Introduction to Hypothesis Testing: Permutation Tests

```
library(fastR2)
library(knitr)
library(resampledata)
```

Introduction to Hypothesis Testing

In this notebook we introduce one of the key components of inferential statistics, hypothesis tests. We will closely follow the presentation from Chapter 3 of *Mathematical Statistics with Resampling and R* by Chihara and Hesterberg.

The biggest challenge with starting to learn about hypothesis testing is that there is initially a lot of terminology to digest. We will start with a very simple motivating example that should help to develop some intuition for what is going on beofre we dive into learning the necessary nomenclature.

A Motivating Problem

Suppose scientists invent a new drug that supposedly will inhibit a mouse's ability to run through a maze. In order to test this hypothesis, the scientists design an experiment in which three randomly chosen mice are given the drug and another three are given a placebo in order to compare between a control group and a drug group. All six mice are placed in a maze and the time it takes each mouse to complete the maze is recorded. The collected data looks as follows:

```
mouse_data <- data.frame(times=c(30,25,20,18,21,22),
                         group=c(rep("Drug",3),rep("Control",3)),
                         row.names = c("Mouse1","Mouse2","Mouse3","Mouse4","Mouse5","Mouse6"))
kable(mouse_data)
```

	times	group
Mouse1	30	Drug
Mouse2	25	Drug
Mouse3	20	Drug
Mouse4	18	Control
Mouse5	21	Control
Mouse6	22	Control

A convenient way to summarise this data is to compute the mean time from the three mice in each of the two groups:

```
mouse_data %>% group_by(group) %>% summarise(mean_time=mean(times))
```

```
## # A tibble: 2 x 2
##   group   mean_time
##   <fct>       <dbl>
## 1 Control      20.3
## 2 Drug         25
```

We can go even one step further and turn this into a single number by computing the mean difference in

times:

```
data_summary <- mouse_data %>%
  group_by(group) %>%
  summarise(mean_time=mean(times))
data_summary$mean_time %>% unlist() %>% diff()
```

```
## [1] 4.666667
```

This is an example of an observed test statistic.

Let's actually turn this into an R function for future convenience.

```
mean_diff <- function(mouse_data){
  data_summary <- mouse_data %>%
  group_by(group) %>%
  summarise(mean_time=mean(times))
  mean_diff_val <- data_summary$mean_time %>% unlist() %>% diff()
  return(mean_diff_val)
}
```

We can check that our function works:

```
(mean_diff_obs <- mean_diff(mouse_data))
```

```
## [1] 4.666667
```

Here is the question we seek to answer via statistical inference:

Question: Is the observed value of 4.67 in the difference between the mean maze times for the two groups of mice due simply to natural random variablility, or is it because there is some real effect?

Here is a simple way to address this question:

Observation: If the observed difference of 4.67 in the mean maze times between the two groups of mice **is** due to natural random variability, then the group labeling of Drug versus Control for the six mice is arbitrary and should not matter.

Another way to think of this is as follows:

1) Consider the mice as a single group of six.
2) Randomly assign three to the "Control" group and three to the "Drug" group.
3) Compute the mean difference in times again.
4) Repeat this process for all the ways that we can possibly assign three to "Control" and three to "Drug", that is, consider all possible permutations of the six mice where after each permutation we separate the six into two groups of three.
5) In the absence of any real effect, a value as or more extreme than 4.67 in the mean maze times between the two groups should not be a rarity. If a value as or more extreme than 4.67 in the mean maze times between the two groups is rare, then there is a high probability that the difference is due to a true effect.

This is the essence of a so-called **permutation test**.

