Probability

DATA 606 - Statistics & Probability for Data Analytics

Jason Bryer, Ph.D. and Angela Lui, Ph.D.

February 21, 2024

One Minute Paper Results

What was the most important thing you learned during this class?

What important question remains unanswered for you?

Probability

There are two key properties of probability models:

1. P(A) = The probability of event A
2. $0 \leq P(A) \leq 1$

This semester we will examine two interpretations of probabilty:

- **Frequentist interpretation**: The probability of an outcome is the proportion of times the outcome would occur if we observed the random process an infinite number of times.

- **Bayesian interpretation**: A Bayesian interprets probability as a subjective degree of belief: For the same event, two separate people could have different viewpoints and so assign different probabilities. Largely popularized by revolutionary advance in computational technology and methods during the last twenty years.

Law of Large Numbers

Law of large numbers states that as more observations are collected, the proportion of occurrences with a particular outcome, $\hat{p}_n$, converges to the probability of that outcome, p.

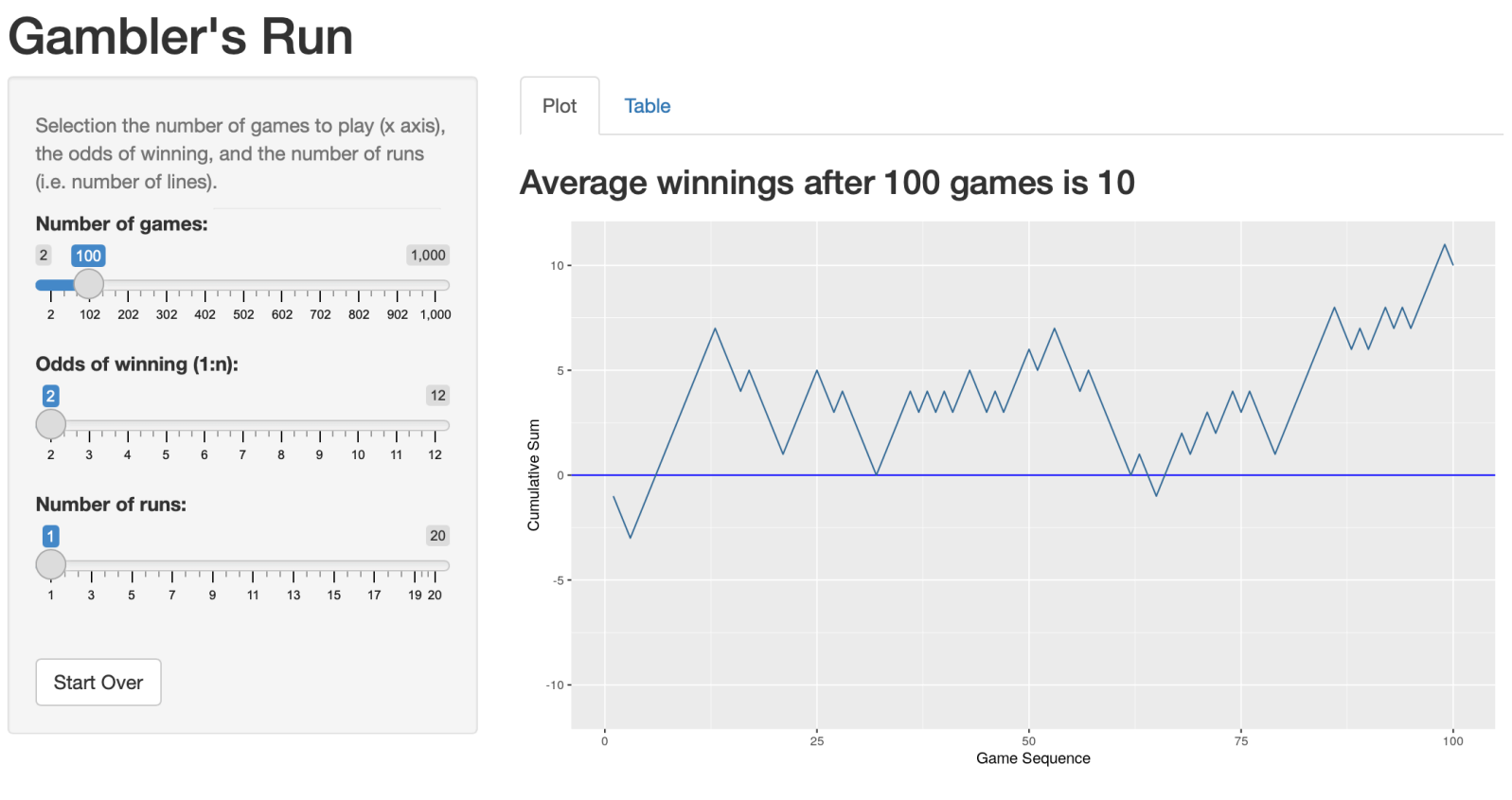
When tossing a fair coin, if heads comes up on each of the first 10 tosses, what do you think the chance is that another head will come up on the next coin toss? 0.5, less 0.5, or greater 0.5?

When tossing a fair coin, if heads comes up on each of the first 10 tosses, what do you think the chance is that another head will come up on the next coin toss? 0.5, less 0.5, or greater 0.5?

- The probability is still 0.5, or there is still a 50% chance that another head will come up on the next toss.
- The coin is not "due"" for a tail.
- The common misunderstanding of the LLN is that random processes are supposed to compensate for whatever happened in the past; this is just not true and is also called **gambler's fallacy** (or **law of averages**).

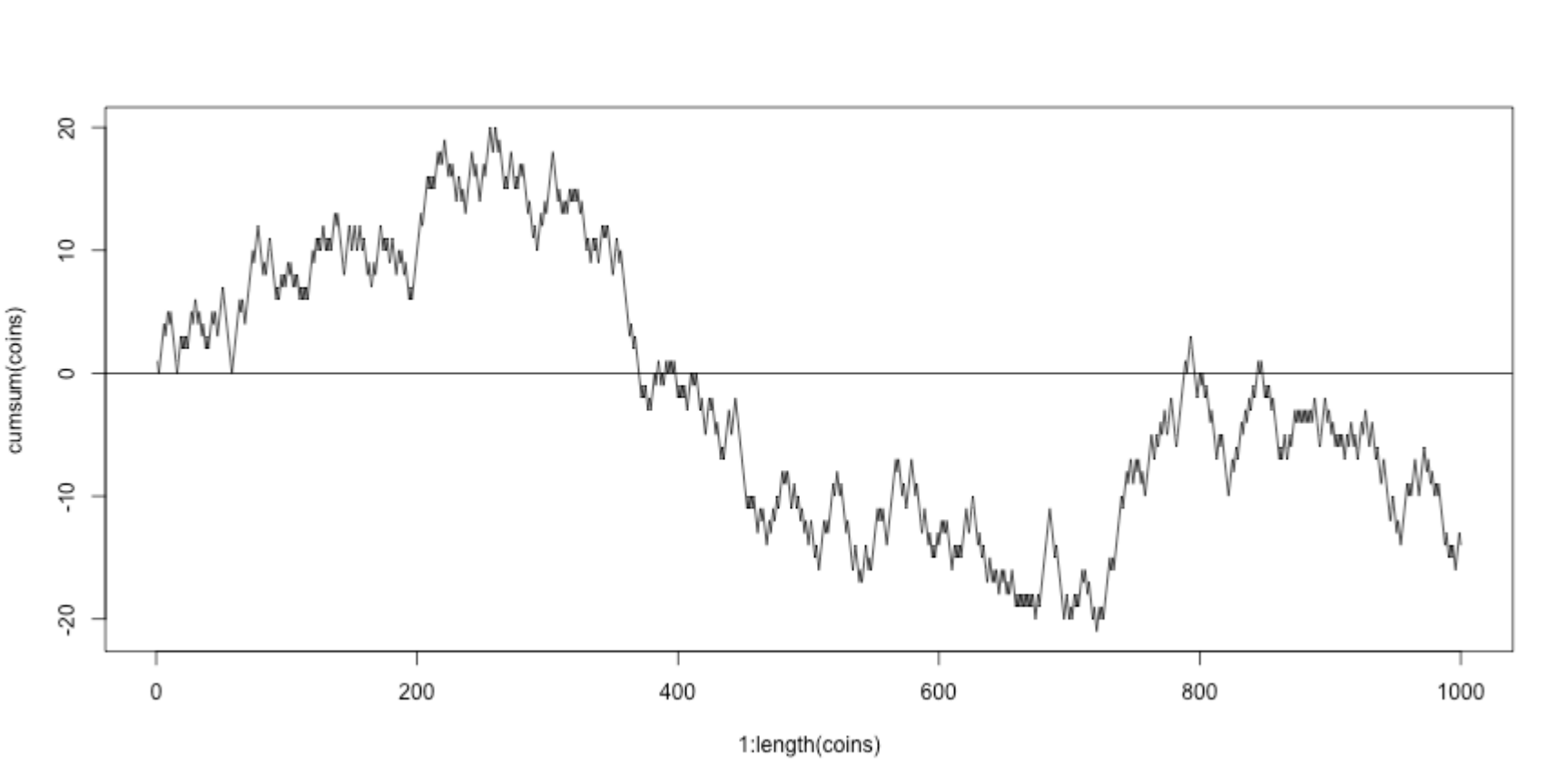
Coin Toss Demo

```
library(DATA606)
shiny_demo('gambler')
```



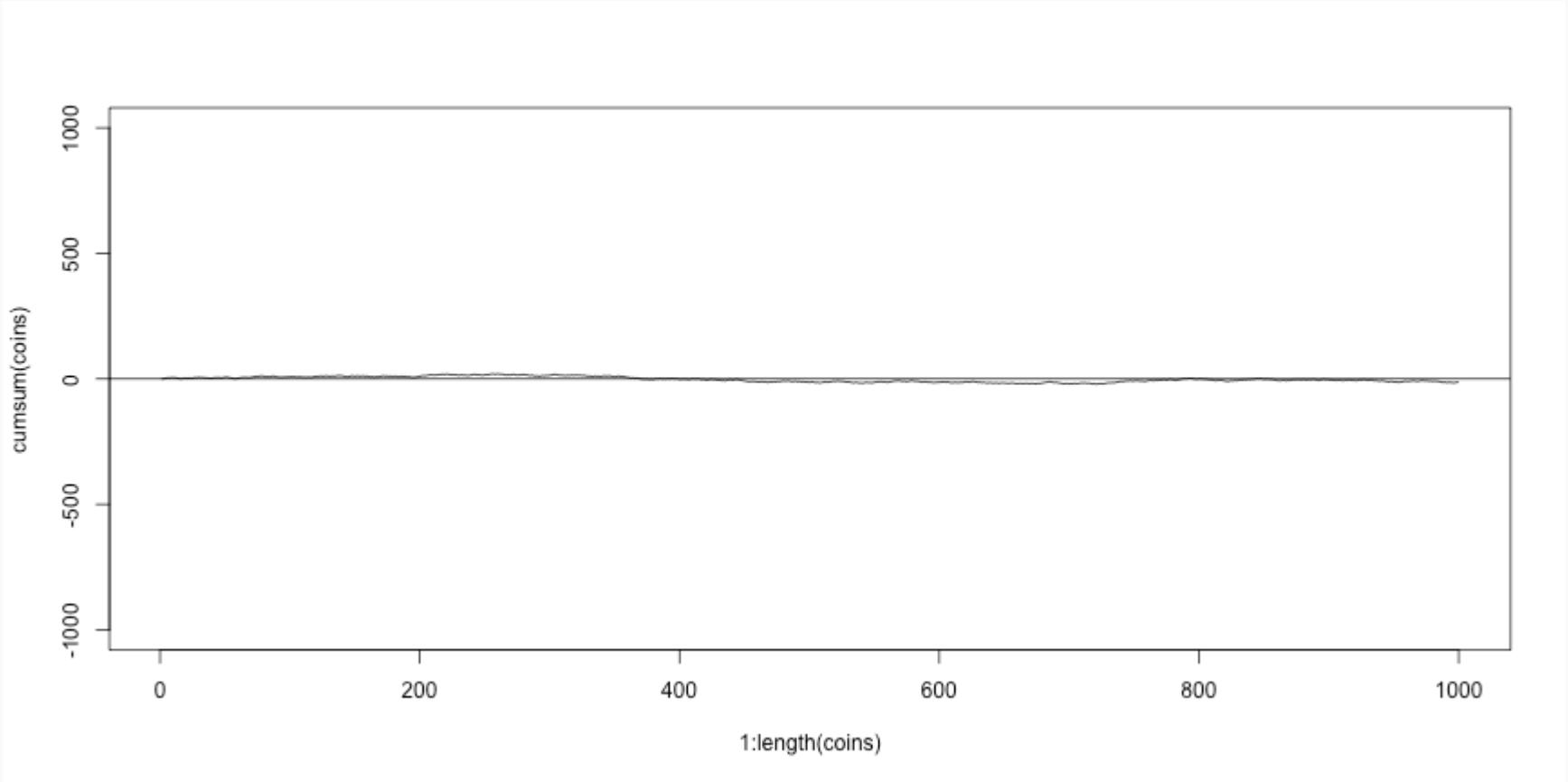
Coin Tosses

```
coins <- sample(c(-1,1), 1000, replace=TRUE)
plot(1:length(coins), cumsum(coins), type='l')
abline(h=0)
```



