Lecture 5

Caden Hewlett

2024-01-19

The Permutation Test

A good alternative to the Wilcoxon Rank-Sum test when we have many observations that are tied. As all non-parametric methods, it is good for small amounts of non-normal data. This will be exemplified later.

In this case, we follow the theory of the Permutation Test.

Theory and Process

If the data are sufficiently small, we can compute the "full" permutation test.

When inspecting the null hypothesis: $H_0 : \mu_A = \mu_B$ in data with ties.

The general theory behind the permutation test is the intuition that if A and B have the same mean, then the difference of means $\bar{d} = \bar{y}_A - \bar{y}_B$ should be the same **regardless** of the division into samples A and B.

So if we were to combine the data and take a random sample of size n_A, and the remaining data as the sample of size n_B, then the observed difference of means $\bar{d}_2 = \bar{y}_{A_2} - \bar{y}_{B_2}$ should be basically the same. To compute this, we first combine the data.

Step 1 combine Data A and B into a single vecor $Z = \{A, B\}$.

Step 2 we let m be the possible number of ways to draw a sample of size n_A from Z.

$$m = \binom{n_A + n_B}{n_A}$$

If this number is reasonable, we calculate all possible samples of size n_A.

Step 3 We then generate all possible samples of size n_A from Z, which I will call $\mathcal{Z}$, where $|\mathcal{Z}| = m$.

You can think of each $\vec{z_i} \in \mathcal{Z}$ as a sample of size n_A randomly pulled from the pooled data. Therefore, as an interesting aside, $A \in \mathcal{Z}$.

Step 4 We would then be able to compute the mean of the "held out" set of size n_B, too. For each $\vec{h}_j \in \mathcal{H}$, we can compute the j-th mean of the held-out group $\bar{h}_i$ by:

$$\bar{\mathcal{H}} = \left\{ \forall z_i \in \mathcal{Z}, i = [1, 2, \ldots, m], \bar{h}_i = \frac{1}{n_b} \left(\left(\sum_{\ell=1}^{n_A + n_B} \mathrm{z}'_\ell \right) - \sum_{j=1}^{n_A} z_{i,j} \right) \right\}$$

Where z'_ℓ is the ℓ-th element in the *original pooled data* and $z_{i,j}$ is the i-th data point of the j-th element of $\mathcal{Z}$. This seems complex, but, as we'll see, it isn't too bad to compute

Step 5 Then we calculate the sample difference for each calculation, and store them in a vector called $\mathcal{D}$, where, still, $|\mathcal{D}| = m$

$$\mathcal{D} = \left\{(\bar{z}_i - \bar{h}_i), i \in [1, m]\right\}$$

Step 6 We then count the number of observations $\bar{d}_i \in \mathcal{D}$ that are as extreme or more extreme than our original $\bar{d}_{\text{obs}}$, and we denote this as a constant called c_1.

$$c = \sum_{d_i \in \mathcal{D}} \mathbb{I}\big(|d_i| \geq |d_{\text{obs}}|\big)$$

Where $\mathbb{I}$ is the indicator function.

Step 7 Then the p-value is simply the ratio of c to the number of combinations m.

$$p = \frac{c}{m}$$

Note that other methods count the number of "greater than d_{obs}" as c_1 and the number of "equal to d_{obs}" as c_2, letting the numerator be $c_1 + 0.5c_2$.

Worked Example

We will run through an example showing all these steps together.

Consider the following set of data:

```
# Vector for men from textbook example
men <- c(5, 7, 12, 14, 14, 14, 18, 21, 22, 23, 24, 25, 34, 37, 47, 49, 64, 67, 69, 125, 192, 483)
# Vector for from textbook example
women <- c(229, 453)
```

Then, we compute $\bar{d}_obs$.

```
dbar <- mean(men)-mean(women)
dbar
```

```
## [1] -278.9091
```

Step 1 Now we pool the data.

```
allData <- c(men, women)
```

Step 2 Now, we calculate n_A, n_B and m.

```
n1p = length(men); n2p = length(women);
m = choose((n1p+n2p),n1p)
m
```

```
## [1] 276
```

Step 3 Then we can generate $\mathcal{Z}$, all the possible combinations of size n_A, and their means.

```
perm <- combn(length(allData), n1p) # n1p x m matrix
all_samples <- matrix(allData[perm], ncol = m)
sum_sample_j <- apply(all_samples, MARGIN=2, sum)
```

Step 4 Then we can calculate $\mathcal{H}$, in just one line of code.

```
sum_sample_j_complement <- (sum(allData) - sum_sample_j)/n2p
```

Step 5 Now we find $\mathcal{D}$ using the $\mathcal{H}$ and $\mathcal{Z}$ vectors.

```
dj_vals = (sum_sample_j/n1p) - sum_sample_j_complement
```

Step 6 Now we count the number of $\bar{d}_j$ values greater in absolute thaan $\bar{d}_{\text{obs}}$.

```
c1 = sum(abs(dj_vals)>=abs(dbar))
c1
```

```
## [1] 3
```

Step 7 Finally we calculate the p-value.

```
p = c1 / m
p
```

```
## [1] 0.01086957
```

Then, you'd choose to reject or fail to reject H_0 depending on your chosen level of α.

This concludes Lecture 5.