Let's look at some values for the mean difference in maze times between the two groups for some permutations of the maze times for the six mice:

```
time_vals <- c(30,25,20,18,21,22)
mouse_data_perm <- data.frame(times=sample(time_vals,replace = F),
                              group=c(rep("Drug",3),rep("Control",3)),
```

```
                              row.names = c("Mouse1","Mouse2","Mouse3","Mouse4","Mouse5","Mouse6"))
kable(mouse_data_perm)
```

	times	group
Mouse1	20	Drug
Mouse2	30	Drug
Mouse3	21	Drug
Mouse4	25	Control
Mouse5	22	Control
Mouse6	18	Control

Notice that each time we rerun this code, we get a different ordering in the list of times. Furthermore, if there is no real effect, then the ordering should not really matter very much. Let's recompute the mean difference in times for the permuted data:

```
(mean_diff_val_perm <- mean_diff(mouse_data_perm))
```

```
## [1] 2
```

Notice that there are a total of six factorial, that is 720, permutations for the maze times. Since facorials grow very quickly in size, it may not be feasible to check the mean difference in times for all possible permutations. However, we can repeat this a large number of times and the results should be highly representative of what we should see if we were to consider all possible permutations provided that the process is repeated sufficicently many times. Let's do this now and collect the results. It's useful to write another function that incorporates the permuting:

```
mean_diff_perm <- function(){
  time_vals <- c(30,25,20,18,21,22)
  mouse_data_perm <- data.frame(times=sample(time_vals,replace = F),
                                group=c(rep("Drug",3),rep("Control",3)),
                                row.names = c("Mouse1","Mouse2","Mouse3","Mouse4","Mouse5","Mouse6"))
  data_summary <- mouse_data_perm %>%
  group_by(group) %>%
  summarise(mean_time=mean(times))
  mean_diff_val <- data_summary$mean_time %>% unlist() %>% diff()
  return(mean_diff_val)
}
```

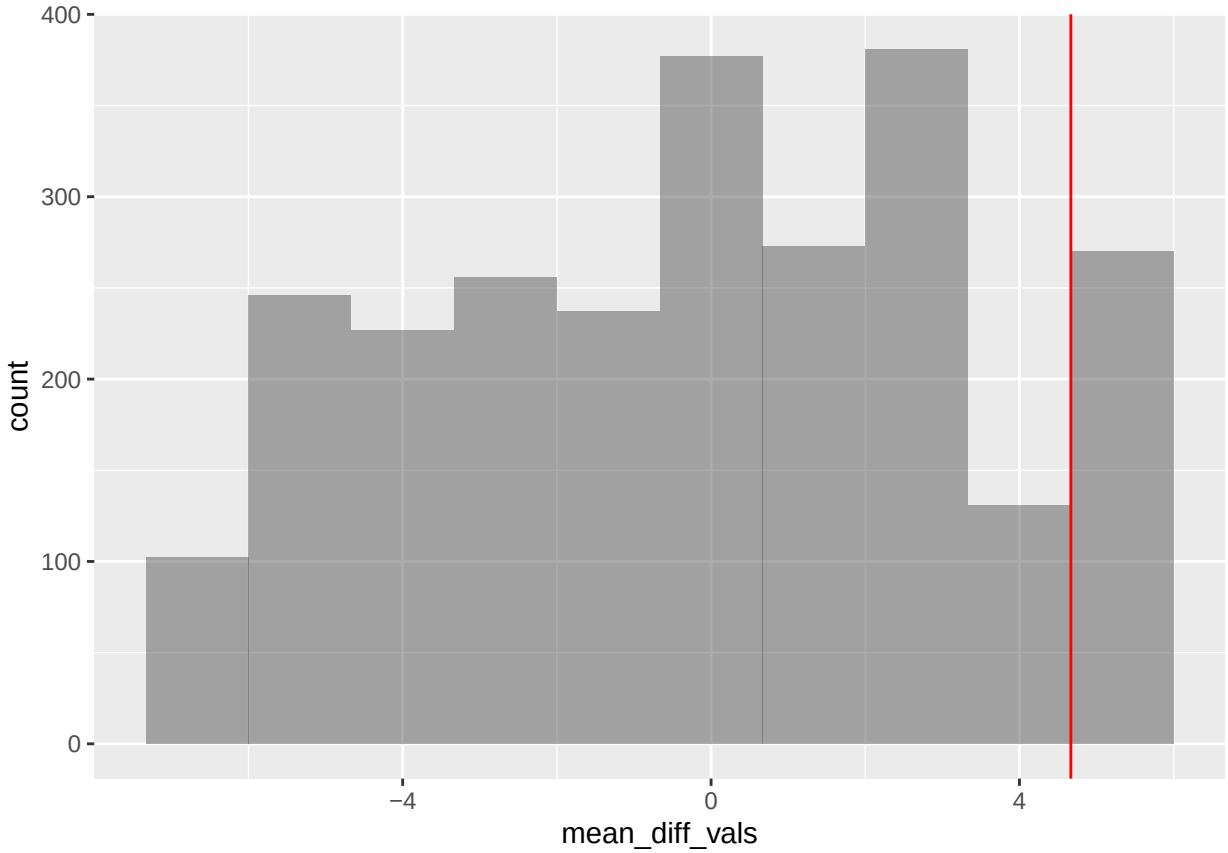
Now we can use the do() function to run this function some very large number of times and collect the results:

```
N <- 2500 # a large number of times
results <- do(N)*c(mean_diff_vals = mean_diff_perm())
```

