

# Permutation Tests

*Canhong Wen*

## Introduction

- In statistical inference, it is important to know the distribution of some statistics under null hypothesis ( $H_0$ ), so that quantities like p-values can be derived.
- The null distribution is available theoretically in some cases.
  - For example, assume  $X_i \sim \mathcal{N}(\mu, \sigma^2)$ ,  $i = 1, \dots, n$ . Under  $H_0 : \mu = 0$ , we have  $\bar{X} \sim \mathcal{N}(0, \sigma^2/n)$ . Then  $H_0$  can be tested by comparing  $\bar{X}$  with  $\mathcal{N}(0, \sigma^2/n)$ .
- When null distribution cannot be obtained, it is useful to use **permutation** test to “create” a null distribution from data.

## Basic idea of permutation test

- Permute data under  $H_0$  for a number of times. Each time recompute the test statistics. The test statistics obtained from the permuted data form the null distribution.
- Compare the observed test statistics with the null distribution to obtain statistical significance, i.e., p-value.

## A Simple Example

```
set.seed(1)
x <- rnorm(100, 0, 1)
y <- rnorm(100, 0.5, 1)
t.test(x,y)

##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -2.6906, df = 197.19, p-value = 0.007745
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.61226146 -0.09434766
## sample estimates:
## mean of x mean of y
## 0.1088874 0.4621919

nsims <- 5000
t.obs <- t.test(x, y)$statistic
t.perm <- numeric(nsims)
for(i in 1:nsims){
  tmp <- sample(c(x,y))
  t.perm[i] <- t.test(tmp[1:100], tmp[110:200])$statistic
}
mean(abs(t.obs) < abs(t.perm))
```

```
## [1] 0.0086
```

## A Simple Example

```
set.seed(1)
x <- rt(100, 2)
y <- rnorm(100, 0.35, 1)
t.test(x,y)

##
## Welch Two Sample t-test
##
## data: x and y
## t = -1.8376, df = 105.17, p-value = 0.06894
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -2.16545118 0.08228894
## sample estimates:
## mean of x mean of y
## -0.5754399 0.4661412
```

```
nsims <- 5000
t.obs <- t.test(x, y)$statistic
t.perm <- numeric(nsims)
for(i in 1:nsims){
  tmp <- sample(c(x,y))
  t.perm[i] <- t.test(tmp[1:100], tmp[110:200])$statistic
}
mean(abs(t.obs) < abs(t.perm))
```

```
## [1] 0.0428
```

## Comparison

Comparing a **standard**  $t$ -test approach to a permutation approach brings out some general points about permutation tests versus traditional formula-based tests

- The hypotheses for a  $t$  test are stated in terms of population means

$$H_0 : \mu_X - \mu_Y = 0$$

- Permutation test hypotheses are more general, i.e. the null hypothesis is **same distribution in both groups**.
- The  $t$  test gives accurate p-values when the sampling distribution of the difference of means is at least roughly normal. The permutation test gives accurate p-values even when the sampling distribution is not close to normal.

## Permutation Test Procedure

1. Compute the observed test statistic  $\hat{\theta}(X, Y) = \hat{\theta}(Z, \nu)$ .

2. For each replicate, indexed  $b = 1, \dots, B$ :
  - (a) Generate a random permutation  $\pi_b = \pi(\nu)$ .
  - (b) Compute the statistic  $\hat{\theta}^{(b)} = \hat{\theta}^*(Z, \pi_b)$ .
3. If large values of  $\hat{\theta}$  support the alternative, compute the empirical p-value by

$$\hat{p} = \frac{1 + \sum_{b=1}^B I(\hat{\theta}^{(b)} \geq \hat{\theta})}{B + 1}$$

For a lower-tail or two-tail test  $\hat{p}$  is computed in a similar way.

