ST 411/511 Lab 4
Rank-Sum Test, Welch's t-Test, Sign Test

Objectives for this Lab

- Perform a Wilcoxon rank-sum test and obtain a confidence interval.
- Do a Welch's t-test, which does not require the assumption that the two populations have the same standard deviation.
- Perform a sign test on the twin hippocampus data.

1. As usual, start up RStudio and open Lab4.R. Load the Sleuth3 and ggplot2 R packages.

```
> library(Sleuth3)
> library(ggplot2)
```

2. R's `wilcox.test()` function performs Rank-Sum tests for two-sample comparisons when the normality assumption is violated (Section 4.2 in the textbook).

   The Rank-Sum test is a permutation/randomization test. The data are "rank-transformed." An observation's rank is its position in the data set that has been sorted from smallest to largest. The Rank-Sum test statistic is the sum of the ranks in one of the groups. The observed ranks are repeatedly randomly reallocated to the two groups. Each random reallocation gives a value of the test statistic. The distribution of all these test statistic values is the sampling distribution of the test statistic under the null hypothesis of no difference between the two populations.

   It doesn't matter which group's ranks are summed to get the test statistic. This is because the sum of *all* the ranks is the sum of the numbers $1, \ldots, n$ where n is the total sample size, and we know a formula for that:

$$1 + 2 + \ldots + n = \sum_{i=1}^{n} i = \frac{n(n+1)}{2}.$$

   Let's do a rank-sum test to analyse the cognitive load data from Section 4.1.2.

   (a) View the data frame.

```
> View(case0402)
```

   This data frame contains three columns named `Time, Treatment,` and `Censored`. The last column contains a 0 for the *uncensored* observations (those times less than 300) and a 1 for the *censored* observations. Censoring is common in studies that observe times-to-event, such as survival times for transplant patients.

   (b) To clarify what is meant by a "rank transformation," bind the columns of the data frame together with the rank-transformed times.

```
> cbind(case0402, rank(case0402$Time))
```

   (c) Check side-by-side boxplots.

```
> qplot(Treatment, Time, data=case0402, geom="boxplot")
```

These don't look so bad, until you notice that the Conventional boxplot lacks an upper whisker, due to the censoring. Checking histograms reveals very skewed distributions.

```
> qplot(Time, data=case0402, geom="histogram", facets=Treatment~.)
```

Aside: This code is slightly different than the code on page 5 of Outline 3. Here, the faceting formula is inside the `qplot()` command, and in Outline 3, it's *added*. The effect is the same. If you want to play around with facets, try: `facets=Treatment~Censored`. The `ggplot2` package is capable of making *very* fancy graphical displays. If you want to delve further, consider taking ST 537 Data Visualization in the spring. It's an online course, and ST 512 is a prerequisite.

(d) We want to do a rank-sum test of the hypothesis that there is no difference between the groups. Since the research question is whether the "Modified" group had lower times than the "Conventional" group, this should be a one-sided test, so first check a summary of `Treatment` to discover how R orders the groups.

```
> summary(case0402$Treatment)
```

Since the "Conventional" group is first, the alternative is that times from the first group are more likely to be *greater* than times from the second group.

Note that `summary()` also gives the sample sizes in the two groups.

(e) Do the rank-sum test, specifying that the alternative is that the "Conventional" group will have higher times.

```
> wilcox.test(Time~Treatment, data=case0402, alternative="greater",
+             exact=FALSE, correct=FALSE)
```

Setting `exact=FALSE` gives a p-value based on the normal approximation (Section 4.2.3; Display 4.7), which means the permutation distribution is approximated by a normal distribution with mean and variance known from the theory of ranks. Setting `correct=FALSE` means a continuity correction is not applied. (The default is `correct=TRUE`, so if you omit `correct=FALSE`, you will get the continuity correction.)

Compare your output with Display 4.7 on page 93. You should have almost the same p-value. Why is it not exactly the same?

