Lecture 20

Computational Methods

We've touched on some these methods in some past lectures and labs, but we are going to spend some dedicated time to addressing them in more depth. The two main techniques we are going to consider is bootstrapping methods and permutation tests, and in particular, where the sample sizes are large. Permutation tests, in particular, where the sample sizes are large, are not feasible for doing an exact approach, but can be approximated relatively easily through sampling methods. Bootstrapping has some similarities, but we will spend some more time discussing underlying assumptions for bootstrapping and looking at cases it can address that fill in some gaps in theoretical models.

Procedurally, permutation tests and bootstrapping are quite similar but there are some key differences.

Permutation tests:
When we sample, we must choose a sample size that is the same size as the original data set, and we must do it without replacement. We can use the order of selection determine which sample the data falls into.

Bootstrapping methods:
When we sample, we can choose any sample size we like, and we choose those samples with replacement (that means values can be repeated).

What they have in common is that the test statistic we are modeling with our test can be common statistics we model with other traditional methods, or it can be less-common statistics such as the median or mode. We can use these methods to conduct hypothesis testing or we can use it to construct confidence intervals. We can make fewer assumptions about our data, or the shape of the sampling distribution.

Some limitations do exist for such methods (eg. for infinite variance, certain discontinuities) and sometimes we may need to account for particular types of bias, but these situations are mostly uncommon.

Let's look at some examples of bootstrapping and permutation tests. While we have looked at simple cases earlier in the course, we are going to look at some slightly more complex cases. The first will be to return to some data we used in an example in a previous lecture for the two-sample Wilcoxon test.

Example.
A study of guinea pigs tests the effects of orange juice vs. synthetic ascorbic acid on odontoblasts. The data is below.

Orange Juice	8.2	9.4	9.6	9.7	10.0	14.5	15.2	16.1	17.6	21.5
Ascorbic Acid	4.2	5.2	5.8	6.4	7.0	7.3	10.1	11.2	11.3	11.5

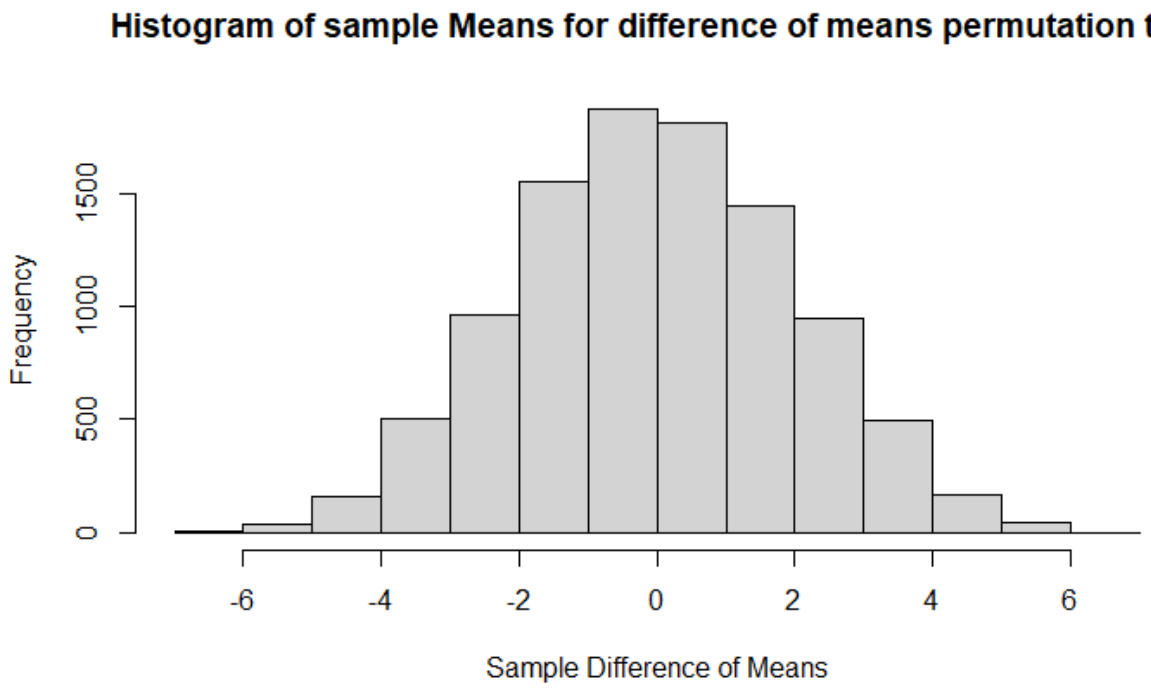
For the permutation test, we will group the data into a simple sample, and then randomly select observations to put into group 1 and group 2. Then calculate the difference of the means in the two groups. There are $\binom{20}{10} = 184{,}756$ different combinations to make group 1. We can sample from these