Coin Tosses (Full Range)

```
plot(1:length(coins), cumsum(coins), type='l', ylim=c(-1000, 1000))
abline(h=0)
```



Disjoint and non-disjoint outcomes

Disjoint (mutually exclusive) outcomes: Cannot happen at the same time.

- The outcome of a single coin toss cannot be a head and a tail. A student both cannot fail and pass a class.
- A single card drawn from a deck cannot be an ace and a queen.

Non-disjoint outcomes: Can happen at the same time.

- A student can get an A in Stats and A in Econ in the same semester.

Probability Distributions

A probability distribution lists all possible events and the probabilities with which they occur.

- The probability distribution for the a coin toss:

Event	Heads	Tails
Probability	0.5	0.5

Rules for probability distributions:

1. The events listed must be disjoint
2. Each probability must be between 0 and 1
3. The probabilities must total 1

Probabilty Distrubtions (cont.)

The probability distribution for two coin tosses:

Event	HH	TT	HT	TH
Probability	0.25	0.25	0.25	0.25

Independence

Two processes are independent if knowing the outcome of one provides no useful information about the outcome of the other.

- Knowing that the coin landed on a head on the first toss does not provide any useful information for determining what the coin will land on in the second toss. → Outcomes of two tosses of a coin are independent.
- Knowing that the first card drawn from a deck is an ace does provide useful information for determining the probability of drawing an ace in the second draw. → Outcomes of two draws from a deck of cards (without replacement) are dependent.

Checking for Independence

If P(A occurs, given that B is true) = P(A | B) = P(A), then A and B are independent.

- P(protects citizens) = 0.58
- P(randomly selected NC resident says gun ownership protects citizens, given that the resident is white) = P(protects citizens | White) = 0.67
- P(protects citizens | Black) = 0.28
- P(protects citizens | Hispanic) = 0.64

P(protects citizens) varies by race/ethnicity, therefore opinion on gun ownership and race ethnicity are most likely dependent.

Random Variables

A random variable is a numeric quantity whose value depends on the outcome of a random event

- We use a capital letter, like X, to denote a random variable
- The values of a random variable are denoted with a lowercase letter, in this case x
- For example, P(X = x)

There are two types of random variables:

- **Discrete random variables** often take only integer values
  Example: Number of credit hours, Difference in number of credit hours this term vs last

- **Continuous random variables** take real (decimal) values
  Example: Cost of books this term, Difference in cost of books this term vs last

Lottery

```
library(DATA606)
shiny_demo('lottery')
```

Lottery Tickets

This application will simulate buying a series of lottery tickets. For example, the default starting point of 365 is meant to simulate buying one lottery ticket a day for a year. The "Odds" tab provides the exact odds of winning each ticket. Clicking the "New Run" button will simulate another "year" of buying tickets showing wins along the way and the total winnings or losses at the end. Past runs will be saved and plotted in light grey to show how the current run compares to previous runs.

Number of tickets:

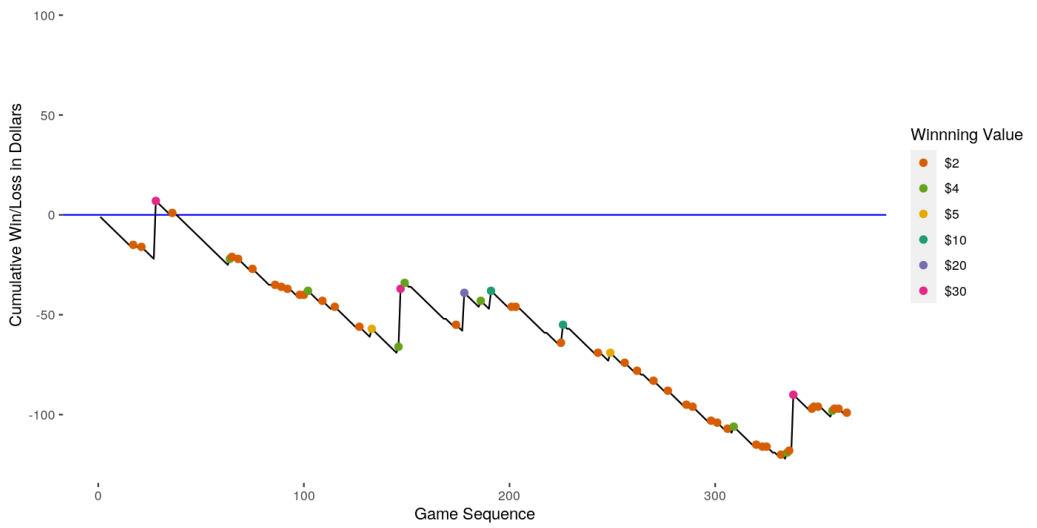
2 365 3,650

2 367 732 1,097 1,462 1,827 2,192 2,557 2,922 3,287 3,650

New Run 10 Runs

Plot Histogram Odds

Average losses after 365 games is $100



Expectation

- We are often interested in the average outcome of a random variable.
- We call this the expected value (mean), and it is a weighted average of the possible outcomes

$$\mu = E(X) = \sum_{i=1}^{k} x_i P(X = x_i)$$

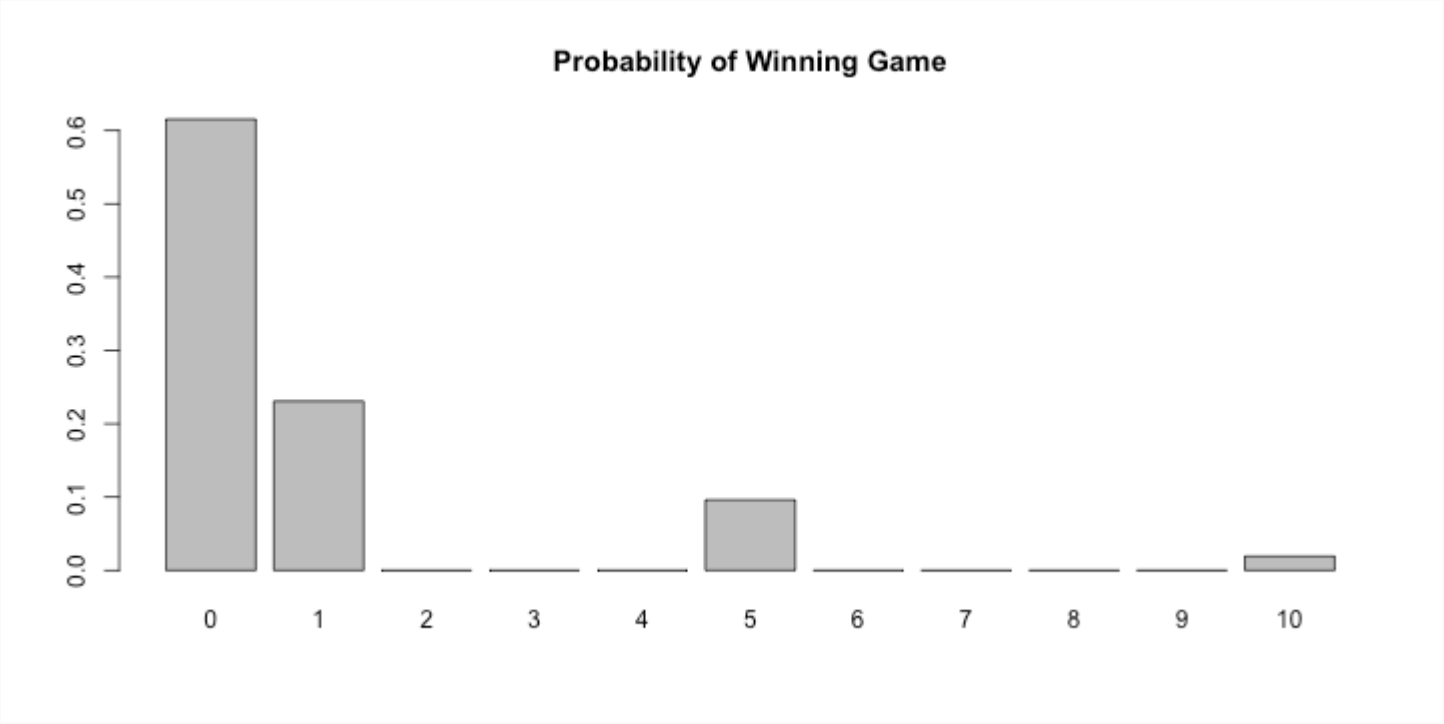
Expected value of a discrete random variable

In a game of cards you win \$1 if you draw a heart, \$5 if you draw an ace (including the ace of hearts), \$10 if you draw the king of spades and nothing for any other card you draw. Write the probability model for your winnings, and calculate your expected winning.