4. Reject  $H_0$  at significance level  $\alpha$  of  $\hat{p} \leq \alpha$ .

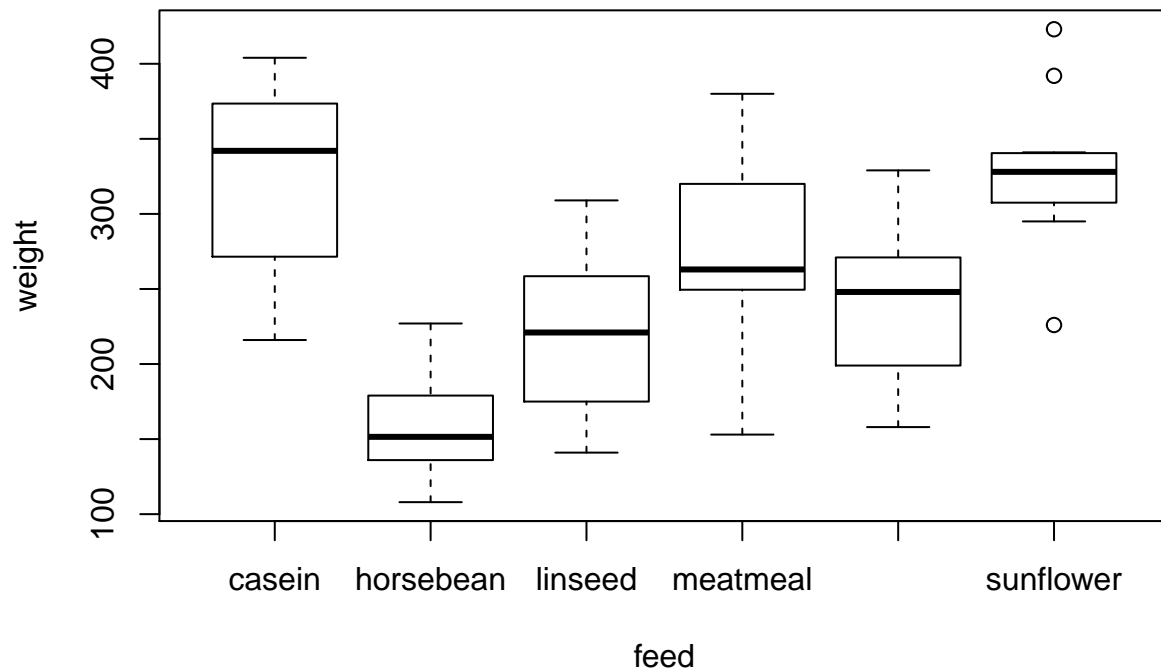
### Example: chickwts

- The permutation distribution of a statistic is illustrated for a small sample, from the data set `chickwts` in R.
- Weights in grams are recorded for six groups of newly hatched chicks fed different supplements.

```
data(chickwts)
attach(chickwts)
```