different configurations. If the number of samples are large enough, then this will be a good approximation to doing a complete permutation. A sample of 10,000 or 100,000 will be sufficient. There are packages in R for supporting bootstrapping and permutation tests, but we won't use those here to see how the methods work internally (though you are free to use them for homework or other assignments as you wish).

```
37  diffs <- c()
38  N=10000
39  n=20
40  for(i in 1:N) {
41    sample <- sample_n(data,n,replace=FALSE)
42    mean1 <- mean(sample$measure[1:10])
43    mean2 <- mean(sample$measure[11:20])
44    diff <- mean1-mean2
45    diffs <- c(diffs, diff)
46  }
```

Recall that we noted above the permutation tests require that you use all the data available with no repetition (we are rearranging the data we have—that's why this is sometimes called a randomization test). Here, I randomly selected the data to scramble the order (or grouping). Then I calculated the mean of the first 10 observations, and then the mean of the second 10 observations. Then I calculated the difference of those means. 10,000 samples take a minute or two on my computer to run. We can then make a histogram to look at the data.

For randomly selected data, it's not surprising that the difference of the means centers around 0.



The difference of means for the original data is 5.18. You can see that that value is way off in the right tail. This seems like the probability is small, but we need to actually calculate it, not guess about it. Filter your list of differences to contain only those that are equal to or greater than the original difference of means. The ratio of the length of that vector to the number of original samples is your P-value for the one-tailed test. Twice that value is the P-value for the two-tailed test.

```
49
50  diffs1<-data.frame(diffs)
51  k <- filter(diffs1, diffs >= 5.18)
52  p_val <- length(k)/N
53  p_val
54
```

End up with the number of permutations in our sample with a difference of means bigger than 5.18 is 19, so our P-value for the one-sample test is $\frac{19}{10{,}000}$ or about 2×10^{-4}. This is much less than the usual significance level of 0.05 so we reject the null hypothesis that the two means are equal. This is a much stronger result the P-value obtained for the Wilcoxon rank-sum test.

If we want to construct a confidence interval, we can use the data we've collected to find the margin of error. While our distribution looks roughly symmetric, it does not have to be. Suppose we want a 95% confidence interval. That means we want 2.5% in each tail which is 250 observations in each tail. We want the value of the difference of means at the 250th observation, and the 9751st observation after sorting the list of differences.

```
53
54  diffs_sorted<-sort(diffs)
55  diffs_sorted[250]
56  diffs_sorted[9751]
57
```

The lower bound is -3.84 and the upper bound is 3.88. Our distribution is centered around the null hypothesis. We can shift the confidence interval to be centered at the observed mean by adding 5.18. Our 95% confidence interval is thus approximately $(1.34, 9.06)$.

