary

You almost get all five from a boxplot.

- Minimum
- 1st quartile
- Median
- 3rd quartile
- Maximum

Summary Statistics

Includes the mean too:

- Minimum
- 1st quartile
- Median
- Mean
- 3rd quartile
- Maximum

summary() behaves differently depending on class of object

```
> fivenum(x)
[1] -2.6609228 -0.4021807  0.1650809  0.7280392  2.3974525
> summary(x)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-2.6609 -0.3964  0.1651  0.1059  0.7216  2.3975
```

Pandas in Python

Confidence intervals

Confidence intervals

- An astronomer has reported that the proportion of stars in binary systems is 0.771 with a 95% confidence interval of (0.63, 0.870).
- *What does this interval mean?*
- *Is a confidence interval a random variable?*

- http://www.rossmanchance.com/applets/ConfSim.html

The Normal Distribution
(and why astronomers use "sigma")

HOT TIP

68% is equivalent to 1-sigma (1 standard deviation) only in the case of Normal/Gaussian distributions

The 68% quantile is not necessarily equal to 1-sigma in non-Gaussian distributions

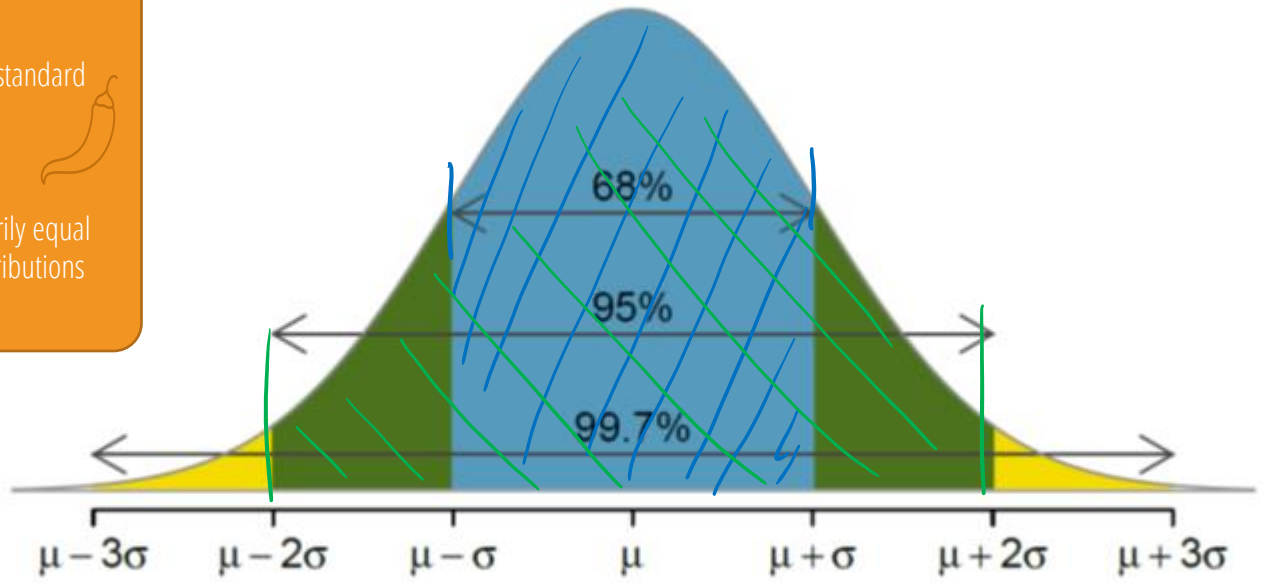
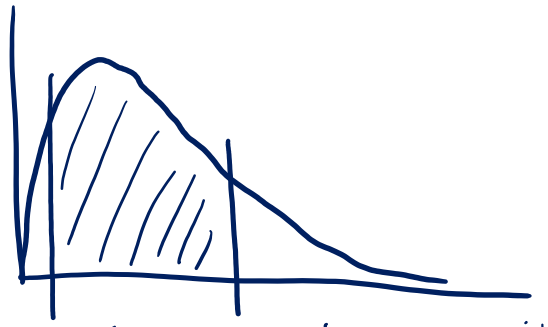


Figure 4.7: Probabilities for falling within 1, 2, and 3 standard deviations of the mean in a normal distribution.

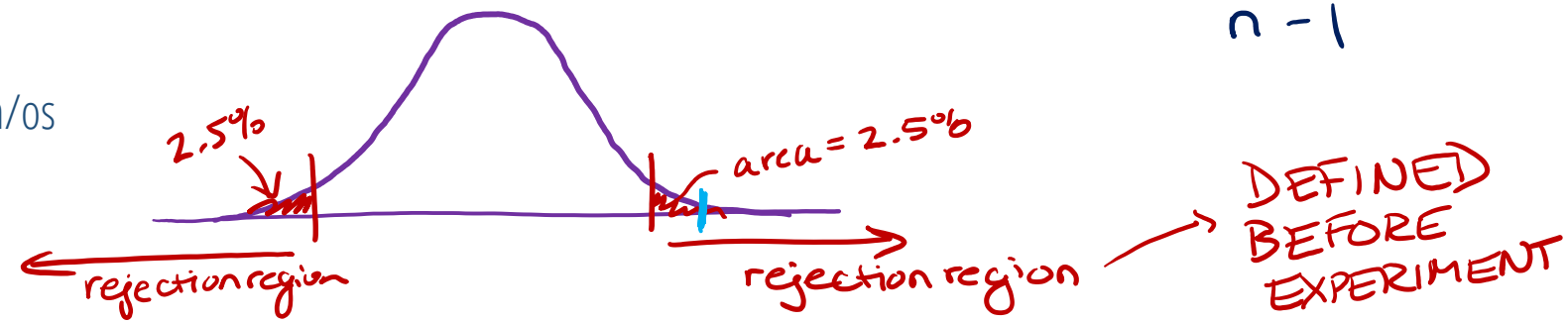
OpenIntro Stats 4th edition, https://leanpub.com/os



68% → not necessarily = 1 sigma

1-sigma = 1 standard deviation

$$\sigma = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n-1}$$



Exercise 1 (Optional)

1. Plot the PDF for the χ^2 distribution, for different values of the degrees of freedom (N) of that distribution.
2. Compare this to the normal distribution.
3. What do you notice about the two distributions?
4. Now compare the normal distribution to the log normal distribution for a range of values of the mean and variance. How do the mean and variance of the log normal distribution map onto the mean, variance of the normal distribution?

COOL CATCH

Remember that R, Python have distributions coded up – don't reinvent the wheel!

Exercise 2

- Use the *accept-reject* approach to transform numbers generated from a uniform distribution into those following the distribution $P(x) = (1/(e-1))\exp(x)$ for $0<x<1$ and 0 elsewhere
  - Draw two random samples x^*, y^* from the U(0,1) distribution
  - If $y^* < c\, f(x^*)$, keep x^* [remember the normalization c here]
  - If not, draw another two random samples from the distribution
  - Continue until you have 100 samples
  - Histogram the samples and over plot the PDF

- Use *CDF sampling* to do the same thing above.
  - To do this, compute the CDF F(X) by integrating the PDF P(x) from $-\infty$ to X
  - Then find the inverse $F^{-1}(X)$of the CDF.
    [HINT: Remember an inverse function $F^{-1}(x)$ is such that $F(F^{-1}(x)) = x$]
  - Draw a random samples x_1 from the U(0,1) distribution
  - Then the variable $y = F^{-1}(x_1)$ will have the probability distribution you seek
  - Continue until you have 100 samples
  - Histogram the samples and over plot the PDF