

Error in Estimates & Probability Distributions

For loops in the Wild

```
# Create a counter to keep track of data clusters
ext$cluster <- rep(NA, nrow(ext))
ext$cluster[1] <- 1
# iterate over the whole column
for (i in 2:nrow(ext)) {

  # If we've moved on to the next cluster, change the counter
  ext$cluster[i] <- ifelse(ext$effect..[i] < ext$effect..[i -
    1], ext$cluster[i - 1] + 1, ext$cluster[i - 1])
}
```

Sample Properties: Variance

How variable was that population?

$$s^2 = \frac{\sum_{i=1}^n (Y_i - \bar{Y})^2}{n-1}$$

- ▶ **Sums of Squares** over $n-1$
- ▶ $n-1$ corrects for both sample size and sample bias
- ▶ σ^2 if describing the population
- ▶ Units in square of measurement...

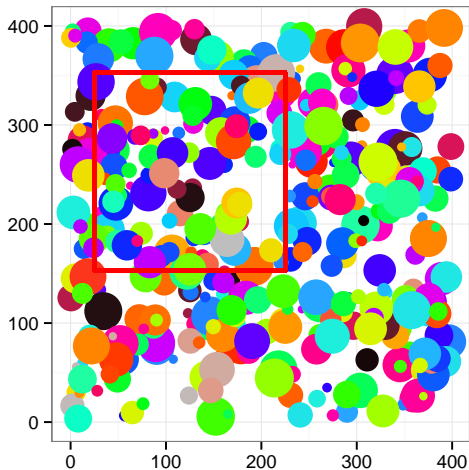
Sample Properties: Standard Deviation

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

- ▶ Units the same as the measurement
- ▶ If distribution is normal, 67% of data within 1 SD
- ▶ 95% within 2 SD
- ▶ σ if describing the population

Remember Samples and Populations?

How representative of our population are the estimates from our sample?



Remember Samples and Populations?

We've seen that we get variation in point estimates at any sample size

What does that variation look like?

Exercise: Variation in Estimation

- ▶ Consider a population with some distribution (rnorm, runif, rgamma)

Exercise: Variation in Estimation

- ▶ Consider a population with some distribution (rnorm, runif, rgamma)
- ▶ Think of the mean of one sample as an individual replicate

Exercise: Variation in Estimation

- ▶ Consider a population with some distribution (`rnorm`, `runif`, `rgamma`)
- ▶ Think of the mean of one sample as an individual replicate
- ▶ Take many (50) 'replicates' from this population of means

Exercise: Variation in Estimation

- ▶ Consider a population with some distribution (`rnorm`, `runif`, `rgamma`)
- ▶ Think of the mean of one sample as an individual replicate
- ▶ Take many (50) 'replicates' from this population of means
- ▶ What does the distribution of means look like? Use *hist*

Exercise: Variation in Estimation

- ▶ Consider a population with some distribution (`rnorm`, `runif`, `rgamma`)
- ▶ Think of the mean of one sample as an individual replicate
- ▶ Take many (50) 'replicates' from this population of means
- ▶ What does the distribution of means look like? Use *hist*
- ▶ How does it depend on sample size (within replicates) or distribution type?

Exercise: Variation in Estimation

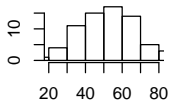
- ▶ Consider a population with some distribution (`rnorm`, `runif`, `rgamma`)
- ▶ Think of the mean of one sample as an individual replicate
- ▶ Take many (50) 'replicates' from this population of means
- ▶ What does the distribution of means look like? Use *hist*
- ▶ How does it depend on sample size (within replicates) or distribution type?

Extra: Show the change in distributions with sample size in one figure.

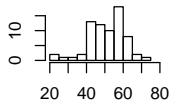
Central Limit Theorem

The distribution of means converges on normality

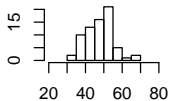
n = 3



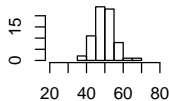
n = 9



n = 15



n = 27



Central Limit Theorem Simulation

```
set.seed(697)
n <- 3
mvec <- rep(NA, times = 100)
# simulate sampling events!
for (i in 1:length(mvec)) {
  mvec[i] <- mean(runif(n, 0, 100))
}
hist(mvec, main = "n=3")
```

Estimating Variation Around a Mean

Great, so, if we can draw many replicated means from a larger population, we can the standard deviation of an estimate!