Event	X	P(X)	X P(X)
Heart (not Ace)	1	12/52	12/52
Ace	5	4/52	20/52
King of Spades	10	1/52	10/52
All else	0	35/52	0
Total			$E(X) = \frac{42}{52} \approx 0.81$

Expected value of a discrete random variable

```
cards <- data.frame(Event = c('Heart (not ace)','Ace','King of Spades','All else'),
    X = c(1, 5, 10, 0),    pX = c(12/52, 5/52, 1/52, 32/52) )
cards$XpX <- cards$X * cards$pX
cards2 <- rep(0, 11)
cards2[cards$X + 1] <- cards$pX
names(cards2) <- 0:10
barplot(cards2, main='Probability of Winning Game')
```



Estimating Expected Values with Simulations

```
tickets <- as.data.frame(rbind(
    c(    '$1',    1,     15),
    c(    '$2',    2,     11),
    c(    '$4',    4,     62),
    c(    '$5',    5,    100),
    c(   '$10',   10,    143),
    c(   '$20',   20,    250),
    c(   '$30',   30,    562),
    c(   '$50',   50,   3482),
    c(  '$100',  100,   6681),
    c(  '$500',  500,  49440),
    c('$1500',  1500, 375214),
    c('$2500',  2500, 618000)
), stringsAsFactors=FALSE)
names(tickets) <- c('Winnings', 'Value', 'Odds')
tickets$Value <- as.integer(tickets$Value)
tickets$Odds <- as.integer(tickets$Odds)
```

Estimating Expected Values with Simulations

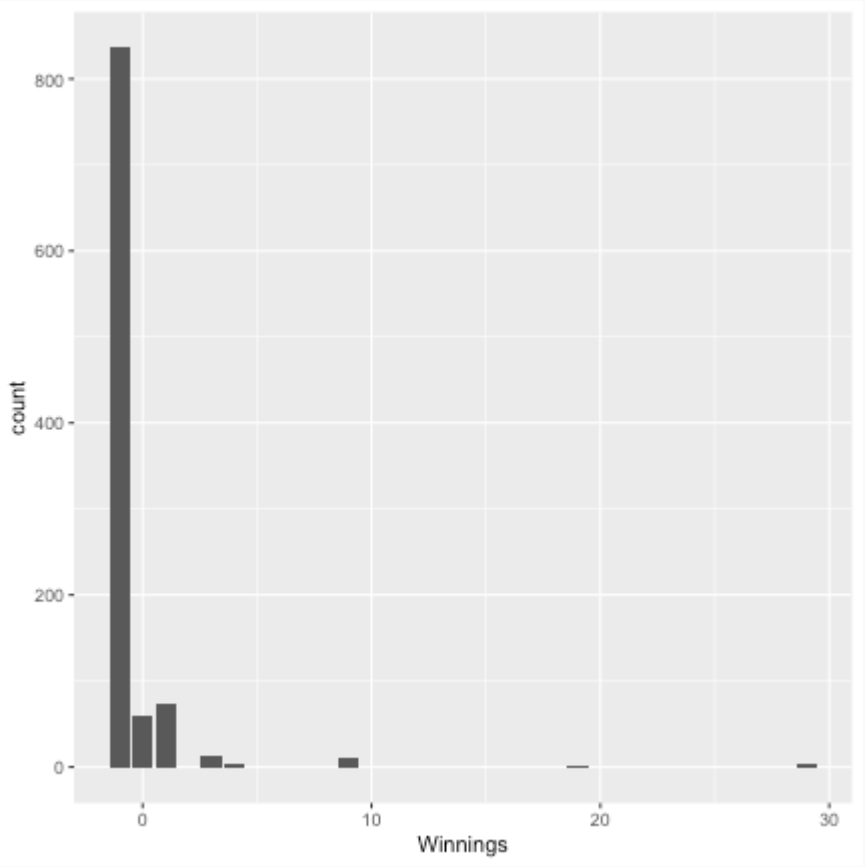
```
m <- 618000 * 375214 # A multiple of all odds
odds <- sample(m, 1000, replace=TRUE)
vals <- rep(-1, length(odds))
for(i in 1:nrow(tickets)) {
    vals[odds %% tickets[i,'Odds'] == 0] <-
        tickets[i,'Value'] - 1
}
head(vals, n=10)
```

```
##  [1] -1 -1 -1 -1 -1  0  0  0 -1 -1
```

```
mean(vals)
```

```
## [1] -0.512
```

```
ggplot(data.frame(Winnings=vals), aes(x=Winnings)) +
    geom_bar(binwidth=1)
```



Expected Value of Lottery Example

$$\mu = E(X) = \sum_{i=1}^{k} x_i P(X = x_i)$$

```
tickets
```

```
##    Winnings Value   Odds         xPx
## 1        $1     1     15 0.066666667
## 2        $2     2     11 0.181818182
## 3        $4     4     62 0.064516129
## 4        $5     5    100 0.050000000
## 5       $10    10    143 0.069930070
## 6       $20    20    250 0.080000000
## 7       $30    30    562 0.053380783
## 8       $50    50   3482 0.014359563
## 9      $100   100   6681 0.014967819
## 10     $500   500  49440 0.010113269
## 11    $1500  1500 375214 0.003997719
## 12    $2500  2500 618000 0.004045307
```

Expected value for one ticket

```
sum(tickets$xPx) - 1
```

```
## [1] -0.3862045
```

Expected Value of Lottery Example (cont)

```
sum(tickets$xPx) - 1 # Expected value for one ticket
```

```
## [1] -0.3862045
```

Simulated

```
nGames <- 1
runs <- numeric(10000)
for(j in seq_along(runs)) {
    odds <- sample(max(tickets$Odds), nGames, replace = TRUE)
    vals <- rep(-1, length(odds))
    for(i in 1:nrow(tickets)) {
        vals[odds %% tickets[i,'Odds'] == 0] <- tickets[i,'Value'] - 1
    }
    runs[j] <- cumsum(vals)[nGames]
}
mean(runs)
```

```
## [1] -0.4337
```

Note on Randomization in R

We will use many different functions throughout the course to randomly generate data. The first is the `sample` function. This function simply randomly samples from the first parameter. Consider the `letters` vector containing the 26 letters of the alphabet. Calling `sample` with just that vector will shuffle the vector.

```
letters
```

```
##  [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o" "p" "q" "r" "s"
## [20] "t" "u" "v" "w" "x" "y" "z"
```

```
sample(letters)
```

```
##  [1] "f" "w" "k" "v" "d" "x" "p" "t" "m" "u" "y" "c" "z" "r" "b" "j" "q" "l" "g"
## [20] "a" "o" "e" "h" "i" "s" "n"
```

Note on Randomization in R (cont.)

You can specify how many you want to return with the `size` parameter.

```
sample(letters, size = 1)
```

```
## [1] "w"
```

The `replace` will ensure that each randomly selected value is independent of the others.

```
sample(letters, size = 30, replace = TRUE)
```

```
##  [1] "n" "t" "x" "o" "m" "x" "p" "t" "a" "m" "n" "g" "b" "k" "p" "e" "n" "g" "x"
## [20] "p" "s" "i" "s" "q" "b" "n" "j" "d" "c" "r"
```

Coins Example

```
coin <- c('H', 'T')
sample(coin)
```

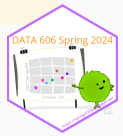
```
## [1] "T" "H"
```

```
sample(coin, 1)
```

```
## [1] "T"
```

```
sample(coin, 100, replace = TRUE)
```

```
##   [1] "H" "H" "T" "T" "T" "H" "H" "H" "H" "T" "H" "H" "H" "H" "T" "T" "T" "H"
##  [19] "H" "H" "T" "H" "T" "T" "T" "T" "H" "T" "H" "T" "H" "T" "H" "H" "T" "H"
##  [37] "H" "H" "H" "H" "H" "T" "H" "H" "T" "H" "T" "H" "T" "H" "H" "H" "T" "T"
##  [55] "H" "H" "H" "T" "T" "H" "H" "T" "H" "H" "T" "T" "H" "H" "T" "H" "H" "T"
##  [73] "H" "T" "T" "H" "H" "H" "T" "H" "T" "H" "H" "T" "T" "T" "H" "H" "H" "T"
##  [91] "T" "T" "H" "T" "H" "T" "H" "T" "H" "T"
```



Seeds

Computers are generally not good at randomizaiton. Instead, R (and really all programs) uses a pseudo random algorithm. These algorithms rely on a seed, or starting point for the algorithm. You can set the seed to ensure that your analysis is reproducible. For example, setting the seed below before calling `sample` will ensure we get the same answer.

```
set.seed(2112); sample(100, 1)
```

```
## [1] 6
```

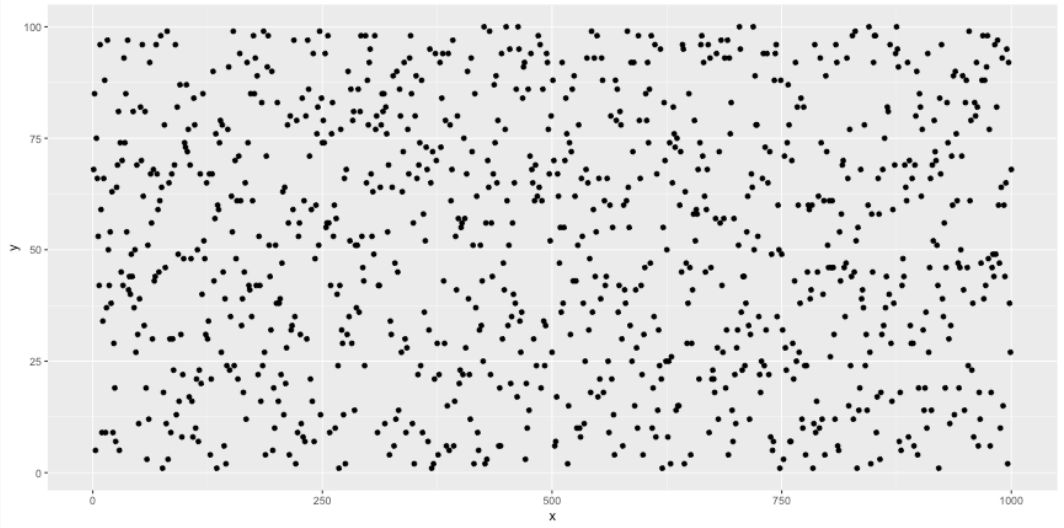
```
set.seed(2112); sample(100, 1)
```

```
## [1] 6
```

Is it really random?

```
df <- data.frame(x = 1:1000, y = NA_integer_)
for(i in 1:nrow(df)) {
    set.seed(i)
    df[i,]$y <- sample(100, 1)
}
```

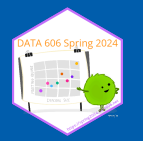
```
ggplot(df, aes(x = x, y = y)) + geom_point()
```




```
cor.test(df$x, df$y)
```