Also, R's W statistic doesn't match $T = 137$, calculated in Display 4.5. The relationship between W and T is $W = T - n_1(n_1 + 1)/2$ where n_1 is the number of observations in the first sample ("Conventional" for R but "Modified" for the *Sleuth*). This is the smallest possible value for T (i.e. if its ranks are $1, 2, \ldots, n_1$).

(f) Here is R code to do the test with "Modified" as the first sample. Instead of giving `wilcox.test()` a formula, give it the two samples as `x=` and `y=` arguments.

```
> with(case0402,wilcox.test(x=Time[Treatment=="Modified"],
+                           y=Time[Treatment=="Conventional"],
+                           alternative="less",
+                           correct=FALSE,
+                           exact=FALSE))
```

Note the alternative hypothesis here is that the modified times will be less than the conventional times. Find the test statistic W in the output. Obtain sample sizes n_1 and n_2 from the output of `summary()` above, and verify that R's test statistic W is equal to $T - n_1(n_1 + 1)/2$ where $T = 137$ is the book's test statistic.

(g) To obtain the confidence interval reported in Display 4.8, we have to make the restrictive assumption that there's a single parameter δ that represents the difference between treatments (i.e. if Y is a student's time after studying the conventional materials, then $Y - \delta$ is the time if the student had instead studied using the modified materials). Run a two-sided `wilcox.test()` and specify `conf.int=TRUE`.

```
> wilcox.test(Time~Treatment, data=case0402, exact=FALSE, correct=FALSE,
+             conf.int=TRUE)
```

The resulting interval is not quite as reported in Display 4.8, but it is very close. The *Sleuth* only considers integer δ.

3. Perform a Welch's t-test to test if two population means are different when the assumption of equal variance is not met.

We will use the finch data from Chapter 2 for illustration, even though we have no reason to doubt the equal-variance assumption. Perform Welch's t-test. This uses our old friend `t.test()` but without the `var.equal=TRUE` option. Welch's t-test is the default for `t.test()`.

```
> t.test(Depth~Year, data=case0201) # Not assuming equal variance
```

Notice that the two-sided p-value is 8.739×10^{-6} whereas *with* `var.equal=TRUE`, it is a bit smaller. This illustrates a general principle: when the assumptions *are* met, the t-tools make better use of the information in the sample. Unfortunately, we never know for sure if the assumptions are met.

The degrees of freedom are `df = 172.98` instead of `df = 176`. With Welch's t-test, the degree of freedom are not sample size minus number of mean parameters. This is because Welch's t-test is an approximate t-test. The formula to calculate these approximate degrees of freedom is shown on page 98 of the textbook. We will not calculate this "by hand."

You can find the formula for $\text{SE}(\overline{Y}_1 - \overline{Y}_2)$ also on page 98 of the textbook. It looks like the formula on page 41 of the textbook, except it doesn't use the pooled standard deviation s_p. This make sense because we wouldn't want to get a pooled estimate of the population standard deviation when we're not assuming the two populations have the same standard deviation.

4. A sign test is a non-parametric alternative to a paired t-test. As discussed at the beginning of Section 4.4.1, the test statistic is the number of positive differences K between the paired responses. If the null hypothesis that the median difference is 0 is true, this statistic has a binomial distribution with probability of success 0.5, that is, we'd expect about half of the differences to be positive and half to be negative. Therefore, to perform this test in R, we use `binom.test()`.

(a) First, calculate the differences.

```
> diffs <- with(case0202, Unaffected-Affected)
```

(b) Count how the number of pairs altogether as well as the number of positive differences.

```
> length(diffs)
> length(which(diffs>0))
```

This tells you there are 15 pairs and 14 of the differences are positive. That's a lot more than half. Calculate a p-value to ascertain the strength of evidence against the null hypothesis using `binom.test()`. We will do a one-sided test because that's what's done on page 100 of the text. However, when we first met this case study, the research question seemed to be two-sided ("is there a difference?" not "is there a positive difference?").

```
> binom.test(14, 15, alternative="greater")
```

You should get an exact p-value of 0.0004883. "Exact" means no normal approximation was used (see page 100 in the *Sleuth*).

Note: The R output gives a confidence interval. This is a confidence interval for the *probability* of a positive difference.