Estimating Variation Around a Mean

Great, so, if we can draw many replicated means from a larger population, we can the standard deviation of an estimate!

This standard deviation of the estimate of the mean is the **Standard Error**.

Estimating Variation Around a Mean

Great, so, if we can draw many replicated means from a larger population, we can the standard deviation of an estimate!

This standard deviation of the estimate of the mean is the **Standard Error**.

But for a single study, we only have one sample...

A Bootstrap Simulation Approach to Standard Error

- ▶ Our sample is representative of the entire population
- ▶ Therefore, we can resample it *with replacement* for 1 simulated sample
- ▶ We use our sample size as the new sample size as well

We set the replace argument in `sample = TRUE`
Try sampling from the bird data with replacement.

A Bootstrap Simulation Approach to Standard Error

```
sample(bird$Count, replace = T, size = nrow(bird))
```

```
## [1] 23 135 1 23 59 4 67 15 3 1 135 13 152 128  
## [15] 67 148 7 1 3 2 67 1 23 3 300 64 2 282  
## [29] 297 33 297 2 25 128 128 173 14 64 1 33 2 297  
## [43] 282
```

```
sample(bird$Count, replace = T, size = nrow(bird))
```

```
## [1] 297 2 625 230 13 33 25 12 4 28 297 2 12 7  
## [15] 3 1 18 28 297 1 282 15 300 148 23 2 33 1  
## [29] 625 282 77 23 12 25 297 2 2 33 230 135 67 18  
## [43] 77
```

Standard Error

$$SE_{\bar{Y}} = \frac{s}{\sqrt{n}}$$

\bar{Y} - sample mean

s - sample standard deviation

n - sample size

95% Confidence Interval and SE

- ▶ Recall that 95% of the data in a sample is within 2SD of its mean

95% Confidence Interval and SE

- ▶ Recall that 95% of the data in a sample is within 2SD of its mean
- ▶ So, 95% of the times we sample a population, the *true* mean will lie within 2SE of our estimated mean

95% Confidence Interval and SE

- ▶ Recall that 95% of the data in a sample is within 2SD of its mean
- ▶ So, 95% of the times we sample a population, the *true* mean will lie within 2SE of our estimated mean
- ▶ This is the 95% **Confidence Interval**

$$\bar{Y} - 2SE \leq \mu \leq \bar{Y} + 2SE$$

Exercise: 95% Confidence Interval

$$\bar{Y} - 2SE \leq \mu \leq \bar{Y} + 2SE$$

- ▶ Draw 20 simulated samples with $n=10$ from a normal distribution of mean 0

Exercise: 95% Confidence Interval

$$\bar{Y} - 2SE \leq \mu \leq \bar{Y} + 2SE$$

- ▶ Draw 20 simulated samples with $n=10$ from a normal distribution of mean 0
- ▶ Calculate the upper and lower confidence interval for each

Exercise: 95% Confidence Interval

$$\bar{Y} - 2SE \leq \mu \leq \bar{Y} + 2SE$$

- ▶ Draw 20 simulated samples with $n=10$ from a normal distribution of mean 0
- ▶ Calculate the upper and lower confidence interval for each
- ▶ Compare the 95% CIs to the true value of the mean

Exercise: 95% Confidence Interval

$$\bar{Y} - 2SE \leq \mu \leq \bar{Y} + 2SE$$

- ▶ Draw 20 simulated samples with $n=10$ from a normal distribution of mean 0
- ▶ Calculate the upper and lower confidence interval for each
- ▶ Compare the 95% CIs to the true value of the mean
- ▶ Extra: graph it with segments