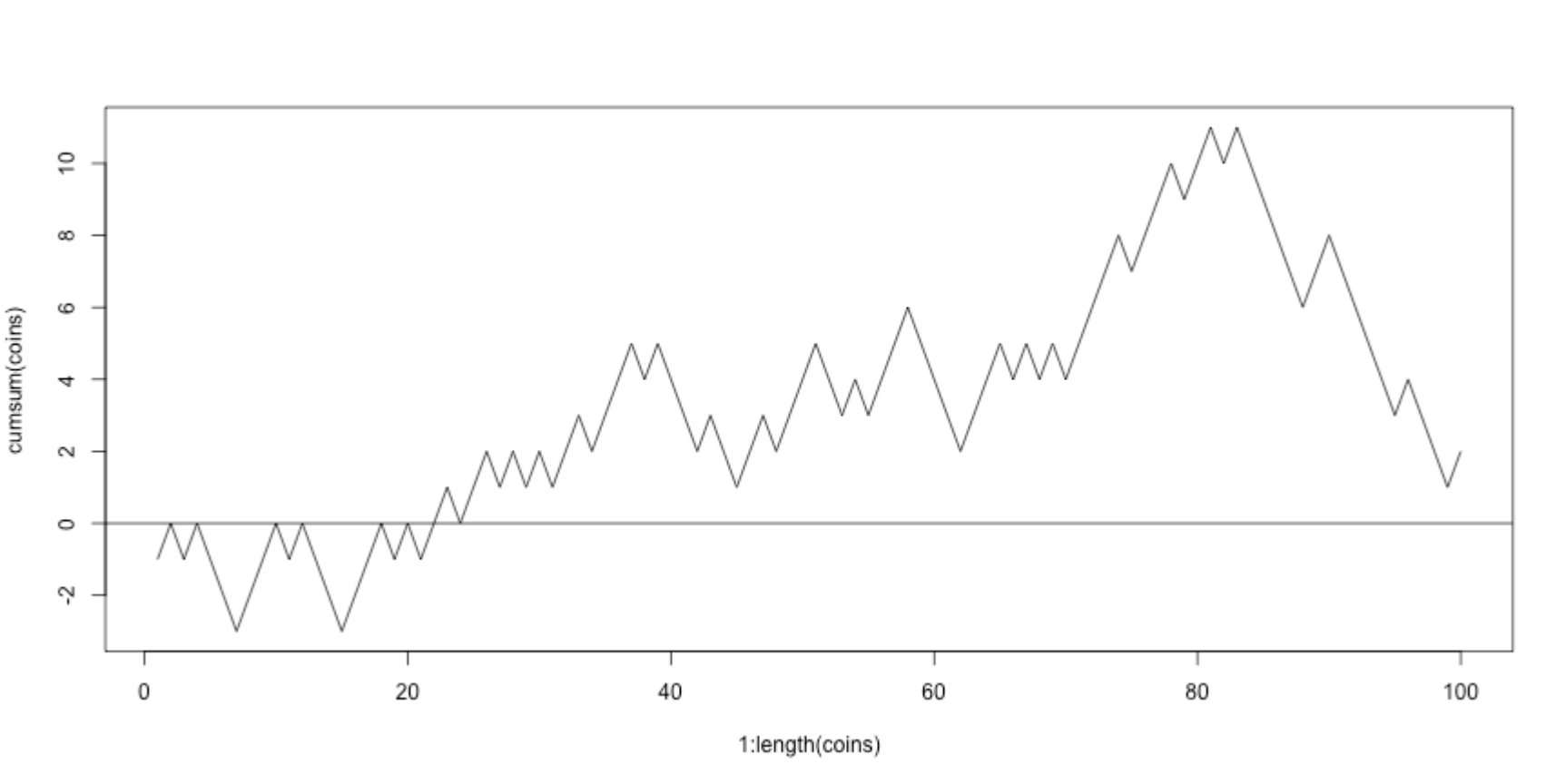
```
##
##     Pearson's product-moment correlation
##
## data:  df$x and df$y
## t = -0.11161, df = 998, p-value = 0.9112
## alternative hypothesis: true correlation is not equa
## 95 percent confidence interval:
##  -0.06551171  0.05847292
## sample estimates:
##          cor
## -0.003532972
```

Distributions



Coin Tosses Revisited

```
coins <- sample(c(-1,1), 100, replace=TRUE)
plot(1:length(coins), cumsum(coins), type='l')
abline(h=0)
```



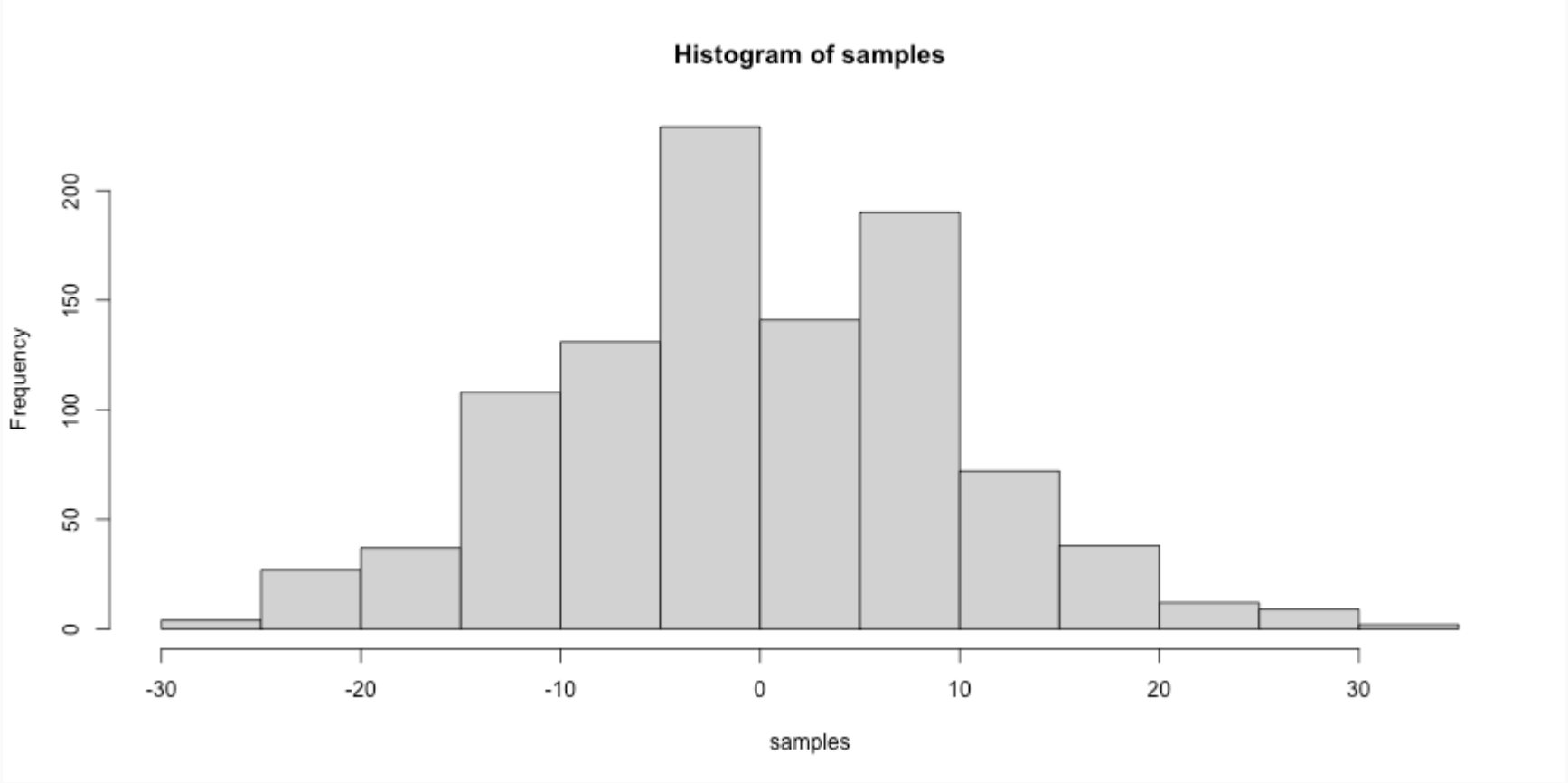
Many Random Samples

```
samples <- rep(NA, 1000)
for(i in seq_along(samples)) {
    coins <- sample(c(-1,1), 100, replace=TRUE)
    samples[i] <- cumsum(coins)[length(coins)]
}
head(samples, n = 15)
```

```
##  [1] -18  -6  -8 -14  -4   0   8   2  -8   6   8   0  -4   0  -6
```

Histogram of Many Random Samples

```
hist(samples)
```



Properties of Distribution

```
(m.sam <- mean(samples))
```

```
## [1] 0.148
```

```
(s.sam <- sd(samples))
```

```
## [1] 10.10942
```

Properties of Distribution (cont.)

```
within1sd <- samples[samples >= m.sam - s.sam & samples <= m.sam + s.sam]
length(within1sd) / length(samples)
```

```
## [1] 0.691
```

```
within2sd <- samples[samples >= m.sam - 2 * s.sam & samples <= m.sam + 2* s.sam]
length(within2sd) / length(samples)
```

```
## [1] 0.958
```

```
within3sd <- samples[samples >= m.sam - 3 * s.sam & samples <= m.sam + 3 * s.sam]
length(within3sd) / length(samples)
```

