R Lab #9d - Handling violations

Contents

Assumptions of *t*-tests

All of the versions of the *t*-test that we learned have underlying assumptions that should be considered before proceeding with the test.

One-sample *t*-test:

- Data are a random sample from the population
- Variable is normally distributed in the population

Paired *t*-test:

- Sampling units are randomly sampled from the population
- Paired differences have a normal distribution in the population

Two-sample *t*-test:

- Each of the two samples is a random sample from its population
- Variable is normally distributed in each population
- Standard deviation of the variable is the same in both populations

In some cases violations to these assumptions can be ignored (see Chap 13 and flow charts on Canvas), but you may also need to transform your data or run an alternative non-parametric test.

Deviations from normality: marine reserve example

Chapter 13 covers an example comparing biomass in marine reserves versus non-reserve controls. In this study the researchers made a ratio of their samples. Thus, the null hypothesis is that the sample of ratios has a mean of 1 (i.e., no difference in biomass in reserves compared to controls). Let's explore this example in R.

Import and inspect data

Import the data and plot the sample of ratios:

```
marineData <- read.csv("chap13e1MarineReserve.csv", stringsAsFactors = T)
str(marineData)
## 'data.frame':    32 obs. of  1 variable:
##  $ biomassRatio: num  1.34 1.96 2.49 1.27 1.19 1.15 1.29 1.05 1.1 1.21 ...
```

We see that there are 32 rows (observations) and 1 column in the data frame. The one column is the sample of biomass ratios.

Plot data to visually inspect for normality

It's generally a good idea to plot the data for visual inspection.

An assumption of the two-sample *t*-test is that each sample has a normal distribution. Generally, it is useful to graph the data to see if it appears normal.

First, let's make frequency distribution histograms of the ratios:

```
library(ggplot2)
ggplot(marineData, aes(x = biomassRatio)) + geom_histogram(color = "black",
    fill = "darkred") + theme_classic()
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
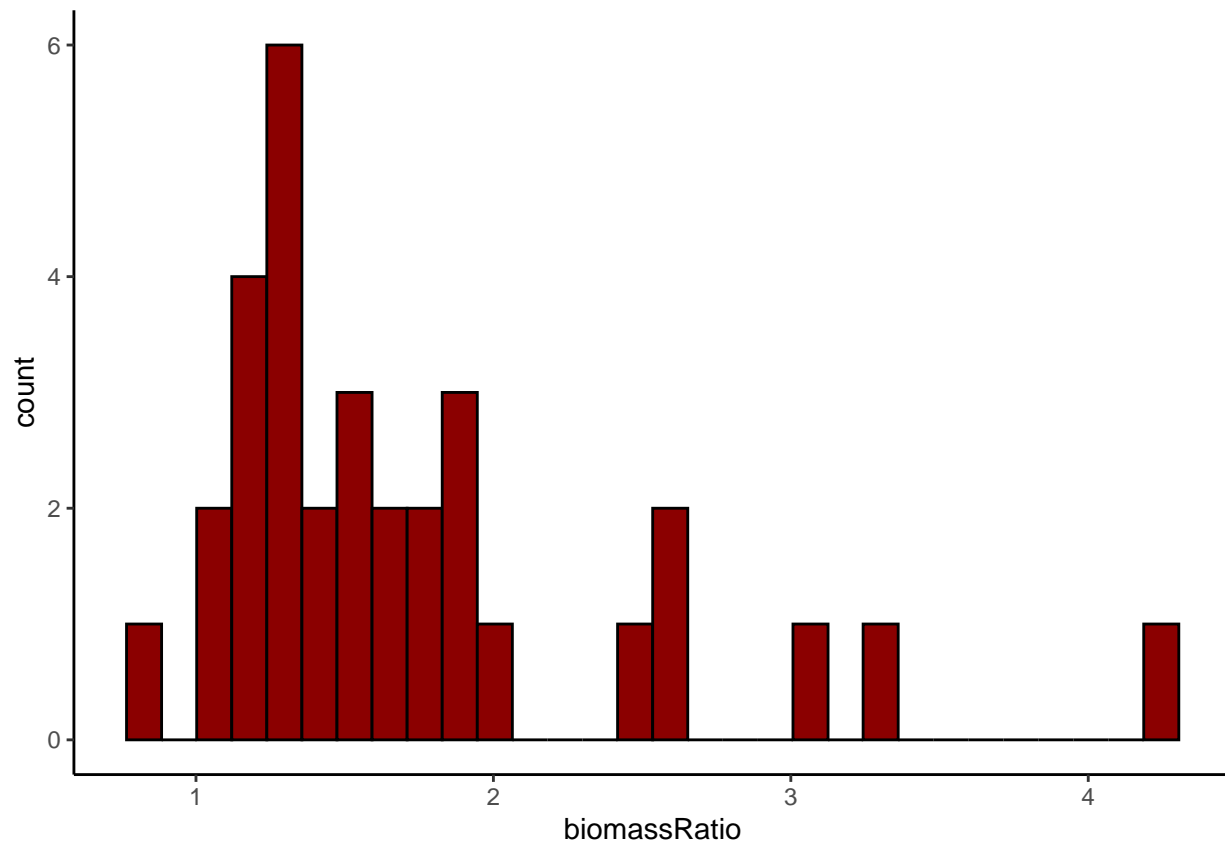
Now a box plot:

```
ggplot(marineData, aes(x = "", y = biomassRatio)) + geom_boxplot(width = 0.25) +
    theme_classic()
```
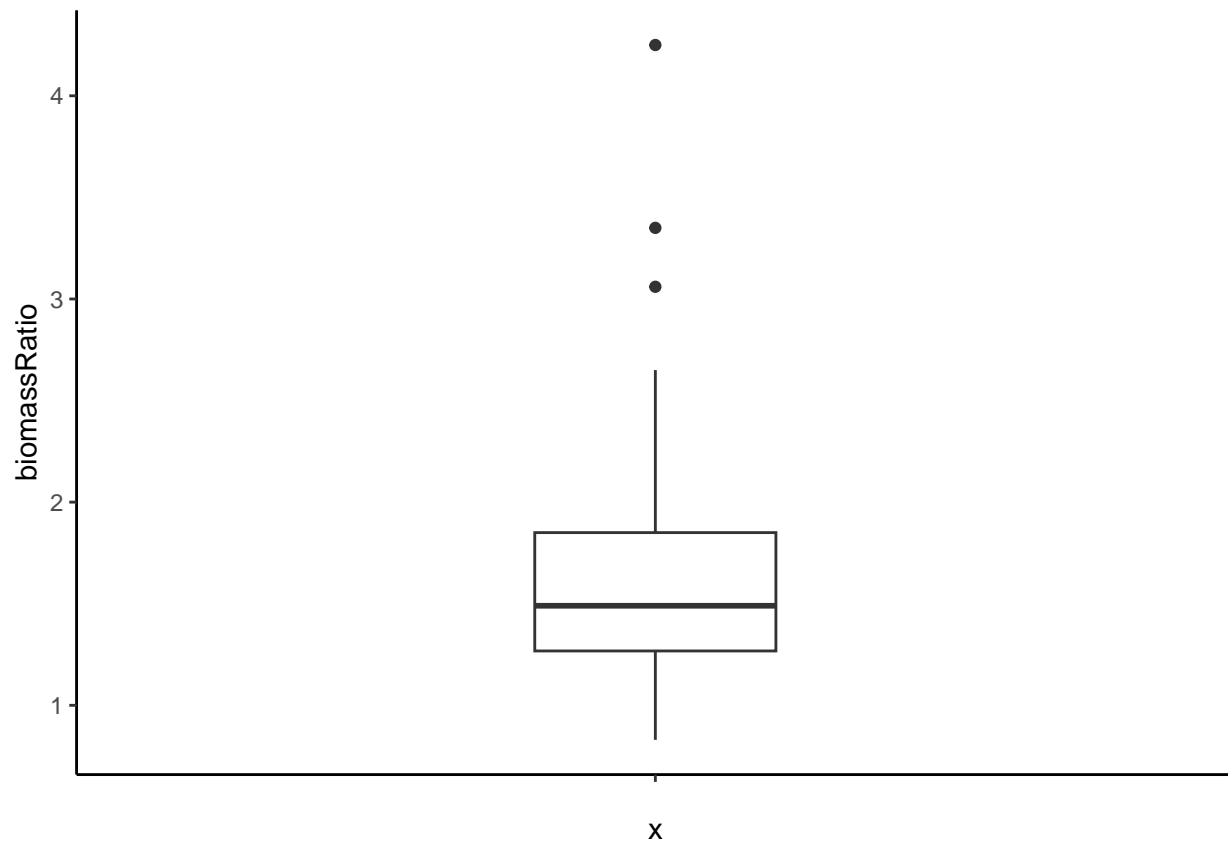
Now a normal quantile plot:

```r
library(dplyr)
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
qqnorm(marineData$biomassRatio, datax = T)
qqline(marineData$biomassRatio, datax = T)
```

Normal Q–Q Plot
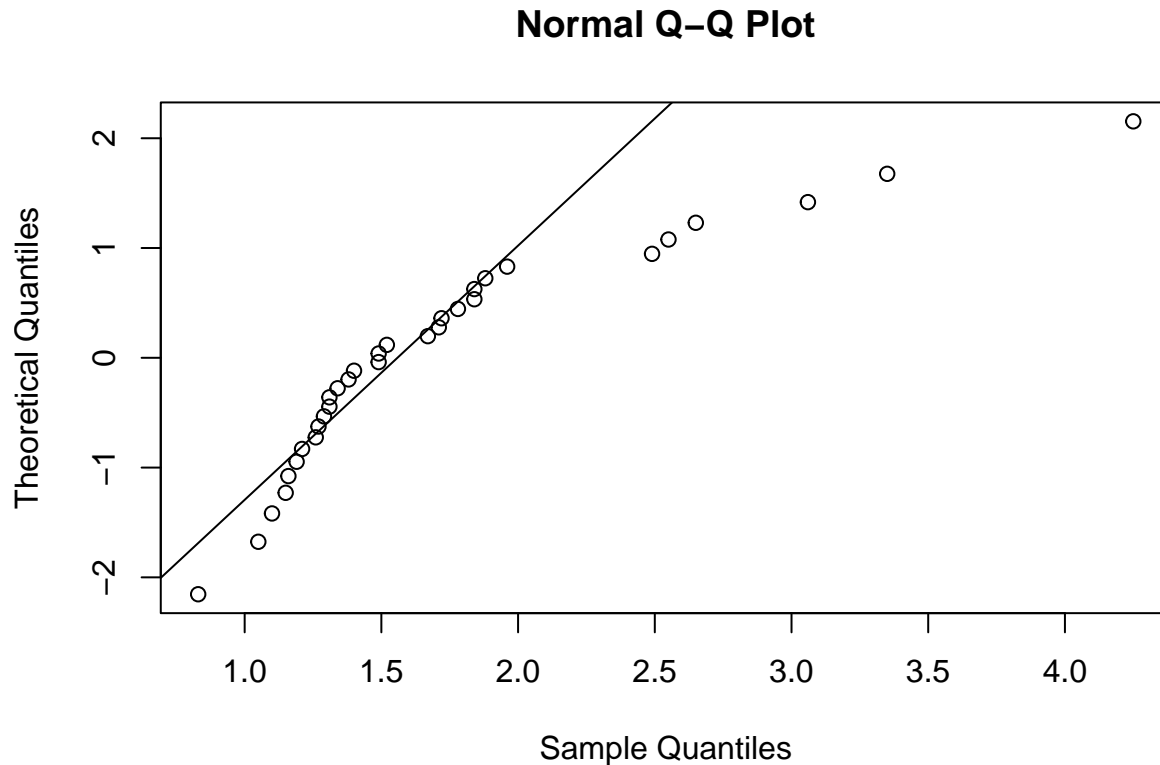
All of these plots show that there is right-skew.

Shapiro-Wilk test

The plots show that there is right-skew of the sample of ratios. Now we want to see if the data is normally distributed with shapiro.test().

```
shapiro.test(marineData$biomassRatio)
##
##  Shapiro-Wilk normality test
##
## data:  marineData$biomassRatio
## W = 0.81751, p-value = 8.851e-05
```

The P-value of 8.851e-05 means that we **reject** the null hypothesis of normality.

Data transformation

Since the sample of ratios do not have a normal distribution we should not run a one-sample t-test on these data. One option is to transform the data in an effort to make it more consistent with a normal distribution. The most common transformation for a sample with right-skew is a natural log transformation. This is done with log().

Let's create a new column with natural log transformed ratios and then re-run the Shapiro-Wilk test:

```
marineData$lnBiomassRatio <- log(marineData$biomassRatio)
shapiro.test(marineData$lnBiomassRatio)
##
##  Shapiro-Wilk normality test
##
## data:  marineData$lnBiomassRatio
## W = 0.93795, p-value = 0.06551
```

Now we get a *P*-value above 0.05, so we **fail to reject** the null hypothesis of normality.

We can visualize the difference by making a frequency distribution histogram of the transformed ratios, which looks more normal:

```
library(ggplot2)
ggplot(marineData, aes(x = lnBiomassRatio)) + geom_histogram(color = "black",
    fill = "darkred") + theme_classic()
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
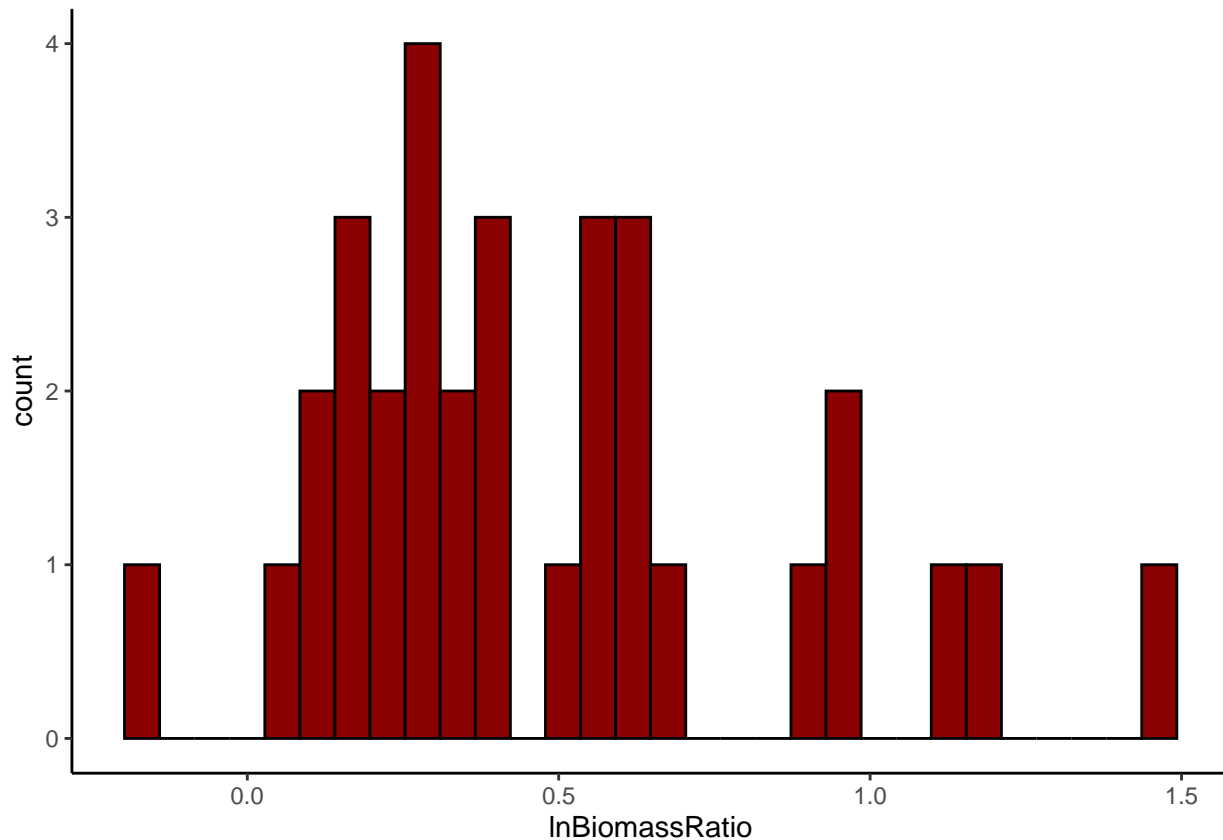

The histogram now looks more normal and the Shapiro-Wilk test confirms that we now have a normal distribution.

Running the one-sample *t*-test on transformed data

Now we can run a one-sample *t*-test. Note, however, that the null hypothesis also needs to be transformed for the test. For the untransformed data the null for the ratio is 1 ($\mu=1$). The natural log of 1 is 0, so we run the test with a null mean of 0 ($\mu'=0$).

```
t.test(marineData$lnBiomassRatio, mu = 0)
##
##  One Sample t-test
##
## data:  marineData$lnBiomassRatio
## t = 7.3968, df = 31, p-value = 2.494e-08
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.3470180 0.6112365
## sample estimates:
```

```
## mean of x
## 0.4791272
```

The P-value is 2.494e-08, so we **reject** the null hypothesis that the mean natural log transformed ratio is 0.