Tip: To bind two vectors together as columns, use *cbind*

Exercise: 95% Confidence Interval

```
set.seed(697)
n <- 20
upperCIvec <- rep(NA, n)
lowerCIvec <- rep(NA, n)

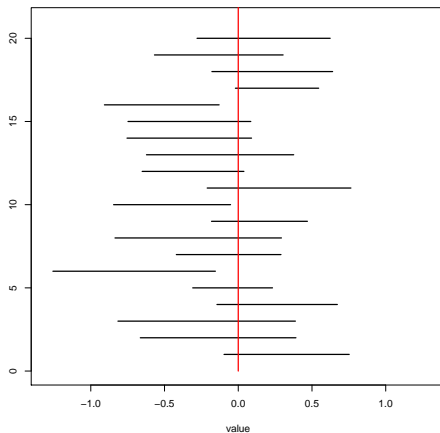
# loop and calculate the 95% CI
for (i in 1:n) {
  samp <- rnorm(10)
  upperCIvec[i] <- mean(samp) + 2 * sd(samp)/sqrt(n)
  lowerCIvec[i] <- mean(samp) - 2 * sd(samp)/sqrt(n)
}
```

Exercise: 95% Confidence Interval

```
# examine the numbers  
cbind(upperCIvec, lowerCIvec)[1:10, ]
```

##		upperCIvec	lowerCIvec
##	[1,]	0.75237	-0.09638
##	[2,]	0.39117	-0.66417
##	[3,]	0.38746	-0.81584
##	[4,]	0.67183	-0.14438
##	[5,]	0.23227	-0.30878
##	[6,]	-0.15508	-1.25684
##	[7,]	0.28960	-0.41992
##	[8,]	0.29285	-0.83584
##	[9,]	0.46890	-0.18128
##	[10,]	-0.05229	-0.84528

Exercise: 95% Confidence Interval



Variation in Other Estimates

- ▶ Many SEs and CIs of estimates have formulae and well understood properties

Variation in Other Estimates

- ▶ Many SEs and CIs of estimates have formulae and well understood properties
- ▶ For those that do not, we can bootstrap the SE of any estimate - e.g., the median

Variation in Other Estimates

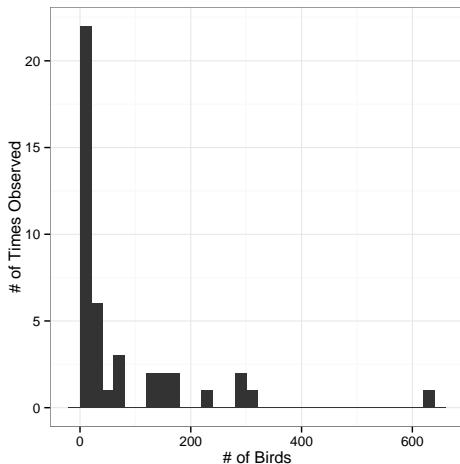
- ▶ Many SEs and CIs of estimates have formulae and well understood properties
- ▶ For those that do not, we can bootstrap the SE of any estimate - e.g., the median
- ▶ Bootstrapped estimates (mean of simulated replicates) can be used to assess bias

Variation in Other Estimates

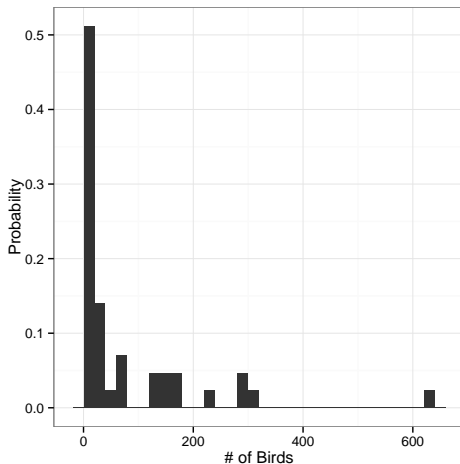
- ▶ Many SEs and CIs of estimates have formulae and well understood properties
- ▶ For those that do not, we can bootstrap the SE of any estimate - e.g., the median
- ▶ Bootstrapped estimates (mean of simulated replicates) can be used to assess bias
- ▶ Bootstrapping is not a panacea - requires a good sample size to start

Distributions!