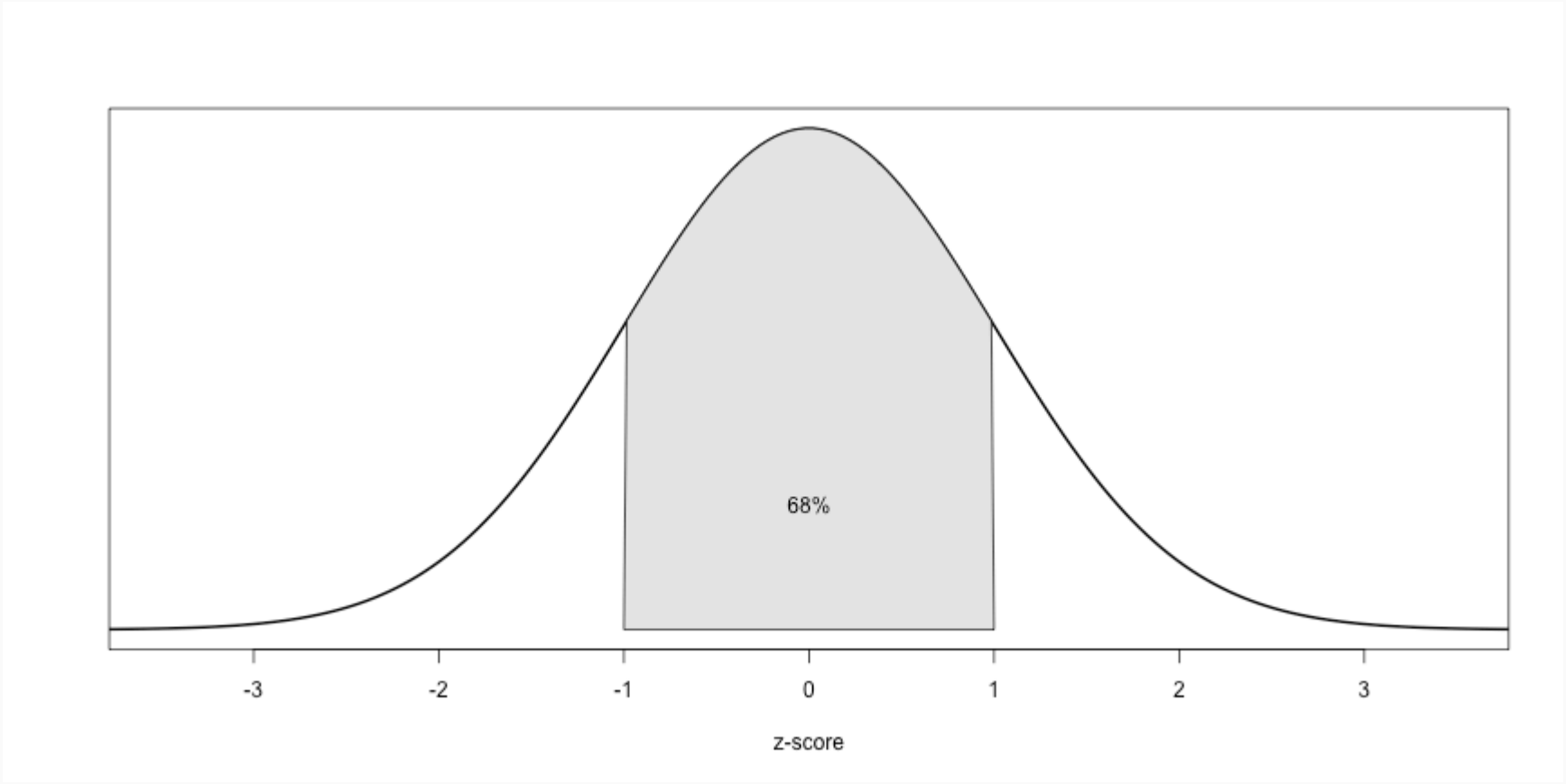
```
## [1] 0.998
```

Standard Normal Distribution

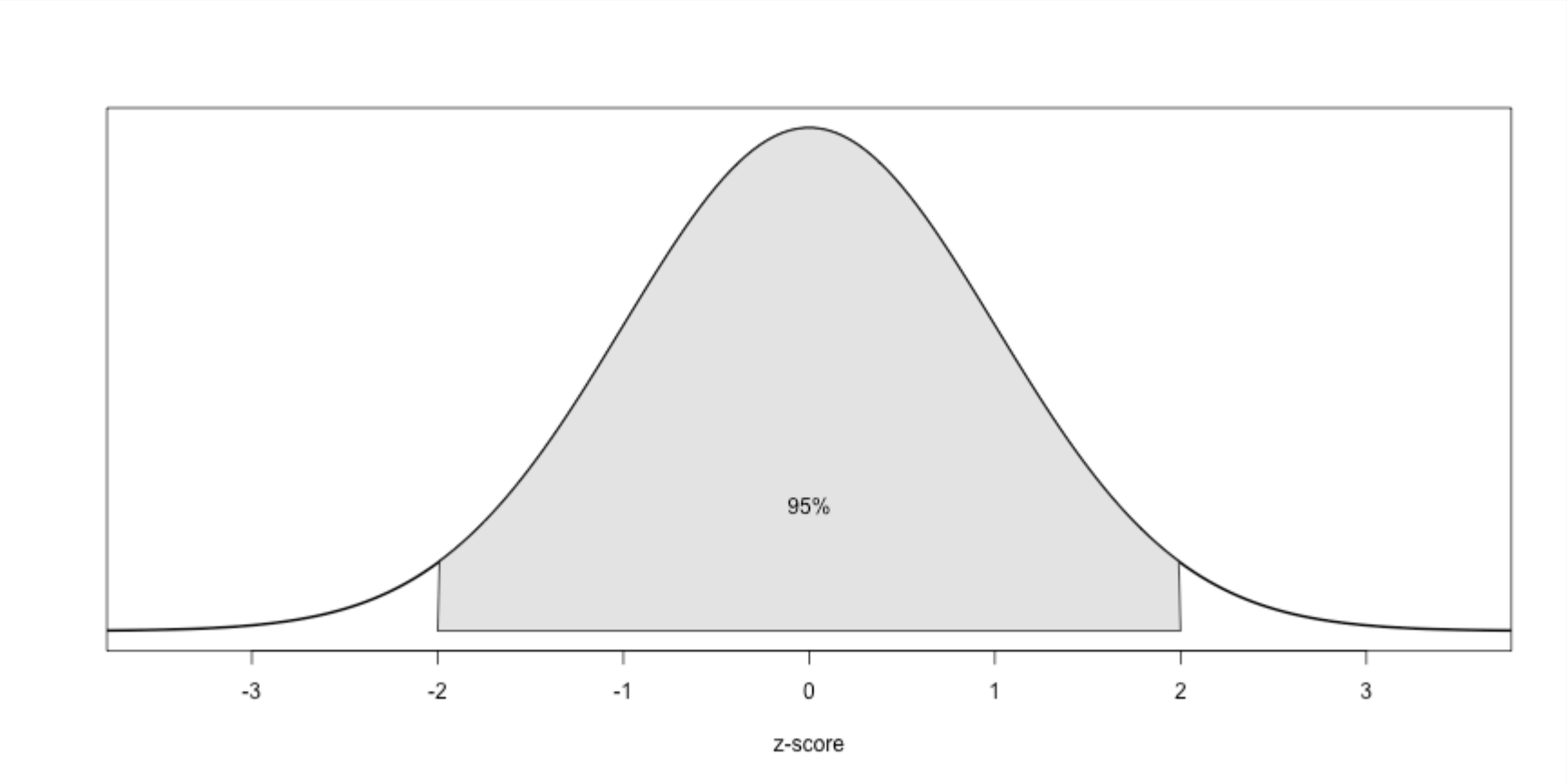
$$f\left(x|\mu,\sigma\right)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

```
x <- seq(-4,4,length=200); y <- dnorm(x,mean=0, sd=1)
plot(x, y, type = "l", lwd = 2, xlim = c(-3.5,3.5), ylab='', xlab='z-score', yaxt='n')
```

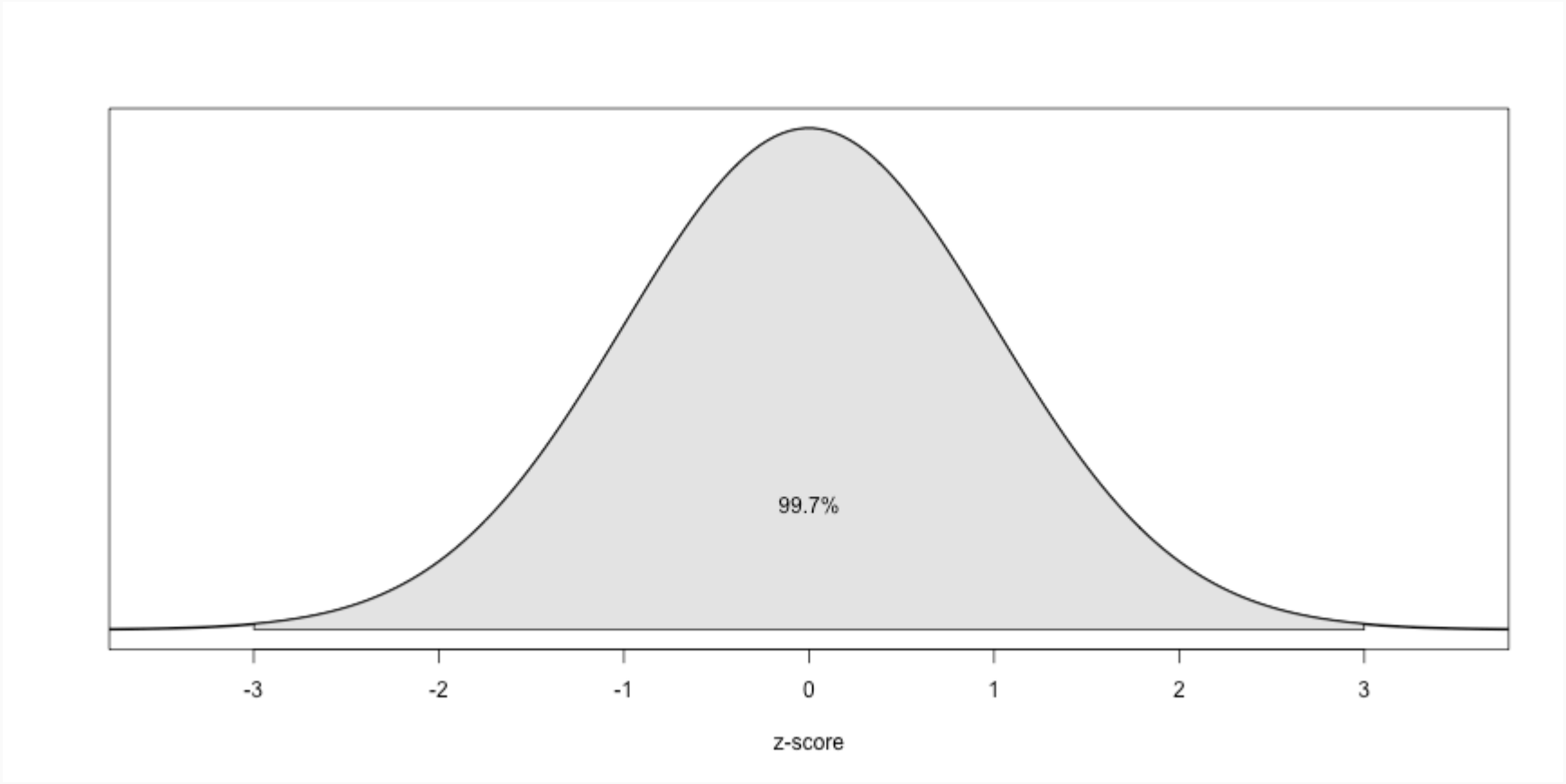
Standard Normal Distribution



Standard Normal Distribution



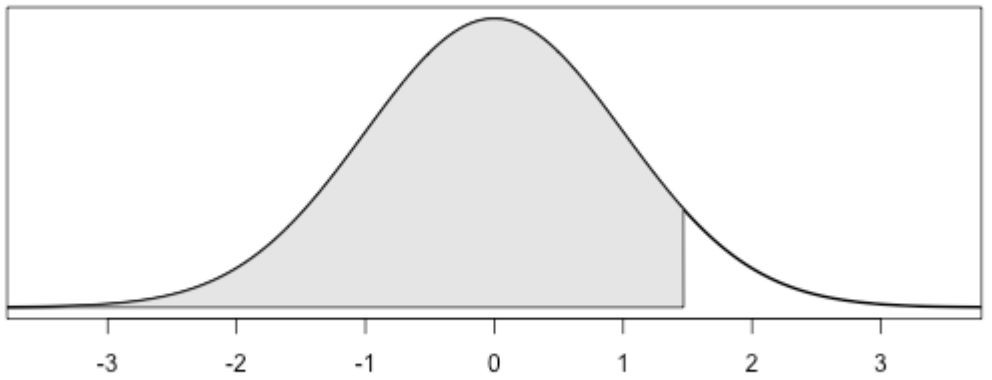
Standard Normal Distribution



What's the likelihood of ending with less than 15?

```
pnorm(15, mean=mean(samples), sd=sd(samples))
```

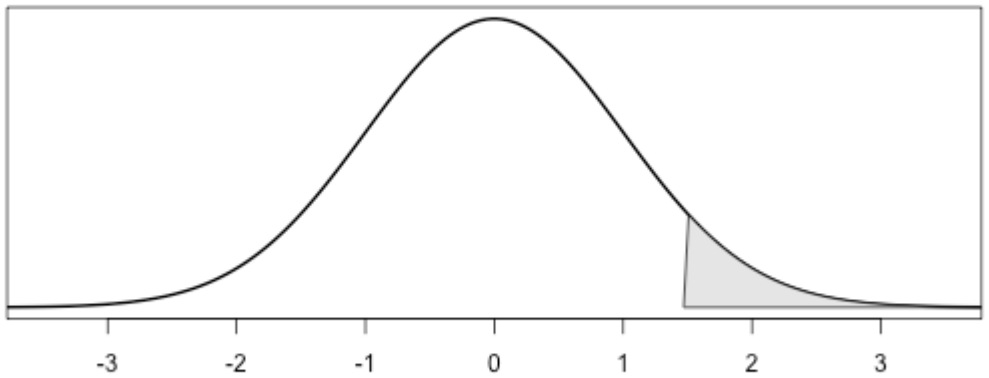
```
## [1] 0.9291006
```



What's the likelihood of ending with more than 15?

```
1 - pnorm(15, mean=mean(samples), sd=sd(samples))
```

```
## [1] 0.07089939
```



Comparing Scores on Different Scales

SAT scores are distributed nearly normally with mean 1500 and standard deviation 300. ACT scores are distributed nearly normally with mean 21 and standard deviation 5. A college admissions officer wants to determine which of the two applicants scored better on their standardized test with respect to the other test takers: Pam, who earned an 1800 on her SAT, or Jim, who scored a 24 on his ACT?

Z-Scores

- Z-scores are often called standard scores:

$$Z = \frac{observation - mean}{SD}$$

- Z-Scores have a mean = 0 and standard deviation = 1.

Converting Pam and Jim's scores to z-scores:

$$Z_{Pam} = \frac{1800 - 1500}{300} = 1$$

$$Z_{Jim} = \frac{24 - 21}{5} = 0.6$$

Dual Scales

Some problems[1]:

- The designer has to make choices about scales and this can have a big impact on the viewer
- "Cross-over points" where one series cross another are results of the design choices, not intrinsic to the data, and viewers (particularly unsophisticated viewers)
- They make it easier to lazily associate correlation with causation, not taking into account autocorrelation and other time-series issues
- Because of the issues above, in malicious hands they make it possible to deliberately mislead

This example looks at the relationship between NZ dollar exchange rate and trade weighted index.