### Example: chickwts

```
boxplot(formula(chickwts))
```



## Example: chickwts

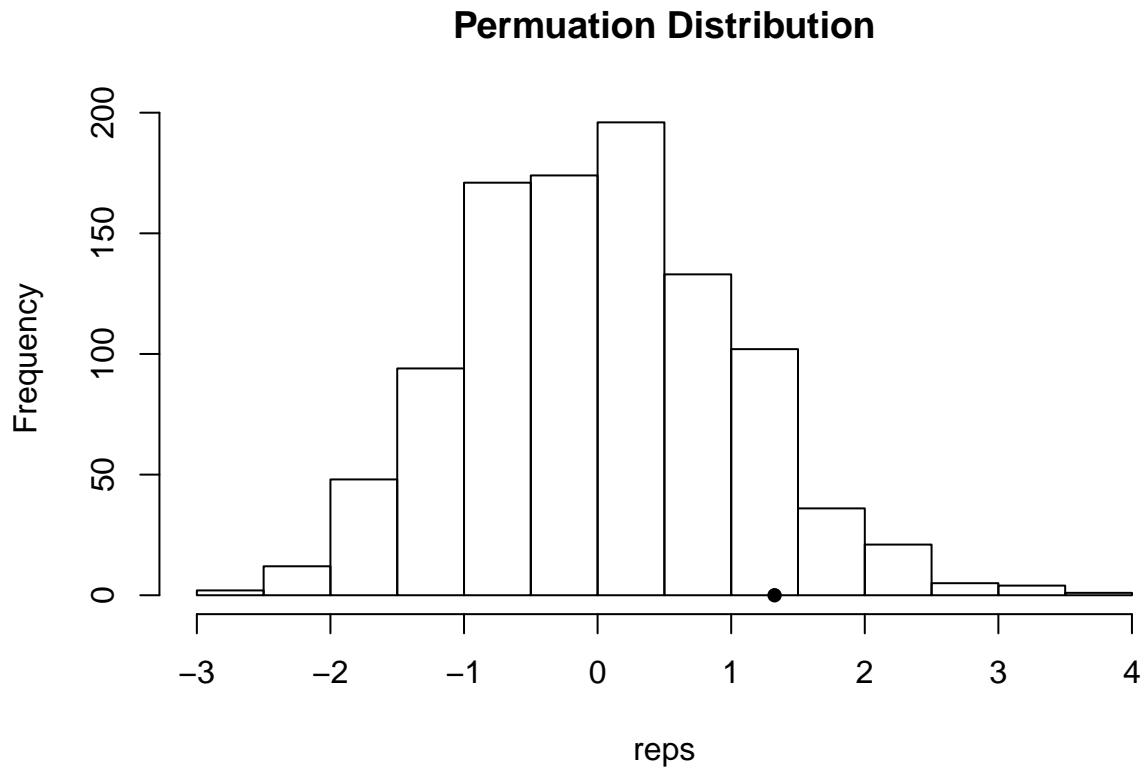
```
detach(chickwts)
X <- as.vector(chickwts$weight[chickwts$feed=="soybean"])
Y <- as.vector(chickwts$weight[chickwts$feed=="linseed"])

B <- 999
Z <- c(X,Y)
reps <- numeric(B)
K <- 1:26
t0 <- t.test(X,Y)$statistic
for(i in 1:B){
  k <- sample(K, size=14, replace=F)
  x1 <- Z[k]
  y1 <- Z[-k]
  reps[i] <- t.test(x1,y1)$statistic
}
p <- mean(c(t0, reps)>=t0)
p
```

```
## [1] 0.089
```

## Example: chickwts

```
hist(reps, main="Permuation Distribution")
points(t0,0, cex=1, pch=16)
```



## Tests for Equal Distributions

- To apply a permutation test of equal distributions, choose a test statistic that measures the difference between two distributions, for example, the two-sample Kolmogorov-Smirnov (K-S) statistic.
- The K-S statistic  $D$  is the maximum absolute difference between the ecdfs of the two samples, defined by

$$D = \sup_{1 \leq i \leq m+n} |F_n(z_i) - G_m(z_i)|$$

where  $F_n$  is the ecdf of the first sample  $X_1, \dots, X_n$  and  $G_m$  is the ecdf of the second sample  $Y_1, \dots, Y_m$ . Note that  $0 \leq D \leq 1$  and large values of  $D$  support the alternative  $H_1 : F_X \neq F_Y$ .

- In R, we can compute this statistic using `ks.test`.