Frequency Distributions Make Intuitive Sense

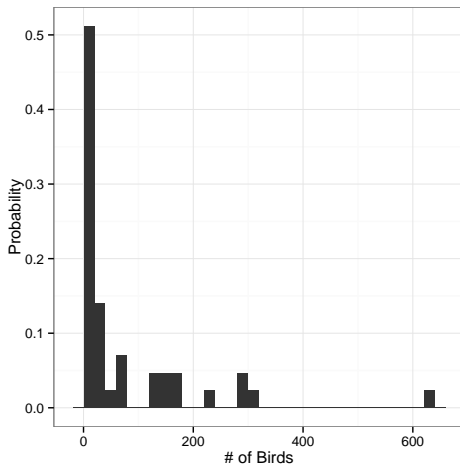


Frequencies Can be Turned Into Probabilities



Just divide by total # of observations

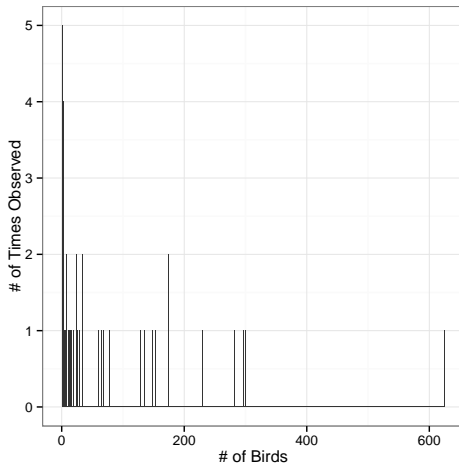
Frequencies Can be Turned Into Probabilities



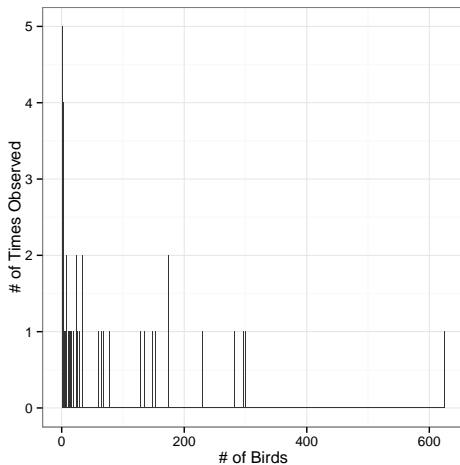
Just divide by total # of observations

But - we have binned observations...