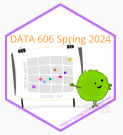
```
DATA606::shiny_demo('DualScales', package='DATA606')
```

My advise:

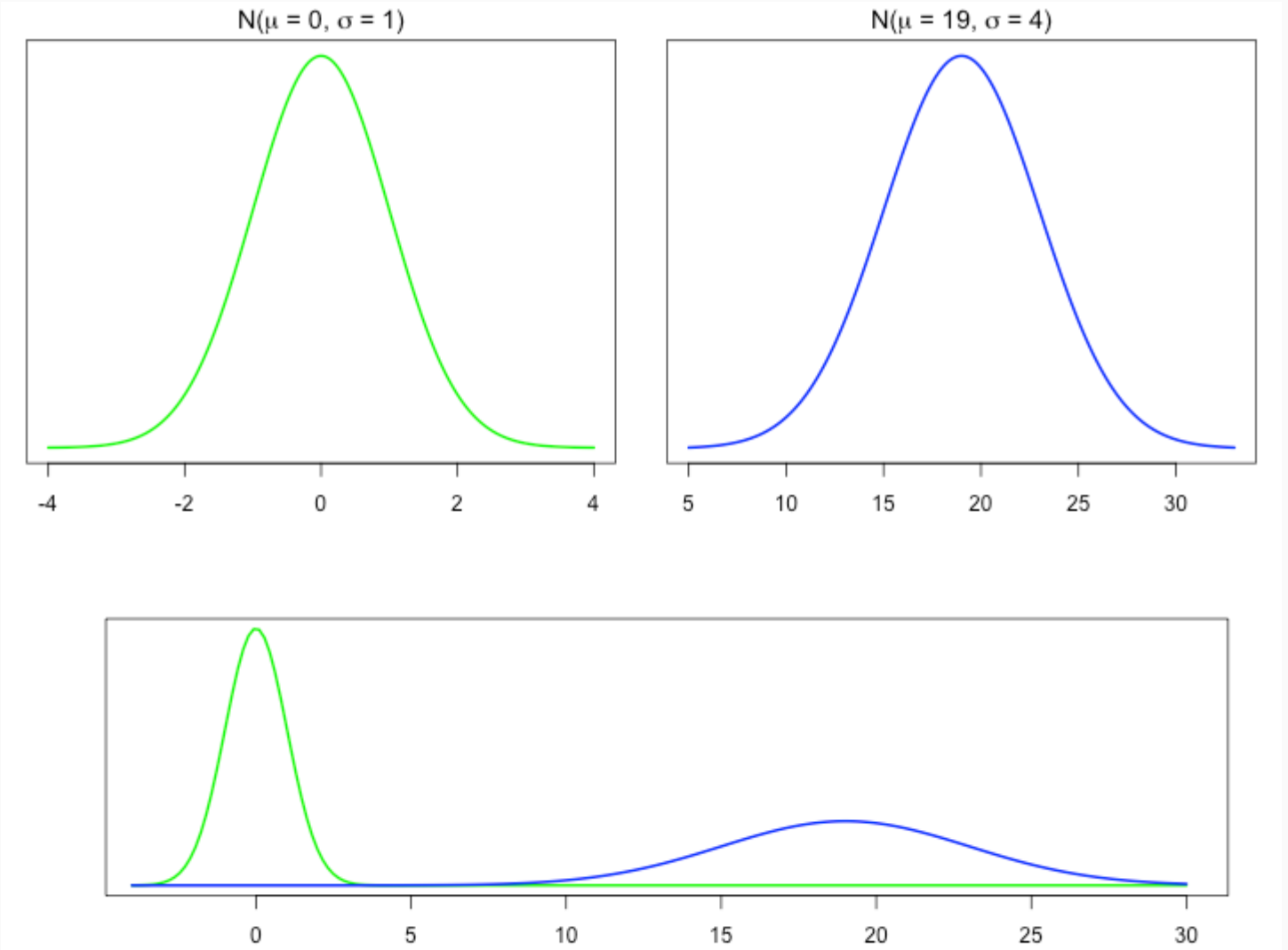
- Avoid using them. You can usually do better with other plot types.
- When necessary (or compelled) to use them, rescale (using z-scores, we'll discuss this in a few weeks)

[1] http://blog.revolutionanalytics.com/2016/08/dual-axis-time-series.html

[2] http://ellisp.github.io/blog/2016/08/18/dualaxes



Standard Normal Parameters



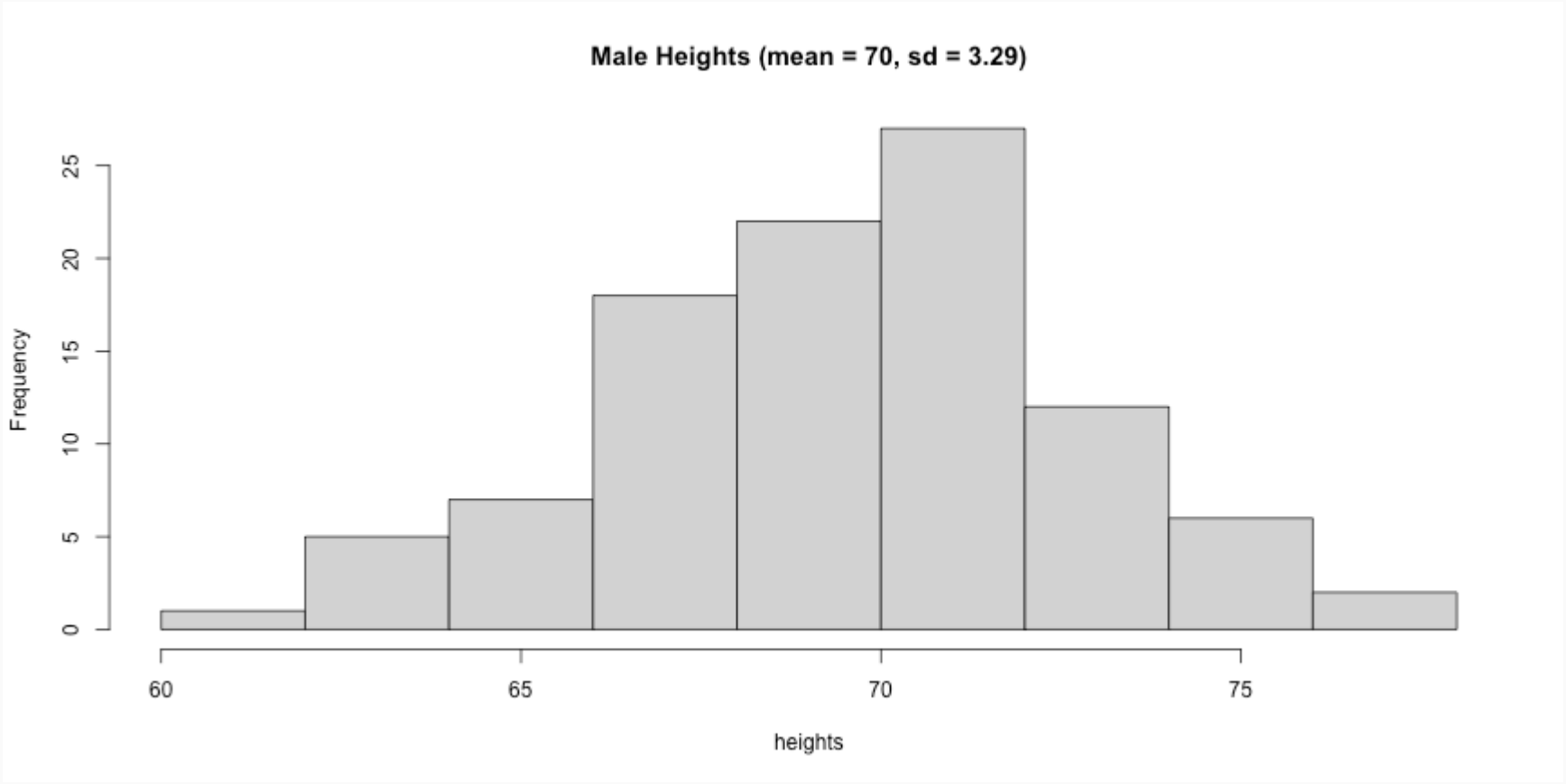
SAT Variability

SAT scores are distributed nearly normally with mean 1500 and standard deviation 300.

- 68% of students score between 1200 and 1800 on the SAT.
- 95% of students score between 900 and 2100 on the SAT.
- 99.7% of students score between 600 and 2400 on the SAT.

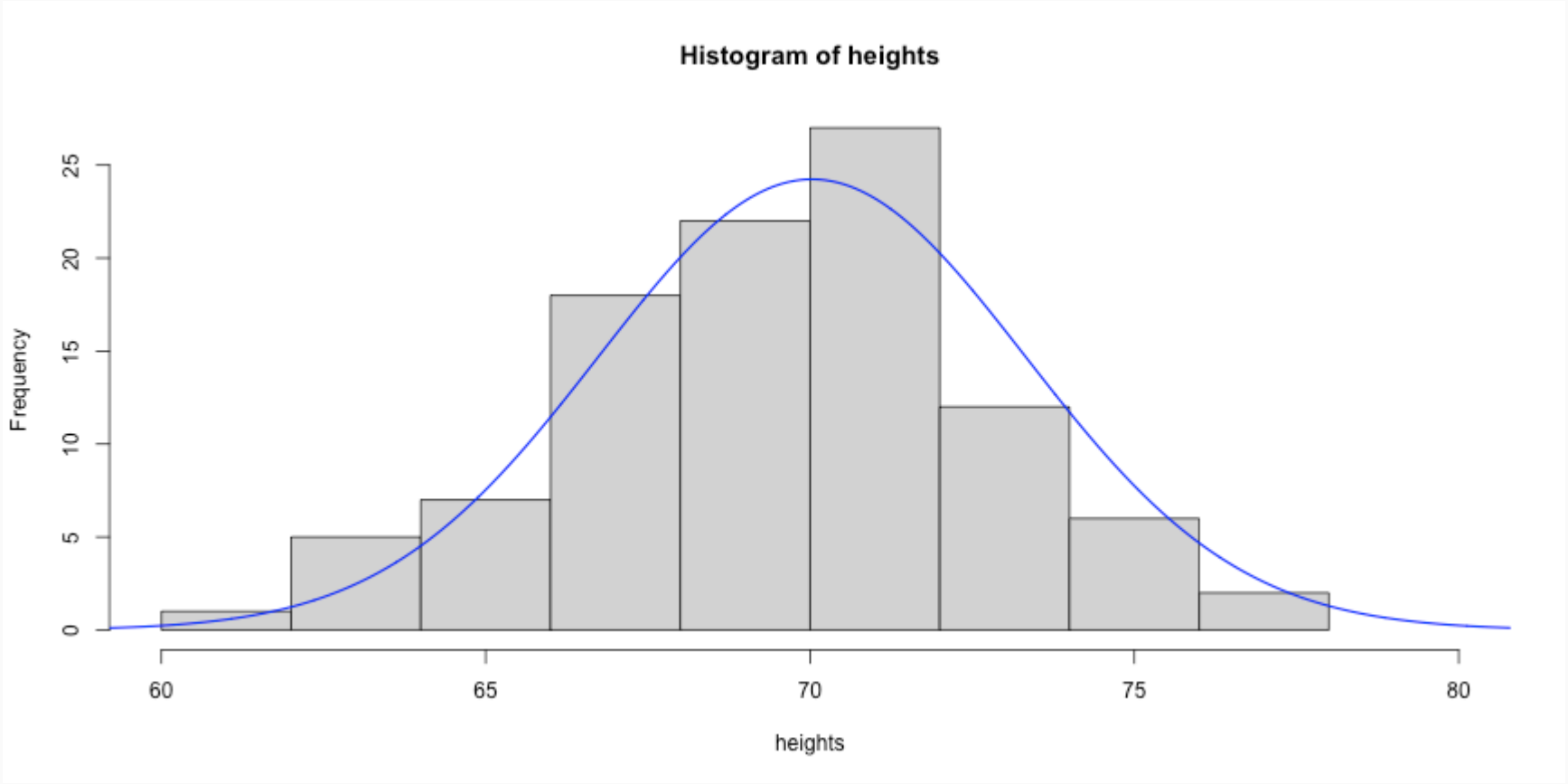
Evaluating Normal Approximation

To use the 68-95-99 rule, we must verify the normality assumption. We will want to do this also later when we talk about various (parametric) modeling. Consider a sample of 100 male heights (in inches).