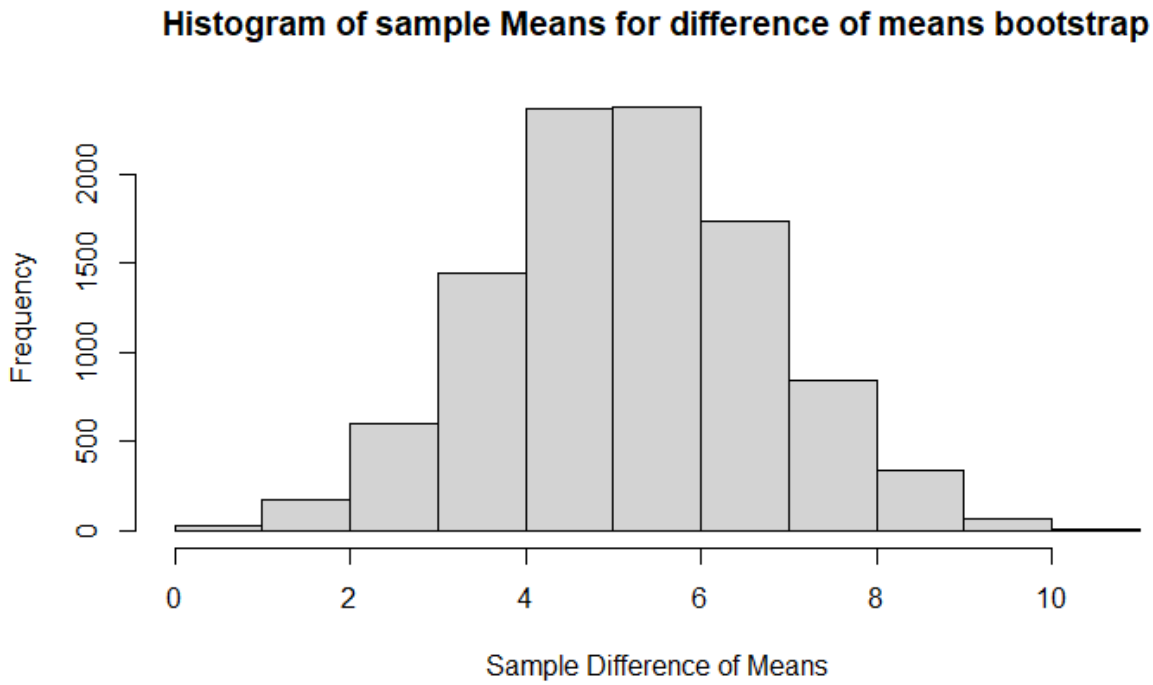
(Note: we didn't set the seed here, so your values will likely be a little bit different than mine, but they should be similar.)

If you wanted to test this result with a larger number of permutations, you can try using 100,000. But you'll see that while the computation will take much longer the P-value obtained will not change by that much.

How does this differ from a bootstrapping procedure? For one thing, we would not combine our two samples into one. Instead, we would select bootstrap samples (of any sample size since we are permitting repetition) from each of the two separate datasets. And then calculate the resulting difference of means. While we can use any sample size, I will select a sample of 10 for our example, with 10,000 trials.

```
59  diffs <- c()
60  N=10000
61  n=10
62  for(i in 1:N) {
63    sample1 <- sample_n(data1,n,replace=TRUE)
64    sample2 <- sample_n(data2,n,replace=TRUE)
65    mean1 <- mean(sample1$measure)
66    mean2 <- mean(sample2$measure)
67    diff <- mean1-mean2
68    diffs <- c(diffs, diff)
69  }
```

We can plot the results with a histogram.



To construct a confidence interval, we can do much as we did before although we don't need to adjust the center of our interval. We need to find the bottom 2.5% and top 2.5% value for our 95% confidence interval. Doing so gives us the confidence interval $(2.14,8.33)$. This is a little narrower than our previous interval, but we are making slightly stronger assumptions here than with the permutation test. Nonetheless, it's not that different.

To conduct a hypothesis test, we can simple check to see if the assumption of the null hypothesis is inside the confidence interval. It's not, which means that the null hypothesis can be rejected. We could also look for 0 in our differences of means to find a more exact P-value.

We find the that smallest difference is actually 0.07, so no observations are less than 0. Thus, we can say that the p-value is approximately less than 1×10^{-4}.

We can do the same for other kinds of tests. The main difference will be calculating the sample statistic in the loop. Some sample statistics are harder to calculate than others, but we just have to set it up once and less R do the rest. We'll get to try it out more in the lab this week.

Next, we'll look at χ^2-tests for various kinds of categorical or binned data, our last big topic for the semester.

References:

1. https://faculty.ksu.edu.sa/sites/default/files/probability_and_statistics_for_engineering_and_the_sciences.pdf
2. https://statisticsbyjim.com/hypothesis-testing/bootstrapping/
3. https://www.mastersindatascience.org/learning/introduction-to-machine-learning-algorithms/bootstrapping/
4. https://towardsdatascience.com/bootstrapping-statistics-what-it-is-and-why-its-used-e2fa29577307
5. https://online.stat.psu.edu/stat555/node/119/
6. https://data-flair.training/blogs/bootstrapping-in-r/