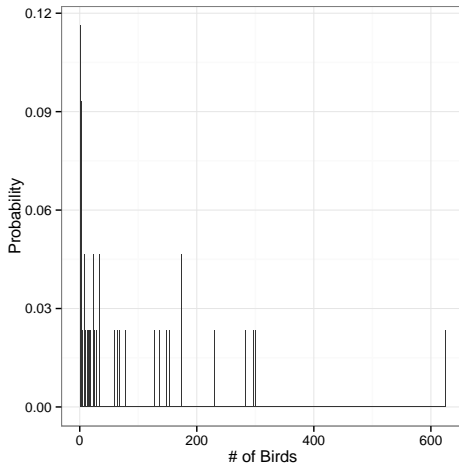
Frequencies of Individual Observations



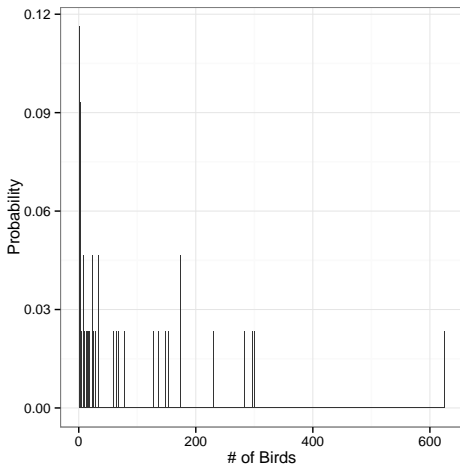
Frequencies of Individual Observations



Probabilities of Individual Measurements

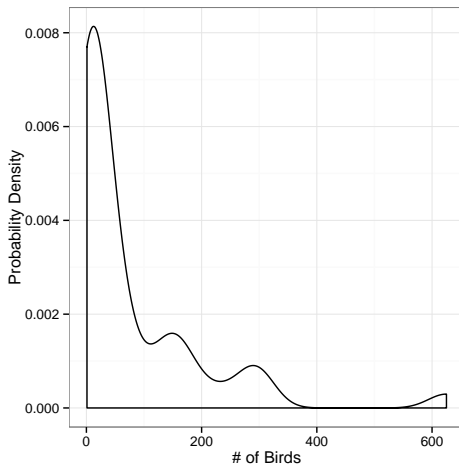


Probabilities of Individual Measurements

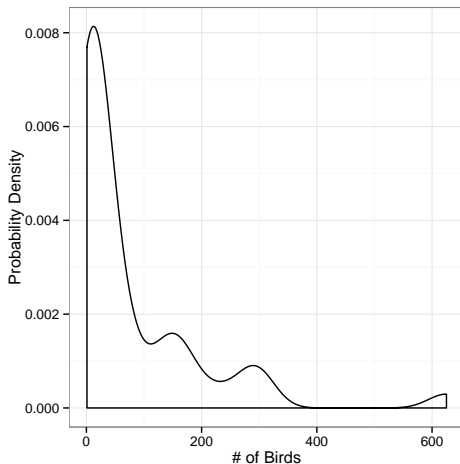


Many probabilities small, and what about the gaps?

Continuous Probability Distributions

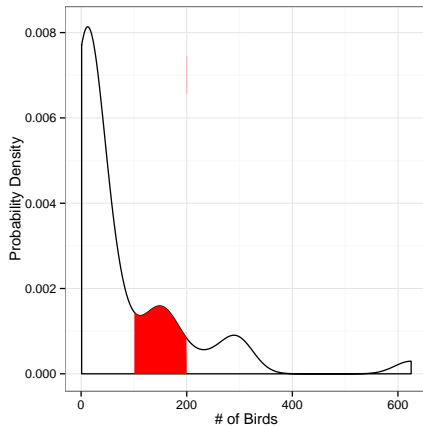


Continuous Probability Distributions

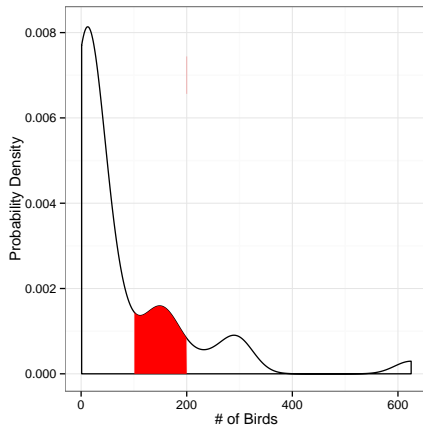


Any individual observation has a *probability density*.

Probability as Integral Under the Curve

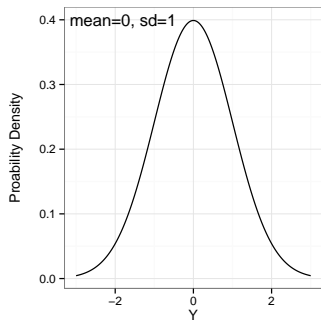


Probability as Integral Under the Curve

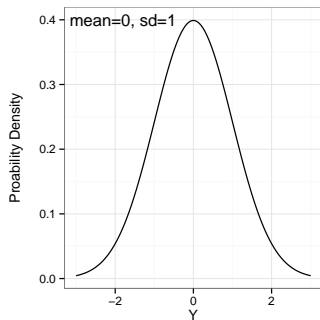


We obtain probabilities of observations between a range of values by integrating the distribution over selected values.

