

# Statistical Computing HW 5

*Mario Ibanez*

*February 29, 2016*

## Assignment

Consider a repeated measure experiment with three treatments as follows:

Subject	A	B	C
1	10	11	13
2	13	14	17
3	6	8	9

Use a randomization test with the systematic data permutations to test:

$$H_o : \mu_A = \mu_B = \mu_C$$

## Solution

As was mentioned in class, the equivalent test is an ANOVA  $F$  test for repeated measures. Because each treatment has (the same) three subjects, and after removing other constant terms, the equivalent test statistic that will be used to compare different permutations is:

$$\sum_{i=1}^k T_i^2$$

where

$$T_i = \sum_{j=1}^n y_{ij}$$

Basically,  $T_i$  is the sum for treatment  $i$ . These sums need to be calculated for each treatment, squared, and then added up. Permutations will be along the rows, in other words, for each patient, the three values for the treatments will be permuted. The p-value is calculated by:

$$\text{p-value} = \frac{\text{number of permutations that have a test statistic greater than the original}}{\text{total number of permutations}}$$

For the original data set, the test statistic is:

```
calculate_stat <- function(x){  
  return(sum(apply(X = x, MARGIN = 2, FUN = sum)^2))  
}  
original <- calculate_stat(data_matrix); original
```

```
## [1] 3451
```

The original test statistic that other permutations will be compared to is 3451.

Since in the question  $k = 3$  and  $n = 3$ , there are  $(k!)^{n-1} = (3!)^2 = 36$  total permutations that need to be checked. The first row does not need to be permuted. If we were to permute the first row, it would just lead to six duplicates of each test statistic calculated by leaving the first row fixed.

We do not have duplicate values in the rows and do not need to take this issue into account. The code below calculates the permutations of the second and third rows, and then iterates through all combinations and calculates the test statistic for each case.

```
# Calculate all the permutations for rows 2 and 3
permutations <- apply(X = data_matrix[2:3, ], MARGIN = 1, permn)
# Initialize vector for store test statistics
vector_of_statistics <- c(rep(0, 36))
# Double for loop to calculate the test statistic for each combination
for(i in 1:length(permutations[[1]])){
  for(j in 1:length(permutations[[2]])){
    temp_matrix <- matrix(c(data_matrix[1,], permutations[[1]][[i]],
                           permutations[[2]][[j]]),
                        nrow = 3, byrow = TRUE)
    vector_of_statistics[6*(i-1) + j] <- calculate_stat(temp_matrix)
  }
}
```

Finally, we can calculate what proportion of the test statistics have a value greater than or equal to the original data set's:

```
# Mean of zeroes and ones
p_value <- mean(vector_of_statistics >= original)
p_value
```

```
## [1] 0.02777778
```

The p-value is 0.0277778. This is because in the data set, Treatment C contains the three highest values in each row, Treatment B contains the second largest values in each row, and Treatment A contains the smallest number in each row. There is no other arrangement of data that produces a larger sum of squared column totals. Thus at a  $\alpha = 0.05$  significance level we reject the null hypothesis and conclude that the population means of the three treatments are not all equal to each other.