This may take a little while to run but once it is done, we can plot a histogram of all the mean difference in times obtained for each of the large number of times that we permute the data, together with the observed mean difference in times:

```
results %>% gf_histogram(~mean_diff_vals,bins = 10) %>%
  gf_vline(xintercept = mean_diff_obs,color="red")
```

```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
```



The red vertical line indicates the observed mean difference in times 4.6666667.

Question: Is it typical to get a mean difference in times greater than or equal to 4.6666667?

We can easily compute the proportion of times that the permutated data results in a value that is greater than or equal to the observed value 4.6666667:

```
(sum(results$mean_diff_vals >= mean_diff_obs) + 1)/(N + 1)
```

```
## [1] 0.1607357
```

About 15% of the time, we observe a mean difference in times that is at least as large as 4.6666667. Thus, we conclude that getting a value as large as the one that we observed in the original data is not extremely rare. Alas we can not reasonably rule out the possibility that the observed difference in maze times is due simply to natural random variation. In the terminology that is soon to be introduced, we would fail to reject the null hypothesis.

Since this example is relatively small, it is actually pretty easy to find all possible distinct values for the difference in means, there are only twenty. Out of twenty, three are greater or equal to 4.67. Notice that

```
3/20
```

```
## [1] 0.15
```

is exactly 15%. This illustrates the fact that it is not really necessary to consider all possible permutations, and that if we just look at some very large number of permutations then that should be sufficient.

Hopefully this example gives you some intuitive idea of what is going on in hypothesis testing and some sense of how a permutation test works. We will now introduce some terminology and then follow that up with some more examples of permutation tests.

Hypothesis Testing Terminology

Definition: The *null hypothesis*, denoted H_0, is a statement that corresponds to no real effect. This is the status quo, in the absence of the data providing convincing evidence to the contrary. The *allternative hypothesis*, denoted H_A, is a statment that there is a real effect. The data may provide convincing evidence that this hypothesis is true.

A hypothesis should involve a statement about a population parameter or parameters, commonly referred to as θ; the null hypothesis is $H_0 : \theta = \theta_0$ for some fixed value θ_0. A **one-sided** alternative hypothesis is of the form $H_A : \theta > \theta_0$ or $H_A : \theta < \theta_0$; a **two-sided** alternative hypothesis is $H_A : \theta \neq \theta_0$.

Note: In practice one either rejects the null hypothesis or fails to reject the null hypothesis.

Example: Consider the mice example. Let μ_d denote the true mean time that a randomly selected mouse that received the drug takes to run through the maze; let μ_c denote the true mean time for a control mouse. Then $H_0 : \mu_d - \mu_c = 0$ and $H_A : \mu_d - \mu_c > 0$. Thus our parameter $\theta = \mu_d - \mu_c$.

Definition: A *test statistic* is a numerical function of the data whose value determines the result of the test. The function itself is generally denoted $T = T(X_1, X_2, \ldots, X_n)$ in a one-sample problem or $T = T(X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_m)$ in a two-sample problem. Notice that a test statistic is a random variable, the distribution of a test statistic is called the sampling distribution of the statistic and the standard deviation of a test statistic is called the standard error of the statistic. A specific test statistic value computed from sample data is called an *observed value*, sometimes denotes by lower-case t.