The Normal Distribution

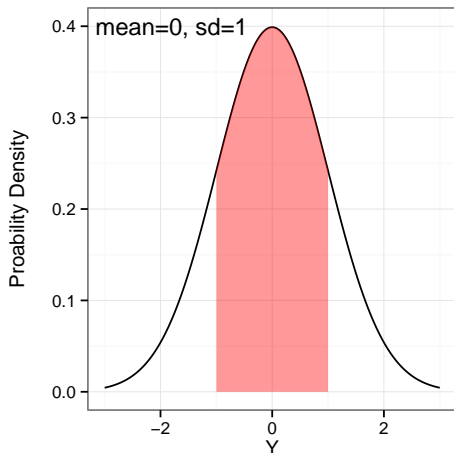


The Normal Distribution

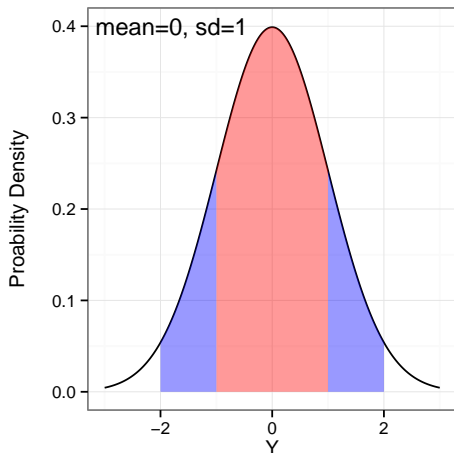


- ▶ Defined by its mean and standard deviation.
- ▶ $Y \sim N(\mu, \sigma)$
- ▶ Single mode
- ▶ Symmetric