## Example: K-S statistic

```
D <- numeric(B)
D0 <- ks.test(X,Y, exact=F)$statistic

## Warning in ks.test(X, Y, exact = F): p-value will be approximate in the
## presence of ties

options(warn=-1)
D <- numeric(B)
for( i in 1:B){
  k <- sample(K, size=14, replace=F)
```

```

x1 <- Z[k]
y1 <- Z[-k]
D[i] <- ks.test(x1, y1, exact=F)$statistic
}
p <- mean(c(D0,D) >= D0)
p

## [1] 0.474

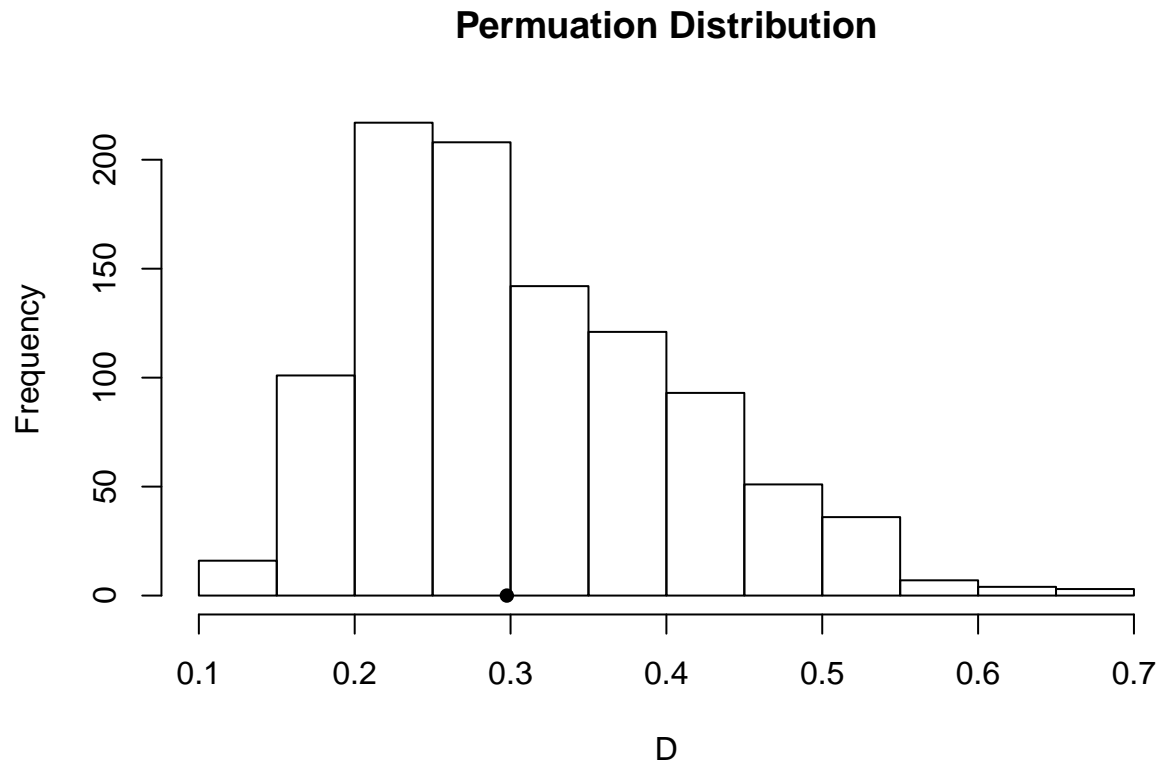
```

### Example: K-S statistic

```

hist(D, main="Permuation Distribution")
points(D0,0, cex=1, pch=16)

```



### Example: Correlation coefficients

- A study by Katz et al. (1990) asked students to answer SAT-type questions without having read the passage on which those questions were based (The SAT is a standardized exam commonly used in university admissions)
- Authors looked to see how performance on such items correlated with the SAT scores those students had when they applied to college
- Expected that those students who had the skill to isolate and reject unreasonable answers, even when they couldn't know the correct answer, would also be students who would have done well on the SAT taken sometime before they came to college