Definition: The p-value is the probability that chance alone would produce a test statistic as or more extreme than the observed test statistic if the null hypothesis were true.

Example: In the mice example, we let the test statistic be the difference in means, $T = T(X_1, X_2, X_3, Y_1, Y_2, Y_3) = \bar{X} - \bar{Y}$, where

$\bar{X} = \frac{X_1+X_2+X_3}{3}$

and

$\bar{Y} = \frac{Y_1+Y_2+Y_3}{3}$

with observed value $t = \bar{x} - \bar{y} = 4.67$, where

$\bar{x} = \frac{30+25+20}{3} = 25$

and

$\bar{y} = \frac{18+21+22}{3} = 20.33.$

The proportion of times that the permutated data results in a value that is greater than or equal to the observed value 4.67 provides an estimate for a p-value since this is approximately the probability that, under the null hypothesis, we obtain a result that is as or more extreme than the observed value.

Definition: The *null distribution* is the distribution of the test statistic if the null hypothesis is true.

Another Example

Consider the Beer and Hot Wings data. This data is in the resampledata package and stored as Beerwings. The data comes from an experiment in which a student recorded the consumption of hot wings by other students. We can examine the first few rows of this data:

```
head(Beerwings)
```

```
##   ID Hotwings Beer Gender
## 1  1        4   24      F
## 2  2        5    0      F
## 3  3        5   12      F
```

```
## 4  4        6   12      F
## 5  5        7   12      F
## 6  6        7   12      F
```

Question: What is contained in this data?

Since the students observed for this data are identified as either female (F) or male (M), we should determine how many are labeled as (F) and how many are labeled as (M).

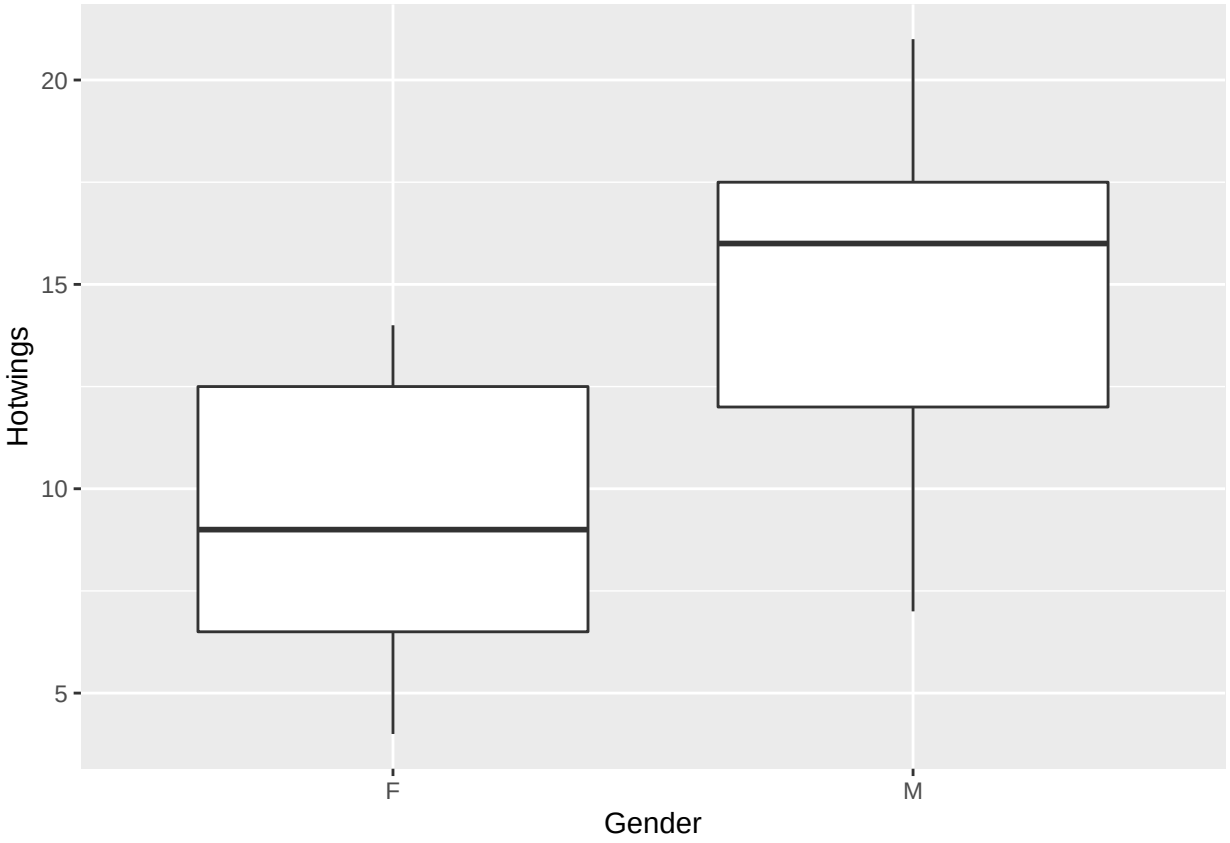
```
table(Beerwings$Gender)
```

```
##
##  F  M
## 15 15
```