Back-transforming the 95% CI

The test results also shows the 95% confidence interval for the mean natural log transformed ratios. Typically, one back-transforms the confidence interval so the units are consistent with the original data.

```
t.test(marineData$lnBiomassRatio, mu = 0)$conf.int
## [1] 0.3470180 0.6112365
## attr(,"conf.level")
## [1] 0.95
exp(t.test(marineData$lnBiomassRatio, mu = 0)$conf.int)
## [1] 1.414842 1.842708
## attr(,"conf.level")
## [1] 0.95
```

Here we see that the 95% CI estimate is consistent with the reserves (numerator of ratio) having 1.4-1.8x the biomass of the controls.

Nonparametric sign test: insect sexual conflict example

What do you do in a paired design if the assumption of normality is violated?

At this point you need to switch to a nonparametric test that does not rely on sample mean or standard deviation statistics.

In this example the distribution of differences of the number of species in multiple minus single-mated groups had an outlier to the far fight.

Import and inspect the data:

```
conflictData <- read.csv("chap13e4SexualConflict.csv", stringsAsFactors = T)
str(conflictData)
## 'data.frame':    25 obs. of  4 variables:
##  $ taxonPair           : Factor w/ 25 levels "A","B","C","D",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ nSpeciesMultipleMating: int   53 73 228 353 157 300 34 3400 20 196 ...
##  $ nSpeciesSingleMating  : int   10 120 74 289 30 4 18 3500 1000 486 ...
##  $ difference          : int   43 -47 154 64 127 296 16 -100 -980 -290 ...
```

The data frame has 25 rows (observations), with separate columns for each variable in the pair (nSpeciesMultipleMating and nSpeciesSingleMating) and the differences.

Test for normality.

Let's just look at a histogram of the sample differences:

```
ggplot(conflictData, aes(x = difference)) + geom_histogram(color = "black",
    fill = "darkred") + theme_classic()
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
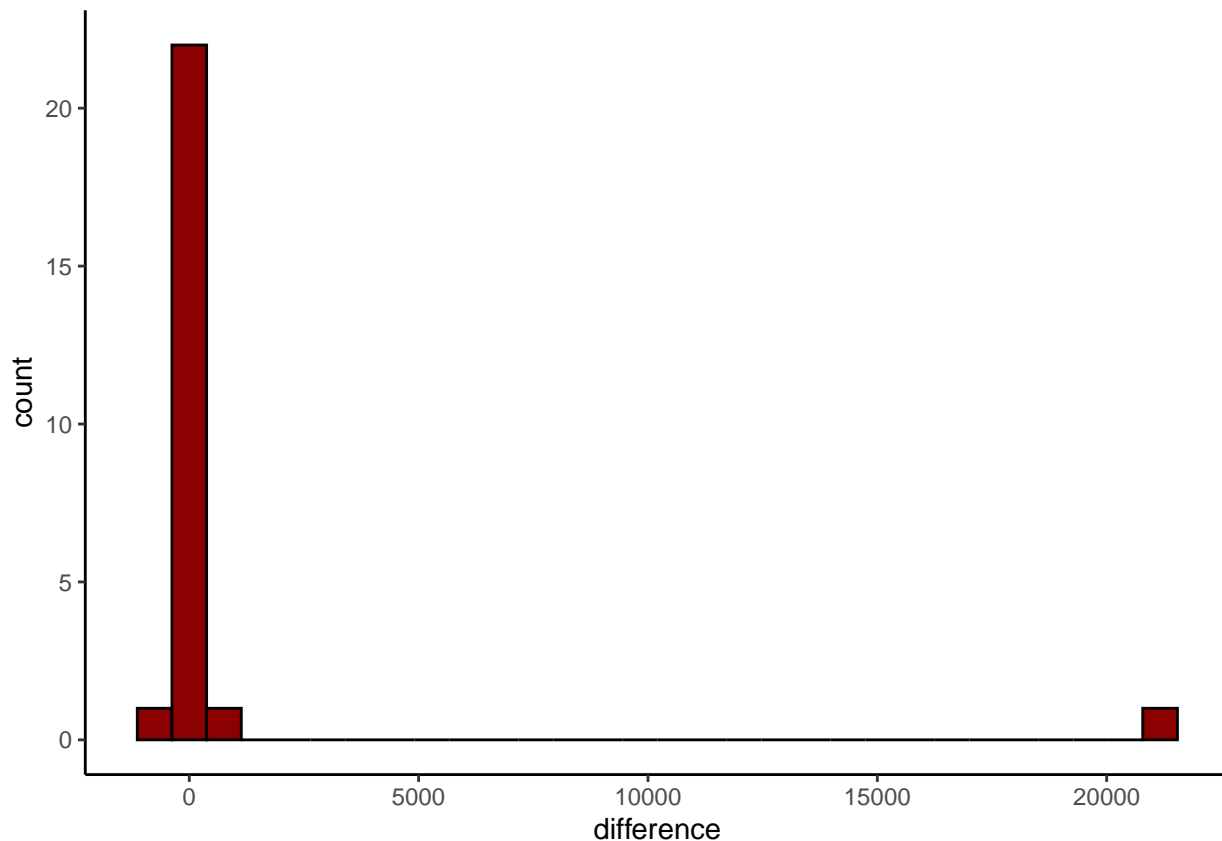
This plot shows the extreme outlier to the right, and it is not likely that this sample conforms to a normal distribution. Let's check formally:

```
shapiro.test(conflictData$difference)
##
##  Shapiro-Wilk normality test
##
## data:  conflictData$difference
## W = 0.25473, p-value = 2.911e-10
```

The sample of differences has a *P*-value below 0.05, so we **reject** the null hypothesis of normality in both cases.

Running the nonparametric sign test

Our sample of differences has both positive and negative numbers, and you cannot compute a natural log of a negative number. Thus, it is not possible to run a natural log transformation on the sample of differences.

Although it's possible to explore looking at the data as a ratio rather than differences (see below), the textbook uses this study to demonstrate the nonparametric sign test, which tests of an equal probability of values being above (+) and below (-) a null value. If the number of matings has no effect on the number of species in these groups then our null value is zero and we expect an equal probability of a positive or negative value.

To run the sign test in R we use binom.test(x,n,p) – recall Chapter 7. Here, we can classify a positive difference as a "success." To run the test we need to count the number of positive differences (x), know the sample size (n), and use a null probability of success of 0.5 (p).

We can get the sample size by computing the length of the sample of differences:

```
length(conflictData$difference)
## [1] 25
```

We can get the number of positives by first filtering the data frame (using filter() in package dplyr) and then getting the length of the difference column in the filtered data frame:

```
library(dplyr)
positive <- filter(conflictData, difference > 0)
length(positive$difference)
## [1] 18
```

Note that you can do this more efficiently with one command:

```
length(filter(conflictData, difference > 0)$difference)
## [1] 18
```

So the sample size is 25 and there are 18 positive differences. However, you should check to see if there are values that are exactly zero. If this is the case then these observations are not counted. That is, if there was one difference that was exactly zero then that observation is removed and the sample size would go down to 24.

```
length(filter(conflictData, difference == 0)$difference)
## [1] 0
```

This confirms that there are no differences that are exactly zero, and you can go ahead and run the sign test with 18 positives in a sample of 25. Again, this is done with binom.test():

```
binom.test(18, 25, 0.5)
##
##  Exact binomial test
##
## data:  18 and 25
## number of successes = 18, number of trials = 25, p-value = 0.04329
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.5061232 0.8792833
## sample estimates:
## probability of success
##                   0.72
```

With a P-value below 0.05, we **reject** the null hypothesis of an equal probability of positive and negative values. Since we have more positives than expected, and the difference was computed as species in multiple-mated minus single-mated groups, you conclude that the multiple-mated groups have significantly more species.

Note that if you characterize negative values as a "success" then you get the exact same P-value. This is because 7 and 18 are the same distance away from the expected number of successes (12.5):

```
binom.test(7, 25, 0.5)
##
##  Exact binomial test
##
## data:  7 and 25
## number of successes = 7, number of trials = 25, p-value = 0.04329
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.1207167 0.4938768
## sample estimates:
## probability of success
```

```
##                    0.28
```

Transform and re-test for normality

The textbook goes right to the nonparametric sign test after the assumption of normality is violated. However, it might be possible to transform the data.

Usually we try a natural log transformation, but our sample of differences has negative numbers and thus this transformation is not possible. So instead let's work with a sample of ratios instead of differences.