## Example: Correlation coefficients

```
Score <- c(58, 48, 48, 41, 34, 43, 38, 53, 41, 60, 55, 44,
          43, 49, 47, 33, 47, 40, 46, 53, 40, 45, 39, 47,
          50, 53, 46, 53)
SAT <- c(590, 590, 580, 490, 550, 580, 550, 700, 560, 690, 800, 600,
        650, 580, 660, 590, 600, 540, 610, 580, 620, 600, 560, 560,
        570, 630, 510, 620)
r.obt <- cor(Score, SAT)
cat("The obtained correlation is ",r.obt,'\n')

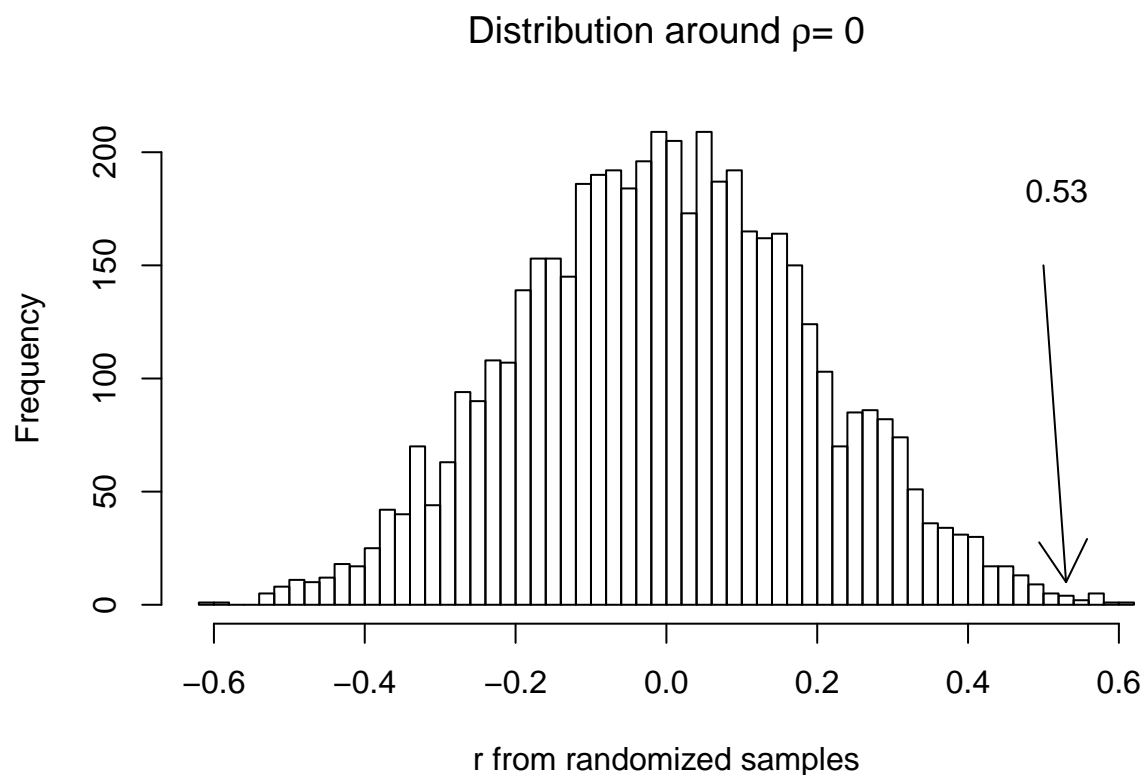
## The obtained correlation is 0.531767
```

## Example: Correlation coefficients

```
nreps <- 5000
r.random <- numeric(nreps)
for (i in 1:nreps) {
  Y <- Score
  X <- sample(SAT, 28, replace = FALSE)
  r.random[i] <- cor(X,Y)
}
prob <- length(r.random[r.random >= r.obt])/nreps
cat("Probability randomized r >= r.obt",prob)

## Probability randomized r >= r.obt 0.0018
```

## Example: Correlation coefficients



## Permutation with package coin

```
library(coin)
```

```
## Loading required package: survival
```

```
oneway_test(weight~feed, data = chickwts[chickwts$feed%in%c("linseed", "soybean"),])
```

```
##
```

```
## Asymptotic Two-Sample Fisher-Pitman Permutation Test
```

```
##
```

```
## data: weight by feed (linseed, soybean)
```

```
## Z = -1.3015, p-value = 0.1931
```

```
## alternative hypothesis: true mu is not equal to 0
```

```
library(coin)
```

```
oneway_test(weight~feed, data = chickwts)
```

```
##
```

```
## Asymptotic K-Sample Fisher-Pitman Permutation Test
```

```
##
```



```
## data: weight by
## feed (casein, horsebean, linseed, meatmeal, soybean, sunflower)
## chi-squared = 37.918, df = 5, p-value = 3.919e-07
```

## Bootstrap versus permutation

- When we bootstrap for correlations, we keep  $x_i$  and  $y_i$  pairs together, and randomly sample pairs of scores with replacement. That means that if one pair is 45 and 360, we will always have 45 and 360 occur together, often more than once, or neither of them will occur. What this means is that the expectation of the correlation between  $X$  and  $Y$  for any resampling will be the correlation in the original data.