We see there are 15 (F) and 15 (M).

Let's examine a box plot that shows the number of hot wings per Gender in the data:

```
Beerwings %>% gf_boxplot(Hotwings ~ Gender)
```



It appears based on this boxplot that on average, individuals recorded as female (F) consume fewer hotwings than those individuals recorded as male (M).

Question: Is this observed difference likely due to random chance or might there be a real effect? We can employ a permutation test once again in an attempt answer this question.

We need a test statistic and we can again take a difference in means. Let's compute the observed value for this test statistic:

First, compute the mean number of hotwings per Gender in the observed data:

```
Beerwings %>%
  group_by(Gender) %>%
  summarise(mean=mean(Hotwings))
```

```
## # A tibble: 2 x 2
##   Gender  mean
##   <fct>  <dbl>
## 1 F       9.33
## 2 M      14.5
```

Now compute the observed difference in means:

```
(obs_diff <- 14.53 - 9.33)
```

```
## [1] 5.2
```

Let μ_M be the mean number of hotwings for a male student and μ_F be the mean number of hotwings for a female student. Then $\mu_M - \mu_F$ is our test statistic and we seek to test the hypothesis:

Null Hypothesis - $H_0 : \mu_M - \mu_F = 0$

versus

Alternative Hypothesis - $H_A : \mu_M - \mu_F > 0$

So this is a one-sided test.

Under the null hypothesis, we should expect that the Gender label has no real significance for the number of hotwings an individual eats. Thus, we can treat Hotwings as a single list of 30 observations, permute it and divide it in half (15 and 15), then compute the mean value for each half, and finally compute the difference in means. If we repeat this process a large number of times the proportion of values obtained that are as or more extreme than our observed value of 5.2 will be a p-value (at least appoximately).

Let's carry this out in R code:

First we write a function that will input the Hotwings variable, permute it, split it in two equal parts, then compute the difference in means for the two parts.

```
diff_mean_perm <- function(wings_data){
  perm_data <- sample(wings_data,replace = F)
  F_wings <- perm_data[1:15]
  M_wings <- perm_data[16:30]
  mu_M <- mean(M_wings)
  mu_F <- mean(F_wings)
  return(mu_M - mu_F)
}
```