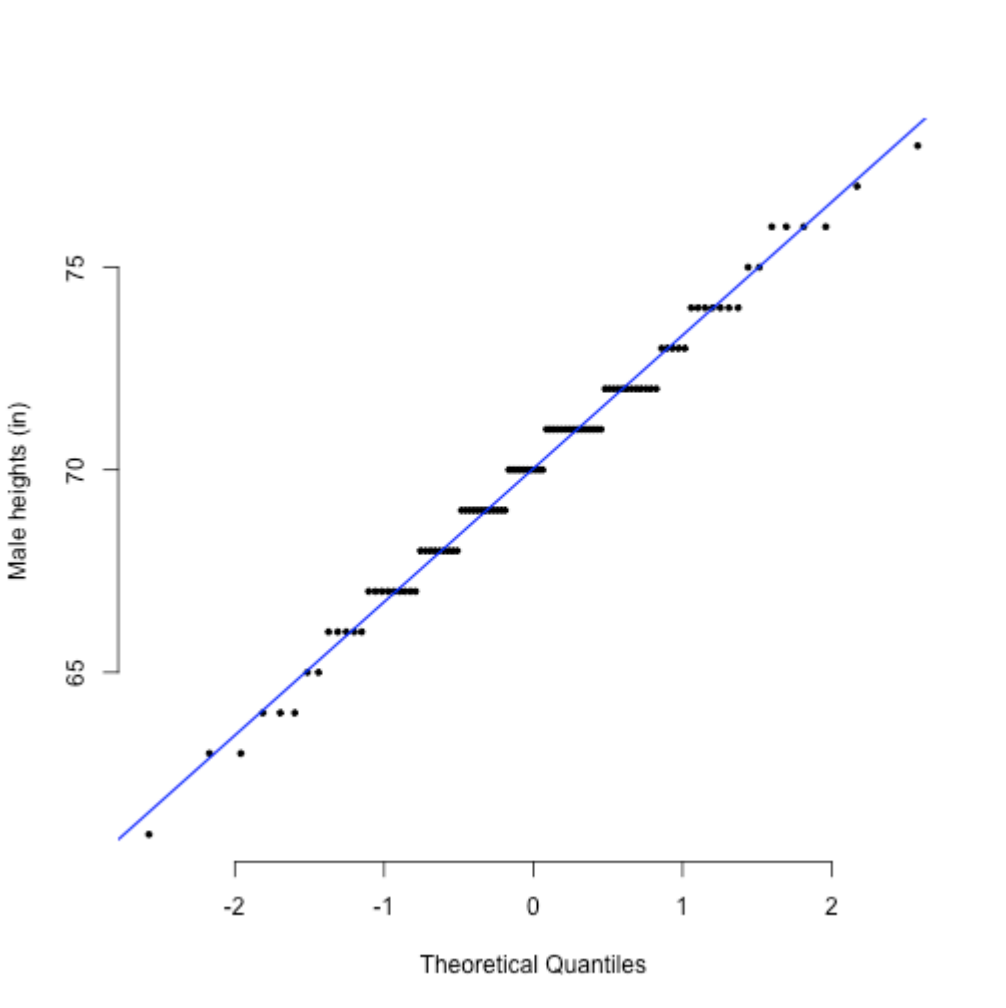


Evaluating Normal Approximation

Histogram looks normal, but we can overlay a standard normal curve to help evaluation.

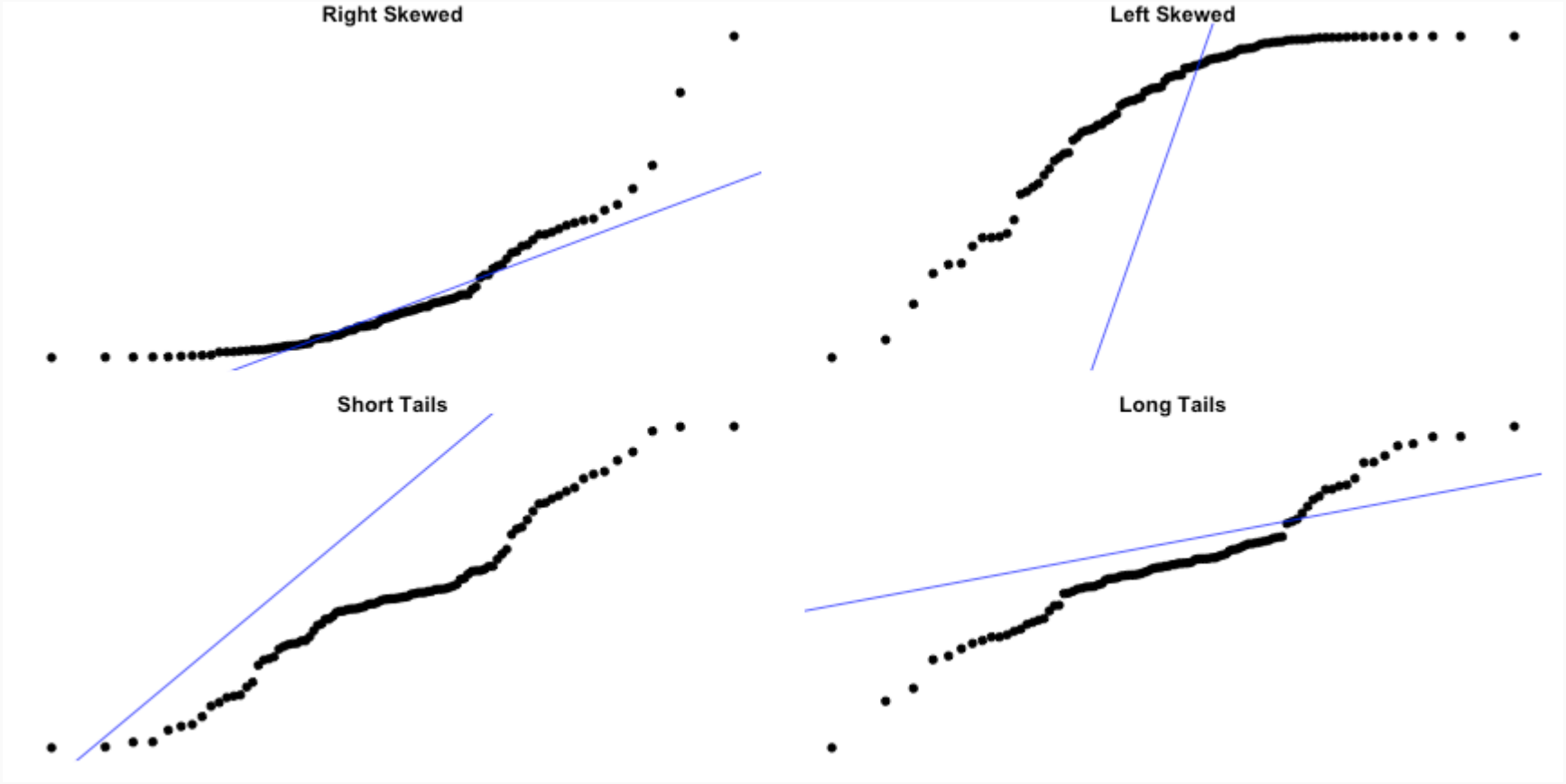


Normal Q-Q Plot



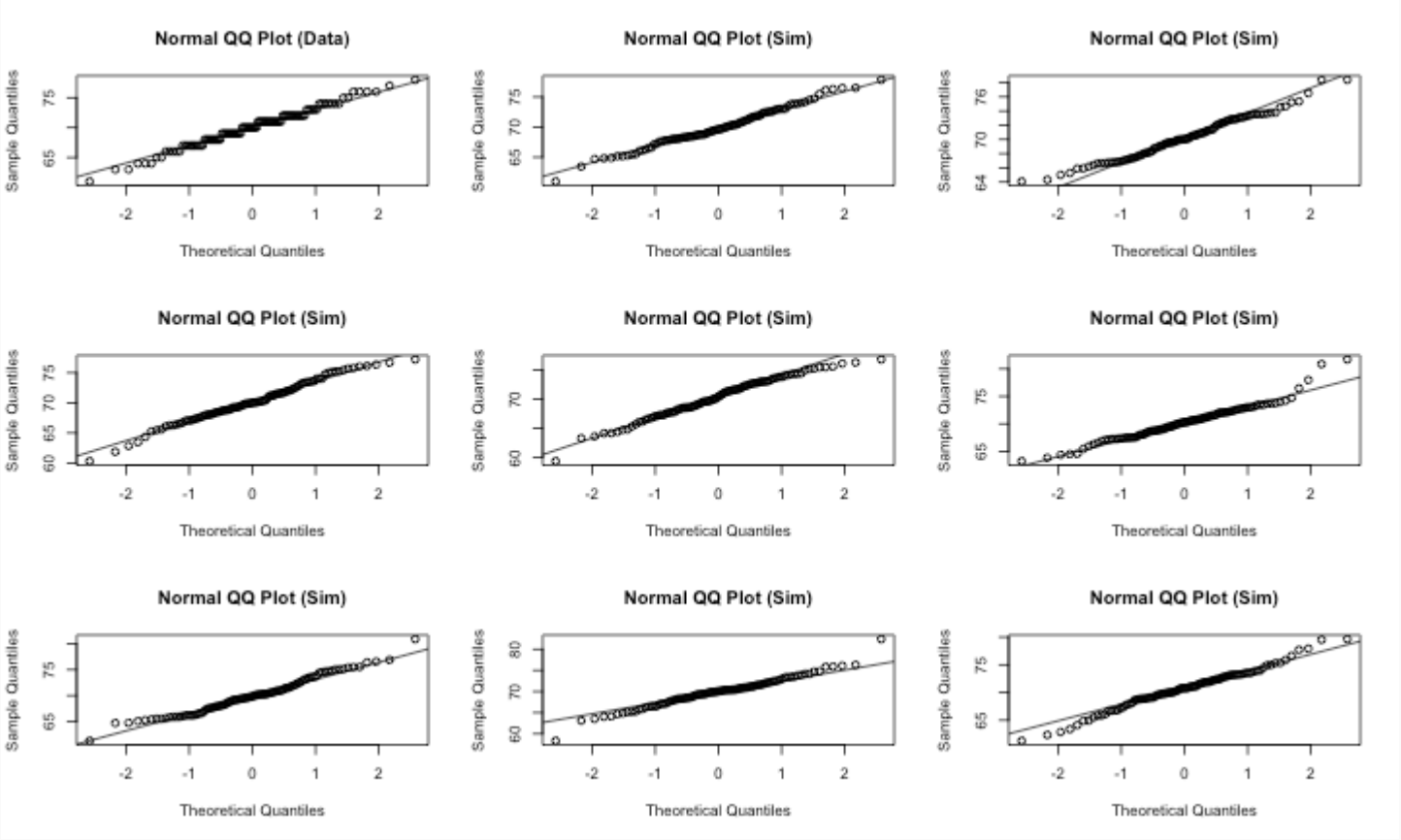
- Data are plotted on the y-axis of a normal probability plot, and theoretical quantiles (following a normal distribution) on the x-axis.
- If there is a linear relationship in the plot, then the data follow a nearly normal distribution.
- Constructing a normal probability plot requires calculating percentiles and corresponding z-scores for each observation, which is tedious. Therefore we generally rely on software when making these plots.