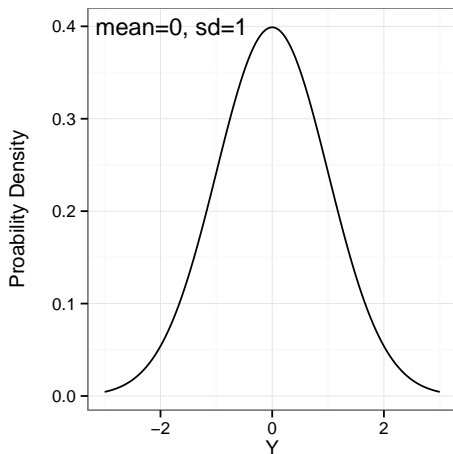
67% of Values within 1 SD



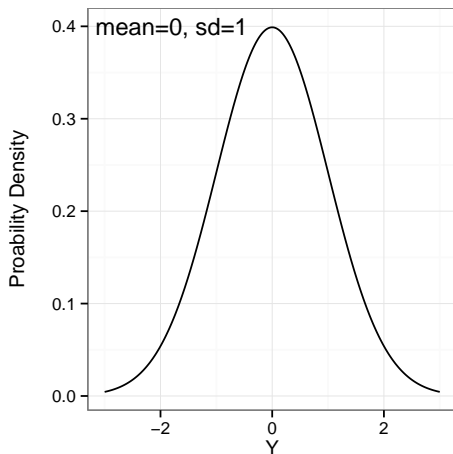
95% of Values within 2 (1.96) SD



How to Get A Probability Density in R

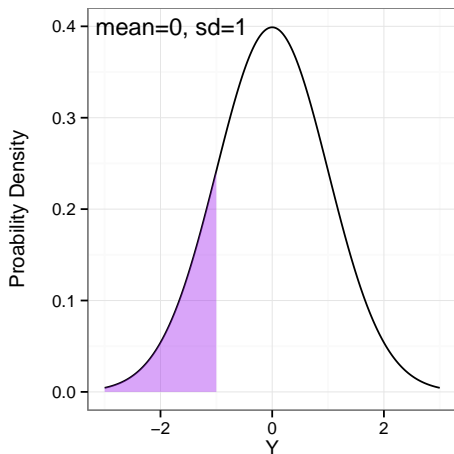


How to Get A Probability Density in R

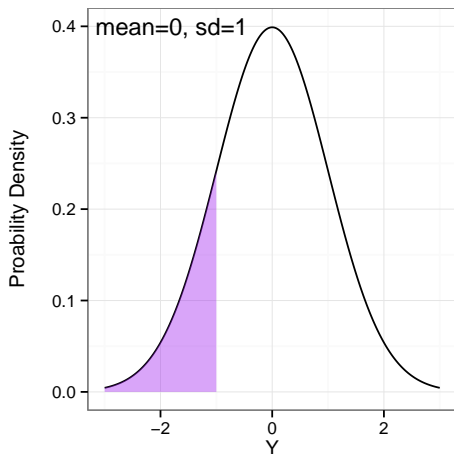


```
dnorm(Y, mean = 0, sd = 1)
```

The Probability of a Value or More Extreme Value

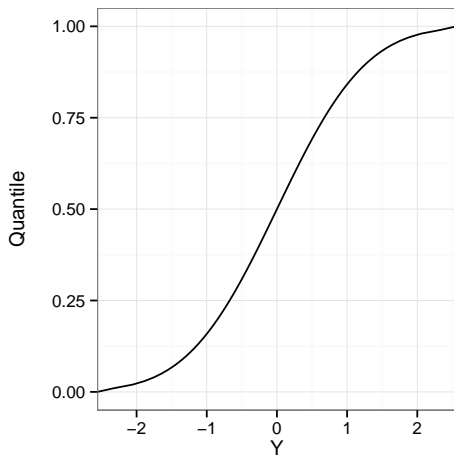


The Probability of a Value or More Extreme Value

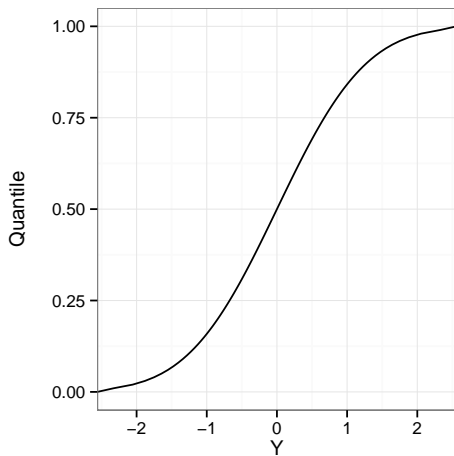


```
pnorm(Y, mean = 0, sd = 1)
```

The Cumulative Distribution/Quantile Function



The Cumulative Distribution/Quantile Function



```
qnorm(p, mean = 0, sd = 1)
```


The Cumulative Distribution/Quantile Function

`pnorm` and `qnorm` are the inverse of one another

```
pnorm(-1)

## [1] 0.1587

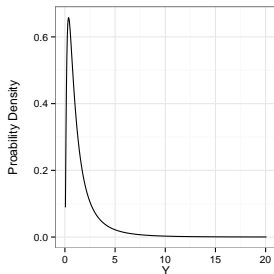
qnorm(pnorm(-1))

## [1] -1

qnorm(0.025)

## [1] -1.96
```

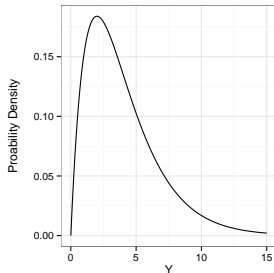
The Lognormal Distribution



- ▶ An exponentiated normal
- ▶ Defined by the mean and standard deviation of its log.
- ▶ $Y \sim \text{LN}(\mu_{\log}, \sigma_{\log})$
- ▶ Generated by multiplicative processes

```
dlnorm(Y, meanlog = 0, sdlog = 1)
```

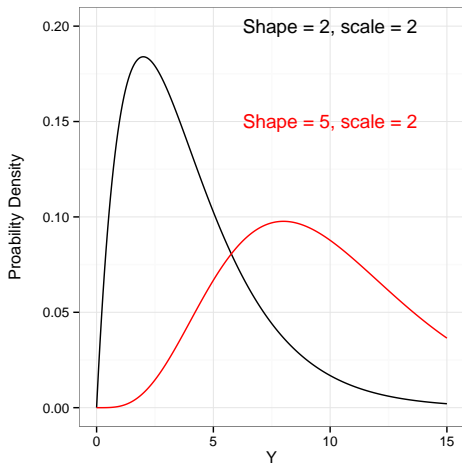
The Gamma Distribution



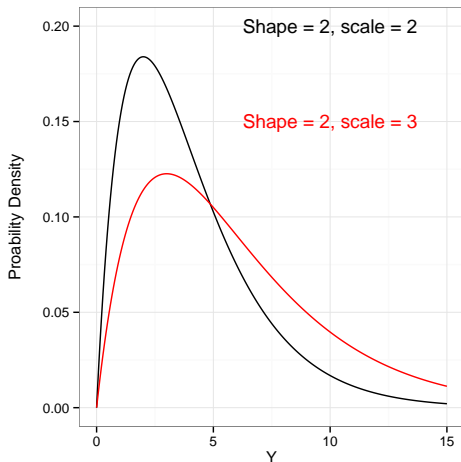
- ▶ Defined by number of events(shape) average time to an event (scale)
- ▶ Can also use rate ($1/\text{scale}$)
- ▶ $Y \sim G(\text{shape}, \text{scale})$
- ▶ Think of time spent waiting for a bus to arrive

```
dgamma(Y, shape = 2, scale = 2)
```