```
conflictData$ratio <- conflictData$nSpeciesMultipleMating/conflictData$nSpeciesSingleMating
```

Let's first check if the sample of ratios follows a normal distribution:

```
shapiro.test(conflictData$ratio)
##
##  Shapiro-Wilk normality test
##
## data:  conflictData$ratio
## W = 0.30174, p-value = 6.722e-10
```

As expected with the extreme outlier, normality is also rejected for the sample of ratios. Now we can try a natural log transformation of the sample of ratios, and then test if the transformed ratios follow normality:

```
conflictData$lnRatio <- log(conflictData$ratio)
shapiro.test(conflictData$lnRatio)
##
##  Shapiro-Wilk normality test
##
## data:  conflictData$lnRatio
## W = 0.94195, p-value = 0.1642
```

With a P-value of 0.16, we **fail to reject** normality for the sample of transformed ratios. Thus, we can run a one-sample t-test on the transformed ratios. Typically when we test a sample of ratios the null hypothesis is $\mu=1$. However, since our ratios have been natural log transformed the μ' is 0 (because the natural log of 1 is 0).

```
t.test(conflictData$lnRatio, mu = 0)
##
##  One Sample t-test
##
## data:  conflictData$lnRatio
## t = 2.4217, df = 24, p-value = 0.02337
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##   0.1346864 1.6887209
## sample estimates:
## mean of x
## 0.9117036
```

Since the P-value is below 0.05, we **reject** the null hypothesis that the mean natural log ratio is 0. Since the 95% confidence interval is above zero, we conclude that the numerator (number of species in multiple-mated group) is significantly higher.

Note that the P-value here is lower than what we got for the sign test. This demonstrates that the parametric t-test has more power compared to the nonparametric sign test.

Nonparametric Mann-Whitney U-test: cricket cannibalism example

What do you do in a two-sample design if transforming the data doesn't make it normal?

At this point you need to switch to a nonparametric test that does not rely on sample mean or standard deviation statistics.

In this example the time to mating was compared between crickets in which females were fed or starved.

Import and inspect the data:

```
cricketData <- read.csv("chap13e5SagebrushCrickets.csv", stringsAsFactors = T)
str(cricketData)
## 'data.frame':    24 obs. of  2 variables:
##  $ feedingStatus: Factor w/ 2 levels "fed","starved": 2 2 2 2 2 2 2 2 2 2 ...
##  $ timeToMating : num  1.9 2.1 3.8 9 9.6 13 14.7 17.9 21.7 29 ...
```

The data frame has 24 rows (observations) and 2 columns, one marking the treatment (fed or starved) and one with the time to mating.

Test for normality.

Let's just look at the stacked frequency distribution histograms:

```
ggplot(cricketData, aes(x = timeToMating)) + geom_histogram(color = "black",
    fill = "darkred") + facet_wrap(~feedingStatus, ncol = 1) +
    theme_classic()
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
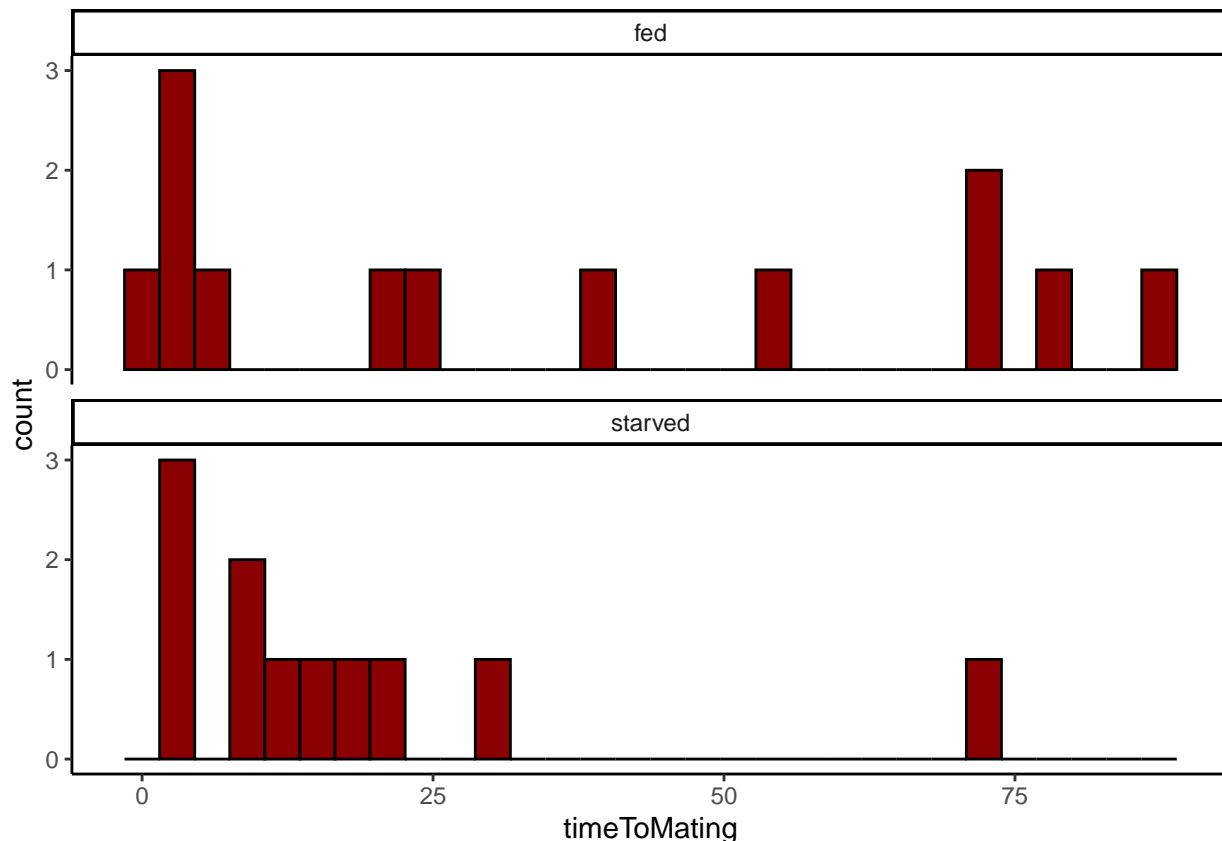
Both samples appear skewed to the right, but we should test for normality *separately* for each sample with the Shapiro-Wilk test. There are multiple ways to run the test separately for each group (see RLab8supp_TwoSample_test).

```
tapply(cricketData$timeToMating, cricketData$feedingStatus, shapiro.test)
## $fed
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.85641, p-value = 0.03455
##
##
## $starved
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.73256, p-value = 0.001232
```

Both samples have a *P*-value below 0.05, so we **reject** the null hypothesis of normality in both cases.

Since both samples appear to be skewed in the same direction we might be justified in ignoring this violation, but the sample sizes are too small in this case (see Chapter 13).

Transform and re-test for normality

Now we could try transforming the data and re-testing for normality.

First let's try a natural log transformation:

```
cricketData$lnTimeToMating <- log(cricketData$timeToMating)
tapply(cricketData$lnTimeToMating, cricketData$feedingStatus,
    shapiro.test)
## $fed
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.85592, p-value = 0.03405
##
##
## $starved
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.95664, p-value = 0.7291
```

Now we **fail to reject** the null of normality for the starved group, but we still **reject** normality for the fed group. Since both transformed samples do not have a normal distribution we should ***not*** run a two-sample *t*-test on the transformed data.

Running the nonparametric Mann-Whitney *U*-test

The best option now is the Mann-Whitney U-test. This test first ranks all of the data, and then compares the distribution of ranks of the two groups. The null hypothesis is that the two groups have the same distribution

of ranks.

The test uses a U-statistic, which is calculated with the command wilcox.test(). Note that the command actually runs a "Wilcoxan rank sum exact test", but the W-statistic shown in the output is actually the same as the U-statistic in the Mann-Whitney test.

```r
wilcox.test(cricketData$timeToMating ~ cricketData$feedingStatus)
##
##  Wilcoxon rank sum exact test
##
## data:  cricketData$timeToMating by cricketData$feedingStatus
## W = 88, p-value = 0.3607
## alternative hypothesis: true location shift is not equal to 0
```

The P-value here is 0.3607, so we **fail to reject** the null hypothesis that the two samples have equal distributions of ranks. This result is consistent with no differences between the two samples.

R commands summary

- **Natural log transformation**
  - log(x)
- **Sign test (run as binomial test)**
  - binom.test(x,n,p=0.5)
- **Mann-Whitney U-test**
  - wilcox.test(y~x)