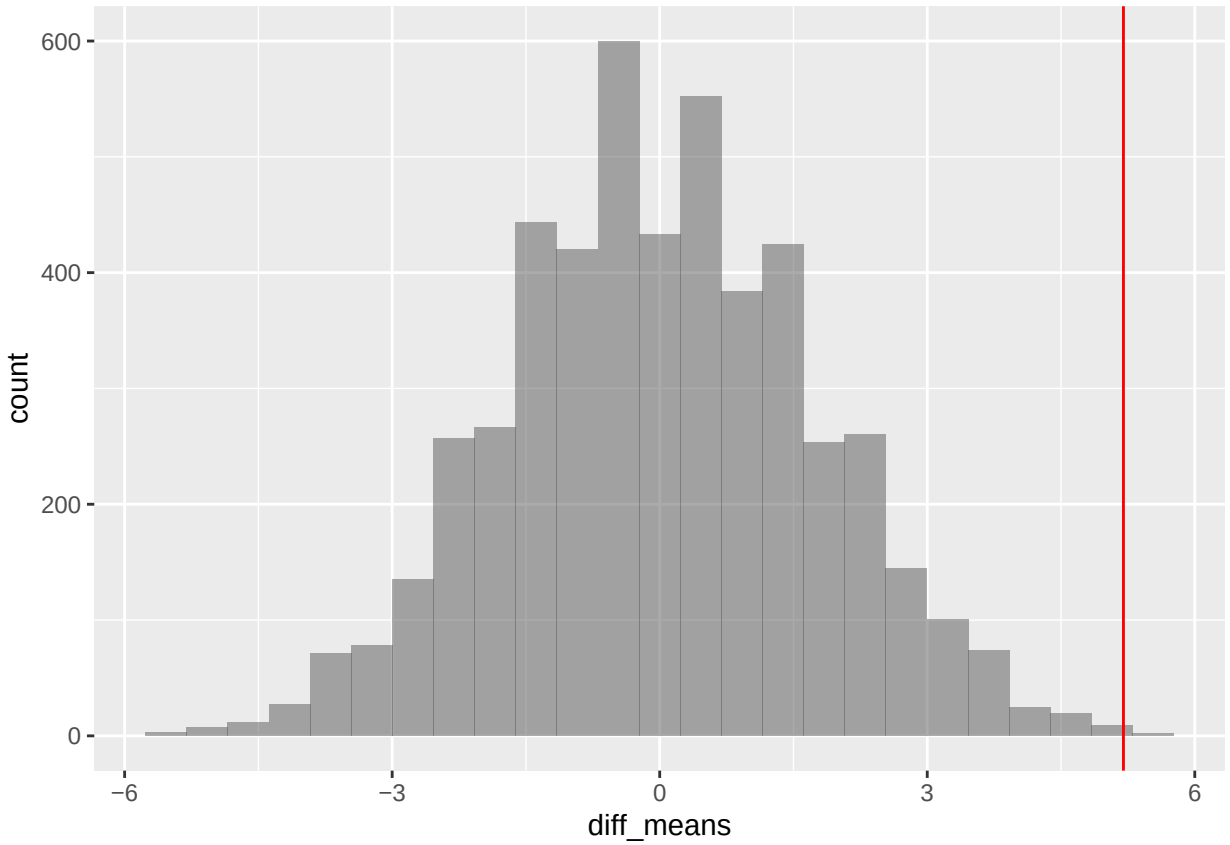
Now we will repeatedly compute the difference in means for the permuted data some large number of times:

```
N <- 5000
wings_results <- do(N) * c(diff_means = diff_mean_perm(Beerwings$Hotwings))
```

Let's look at a histogram of our results:

```
wings_results %>% gf_histogram(~diff_means) %>% gf_vline(xintercept = obs_diff,color="red")
```

```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
```



The red vertical line indicates our observed test statistic value. We can compute the proportion of times that the mean difference in the permuted data is greater or equal to the observed value 5.2. This will give us a p-value (approximately):

```
(sum(wings_results$diff_means >= obs_diff) + 1)/(N + 1)
```

```
## [1] 0.00139972
```

So under the null hypothesis, only about 0.01% of the time would we expect to see an observed difference of 5.2 or greater for the test statistic $\mu_M - \mu_F$. This is quite rare, so we should feel confident that our data provides sufficiently strong evidence to reject the null hypothesis that the observed difference in the number of hotwings individuals labeled as (F) consume compared with the number of hotwings individuals labeled as (M) is unlikely due to pure random chance.

In the next notebook, we will consider additional topics in hypothesis testing and learn a little more about permutation tests.