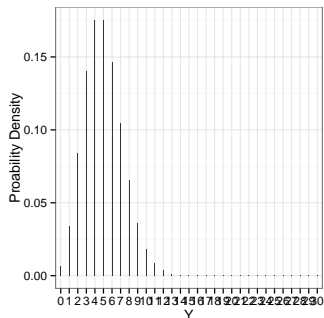
Waiting for more events



Longer average time per event



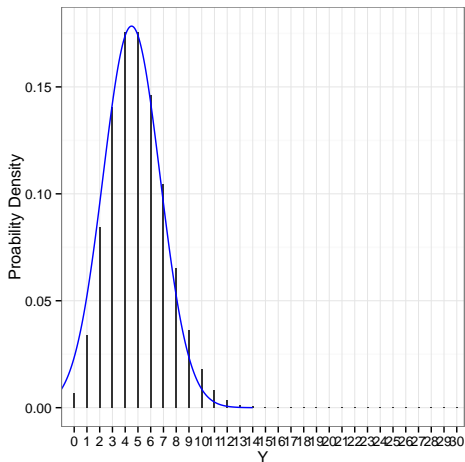
The Poisson Distribution



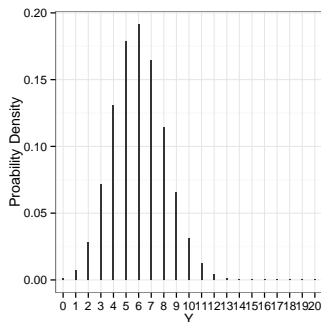
- ▶ Defined by λ - the mean and variance
- ▶ $Y \sim P(\text{lambda})$

```
dpois(Y, lambda = 5)
```

When Lambda is Large, Approximately Normal



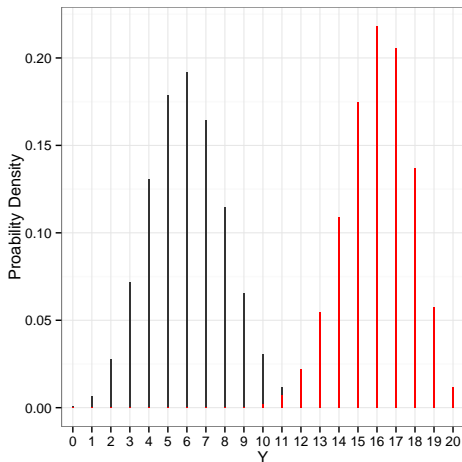
The Binomial Distribution



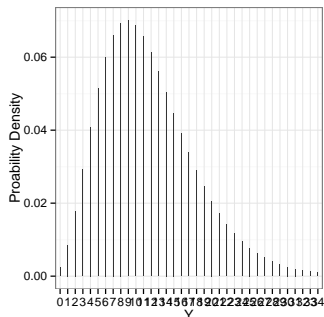
- ▶ Results from multiple coin flips
- ▶ Defined by size (# of flips) and prob (probability of heads)
- ▶ $Y \sim B(\text{size}, \text{prob})$
- ▶ bounded by 0 and size

```
dpois(Y, size, prob)
```


Increasing Probability Shifts Distribution



The Negative Binomial Distribution



- Distribution of number of failures before n number of successes in k trials
- Or mean # of counts, μ , with an overdispersion parameter, size
- $Y \sim \text{NB}(\mu, \text{size})$

```
dnbin(Y, mu, size)
```

Exercise

- ▶ Explore the distributions we have discussed
- ▶ Examine how changing parameters shifts the output of probability function
- ▶ Compare curves generated using density functions (e.g., `dnorm`) and large number of random draws (e.g. from `rnorm`)
- ▶ Overlay these in plots if you can (hist, lines, etc.)
- ▶ Challenge: graphically show integration under the different types of distribution curves