Skewness



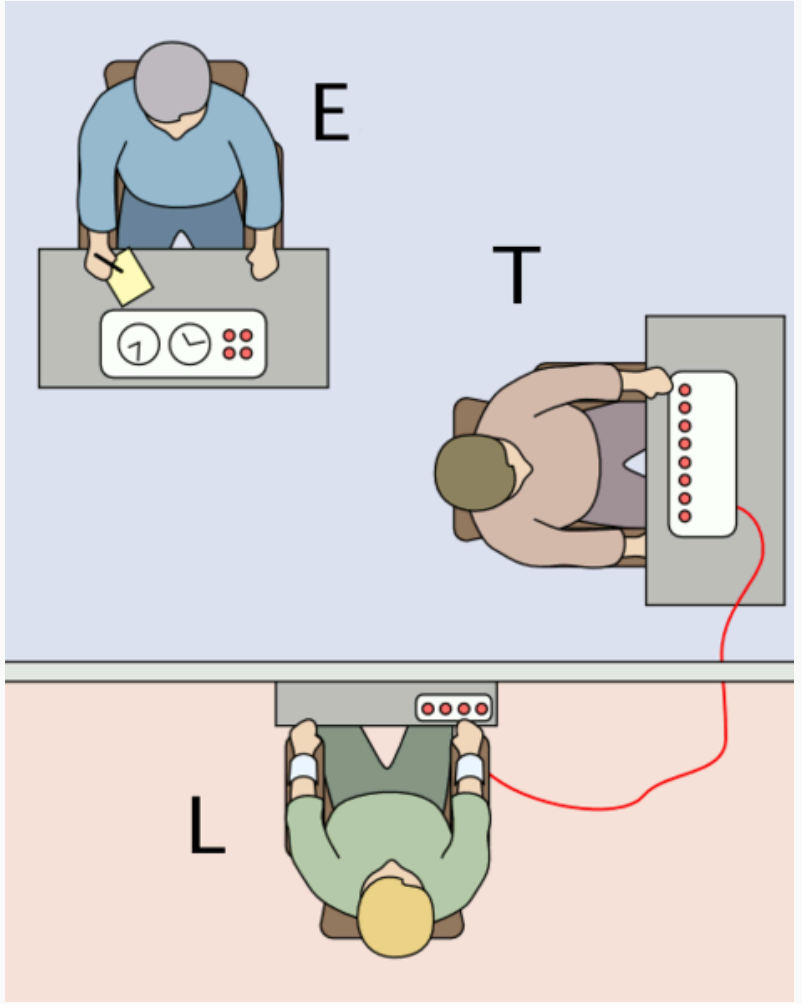
Simulated Normal Q-Q Plots

```
DATA606::qqnormsim(heights)
```



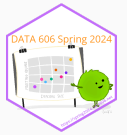
Milgram Experiment

- Stanley Milgram conducted a series of experiments on obedience to authority starting in 1963.
- Experimenter (E) orders the teacher (T), the subject of the experiment, to give severe electric shocks to a learner (L) each time the learner answers a question incorrectly.



Milgram Experiment (cont.)

- The learner is actually an actor, and the electric shocks are not real, but a prerecorded sound is played each time the teacher administers an electric shock.
- These experiments measured the willingness of study participants to obey an authority figure who instructed them to perform acts that conflicted with their personal conscience.
- Milgram found that about 65% of people would obey authority and give such shocks.
- Over the years, additional research suggested this number is approximately consistent across communities and time.



Bernoulli Sequences

- Each person in Milgram's experiment can be thought of as a trial.
- A person is labeled a success if she refuses to administer a severe shock, and failure if she administers such shock.
- Since only 35% of people refused to administer a shock, probability of success is p = 0.35.
- When an individual trial has only two possible outcomes, it is called a **Bernoulli** random variable.

A random variable X has a *Bernoulli distribution* with parameter p if

$$P(X = 1) = p \quad and \quad P(X = 0) = 1 - p$$

for $0 < p < 1$

Geometric distribution

Dr. Smith wants to repeat Milgrams experiments but she only wants to sample people until she finds someone who will not inflict a severe shock. What is the probability that she stops after the first person?

$$P(1^{st} \quad person \quad refuses) = 0.35$$

the third person?

$$P(1^{st} and 2^{nd} shock, 3^{rd} refuses) = \frac{S}{0.65} \times \frac{S}{0.65} \times \frac{R}{0.35} = 0.65^2 \times 0.35 \approx 0.15$$

the tenth person?

Geometric distribution (cont.)

Geometric distribution describes the waiting time until a success for *independent and identically distributed* (iid) Bernouilli random variables.

- independence: outcomes of trials don't affect each other
- identical: the probability of success is the same for each trial

Geometric probabilities

If p represents probability of success, $(1-p)$ represents probability of failure, and n represents number of independent trials

$$P(success \quad on \quad the \quad n^{th} \quad trial) = (1-p)^{n-1}p$$

Expected value

How many people is Dr. Smith expected to test before finding the first one that refuses to administer the shock?

The expected value, or the mean, of a geometric distribution is defined as $\frac{1}{p}$.

$$\mu = \frac{1}{p} = \frac{1}{0.35} = 2.86$$

She is expected to test 2.86 people before finding the first one that refuses to administer the shock.

But how can she test a non-whole number of people?

Expected value and its variability

$$\mu = \frac{1}{p} \qquad \sigma = \sqrt{\frac{1-p}{p^2}}$$

Going back to Dr. Smith's experiment:

$$\sigma = \sqrt{\frac{1-p}{p^2}} = \sqrt{\frac{1-0.35}{0.35^2}} = 2.3$$

Dr. Smith is expected to test 2.86 people before finding the first one that refuses to administer the shock, give or take 2.3 people.

These values only make sense in the context of repeating the experiment many many times.

Milgram Part 2

Suppose we randomly select four individuals to participate in this experiment. What is the probability that exactly 1 of them will refuse to administer the shock

Let's call these people Allen (A), Brittany (B), Caroline (C), and Damian (D). Each one of the four scenarios below will satisfy the condition of "exactly 1 of them refuses to administer the shock":

$$\text{Scenario 1: } \underbrace{0.35}_{\text{(A) } \textit{refuse}} \times \underbrace{0.65}_{\text{(B) shock}} \times \underbrace{0.65}_{\text{(C) shock}} \times \underbrace{0.65}_{\text{(D) shock}} = 0.0961$$

$$\text{Scenario 2: } \underbrace{0.65}_{\text{(A) shock}} \times \underbrace{0.35}_{\text{(B) } \textit{refuse}} \times \underbrace{0.65}_{\text{(C) shock}} \times \underbrace{0.65}_{\text{(D) shock}} = 0.0961$$

$$\text{Scenario 3: } \underbrace{0.65}_{\text{(A) shock}} \times \underbrace{0.65}_{\text{(B) shock}} \times \underbrace{0.35}_{\text{(C) } \textit{refuse}} \times \underbrace{0.65}_{\text{(D) shock}} = 0.0961$$

$$\text{Scenario 4: } \underbrace{0.65}_{\text{(A) shock}} \times \underbrace{0.65}_{\text{(B) shock}} \times \underbrace{0.65}_{\text{(C) shock}} \times \underbrace{0.35}_{\text{(D) } \textit{refuse}} = 0.0961$$

The probability of exactly one 1 of 4 people refusing to administer the shock is the sum of all of these probabilities.

$0.0961 + 0.0961 + 0.0961 + 0.0961 = 4 \times 0.0961 = 0.3844$

Binomial distribution

The question from the prior slide asked for the probability of given number of successes, k, in a given number of trials, n, (k = 1 success in n = 4 trials), and we calculated this probability as

$$\boxed{\text{\# of scenarios} \times \text{P(single scenario)}}$$

Number of scenarios: there is a less tedious way to figure this out, we'll get to that shortly...

$$P(single \quad scenario) = p^k(1-p)^{(n-k)}$$

The *Binomial* distribution describes the probability of having exactly k successes in n independent Bernouilli trials with probability of success p.

Choose Function

The choose function is useful for calculating the number of ways to choose k successes in *n* trials.

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

For example, :

$$\binom{9}{2} = \frac{9!}{2!(9-2)!} = \frac{9 \times 8 \times 7!}{2 \times 1 \times 7!} = \frac{72}{2} = 36$$

```
choose(9,2)
```

```
## [1] 36
```

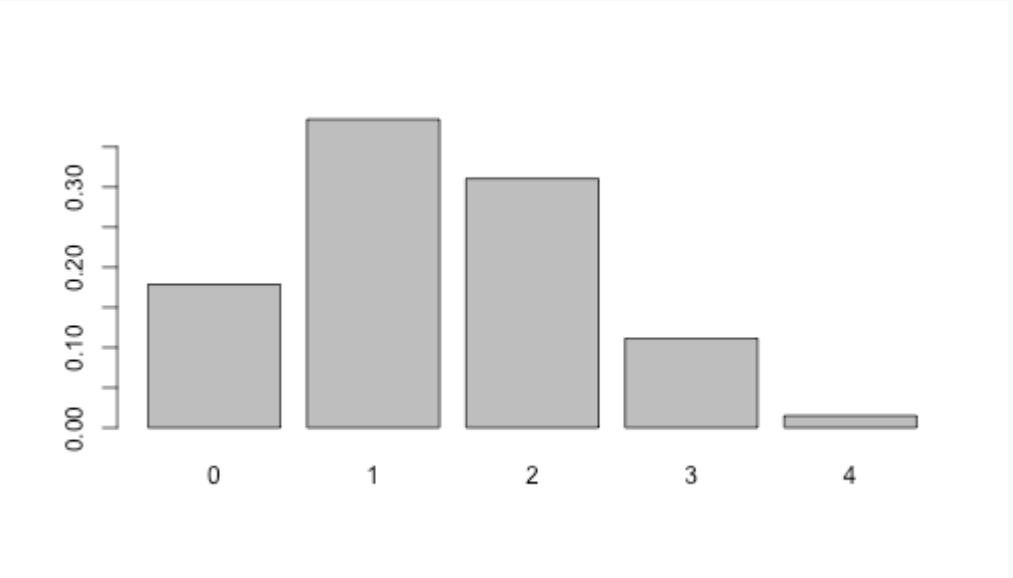
Binomial distribution

If p represents probability of success, (1 - p) represents probability of failure, n represents number of independent trials, and k represents number of successes

$$P(k \quad successes \quad in \quad n \quad trials) = \binom{n}{k} p^k (1-p)^{(n-k)}$$

Binomial distribution

```
n <- 4
p <- 0.35
barplot(dbinom(0:n, n, p), names.arg=0:n)
```




```
dbinom(1, 4, p)
```

```
## [1] 0.384475
```

https://shiny.rit.albany.edu/stat/binomial/

One Minute Paper

1. What was the most important thing you learned during this class?
2. What important question remains unanswered for you?

https://forms.gle/U4UXAosdjHorxY919