- When we use a permutation approach, we permute the  $Y$  values, while holding the  $X$  values constant. For example, if the original data were

```
x <- c(45, 53, 73, 80)
y <- c(22, 30, 29, 38)
```

then two resamples might be

```
rbind(x, sample(y, size=4, replace=F))
```

```
##    [,1] [,2] [,3] [,4]
## x   45   53   73   80
##    29   38   22   30
```

```
rbind(x, sample(y, size=4, replace=F))
```

```
##    [,1] [,2] [,3] [,4]
## x   45   53   73   80
##    22   38   30   29
```

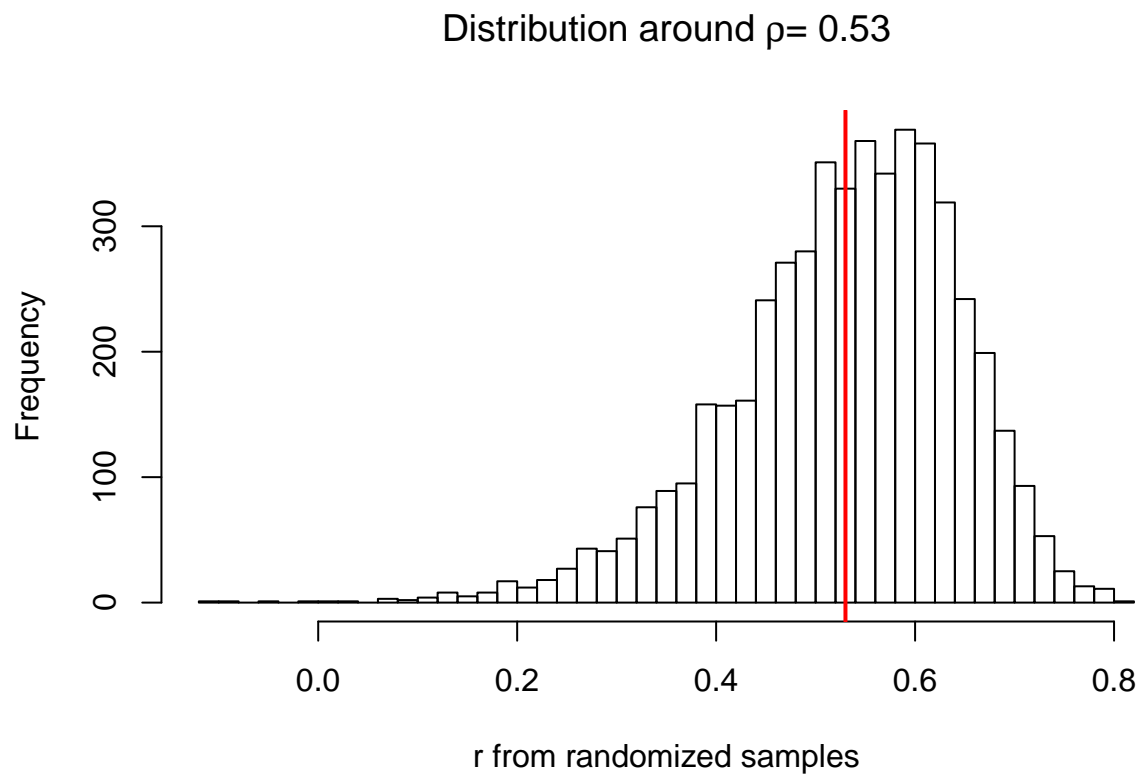
- Notice the top row always stays in the same order, while the bottom row is permuted randomly. This means that the expected value of the correlation between  $X$  and  $Y$  will be 0.00, not the correlation in the original sample.
- Helps to explain why bootstrapping focuses on confidence limits around  $\rho$ , whereas the permutation procedure focuses on confidence limits around 0.

```
nreps <- 5000
r.random <- numeric(nreps)
for (i in 1:nreps) {
  sam <- sample(length(Score), replace = TRUE)
  Y <- Score[sam]
  X <- SAT[sam]
  r.random[i] <- cor(X,Y)
}
```

```
prob <- length(r.random[r.random >= r.obt])/nreps
cat("Probability randomized r >= r.obt",prob)
```

```
## Probability randomized r >= r.obt 0.5454
```

```
hist(r.random, breaks = 50, main = expression(paste("Distribution around ",rho, "= 0.53")), xlab = "r")
r.obt <- round(r.obt, digits = 2)
abline(v = r.obt, col="red", lwd =2)
```



## Summary

- Permutation procedure
  - Test for equality of mean
  - Test for equality of distribution
  - Test for independence