732A90 Computational Statistics - Lab 5

Joris van Doorn - jorva845

28 February 2020

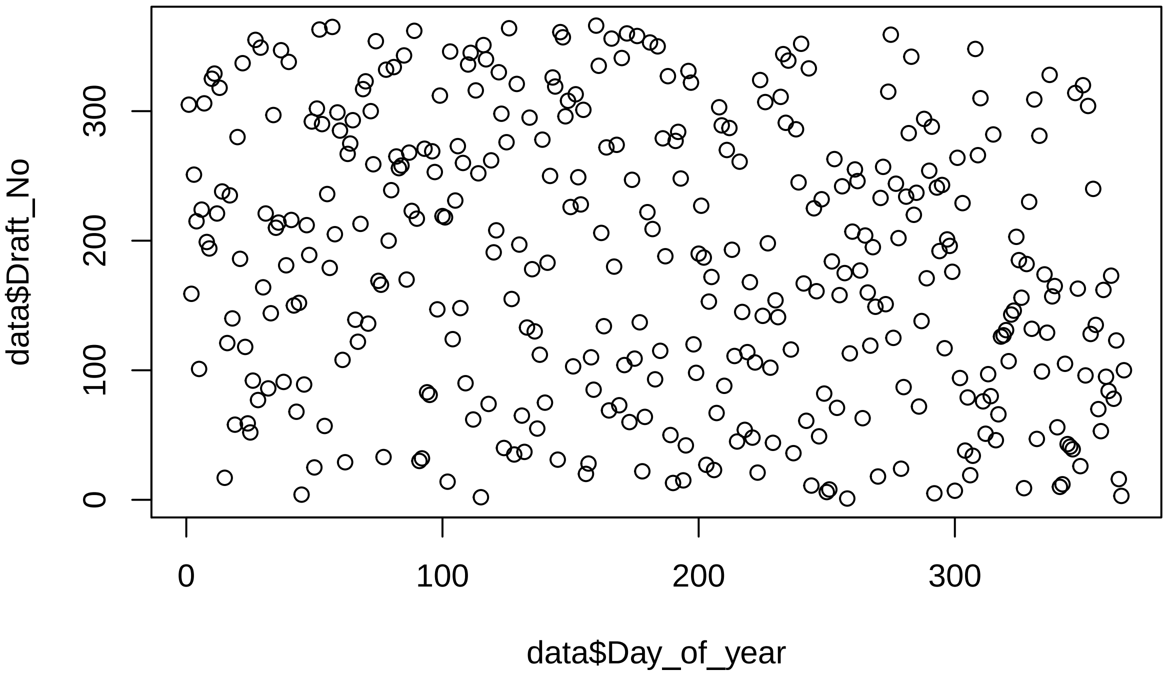
Q1: Hypothesis testing

In 1970, the US Congress instituted a random selection process for the military draft. All 366 possible birth dates were placed in plastic capsules in a rotating drum and were selected one by one. The first date drawn from the drum received draft number one, the second date drawn received draft number two, etc. Then, eligible men were drafted in the order given by the draft number of their birth date. In a truly random lottery there should be no relationship between the date and the draft number. Your task is to investigate whether or not the draft numbers were randomly selected. The draft numbers (Y=Draft No) sorted by day of year (X=Day of year) are given in the file lottery.xls.

1.

Make a scatterplot of Y versus X and conclude whether the lottery looks random.

```
data <- read.csv2("lottery.csv")
plot(data$Day_of_year, data$Draft_No)
```

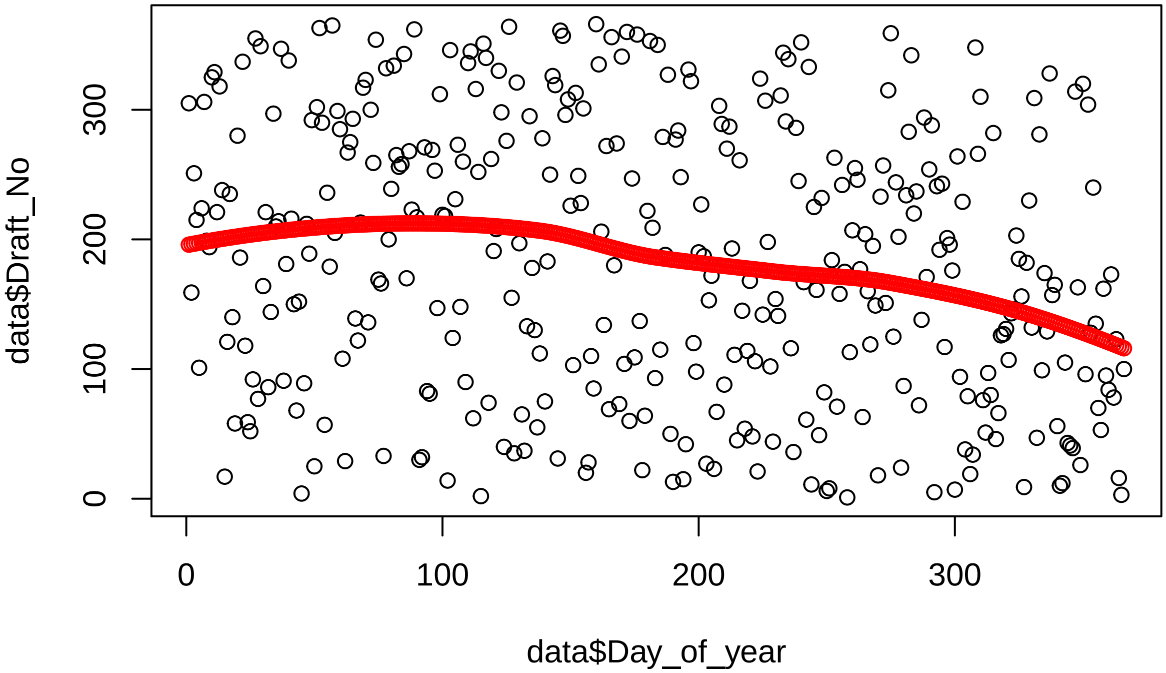


On first glance, it looks pretty random to me. The datapoints are distributed over the whole plot. No clear patterns or clusters to be detected. No relationship to be found by eye.

2.

Compute an estimate $\hat{Y}$ of the expected response as a function of X by using a loess smoother (use loess()), put the curve $\hat{Y}$ versus X in the previous graph and state again whether the lottery looks random.

```
loess1 <- loess(data$Draft_No ~ data$Day_of_year, data = data)
Y_hat <- predict(loess1, data = data)
plot(data$Day_of_year, data$Draft_No)
points(Y_hat, col = "red")
```



The red line represents the prediction of the loess model. As it becomes clear from the model, there is a clear relation ship. If you are born early in the year, you are more likely to be drawn then if you are born later in the year. There seems to be a peak around the start of april, so if you are born in april you are almost twice as likely to be sent out then if you are born in november or december.

3.

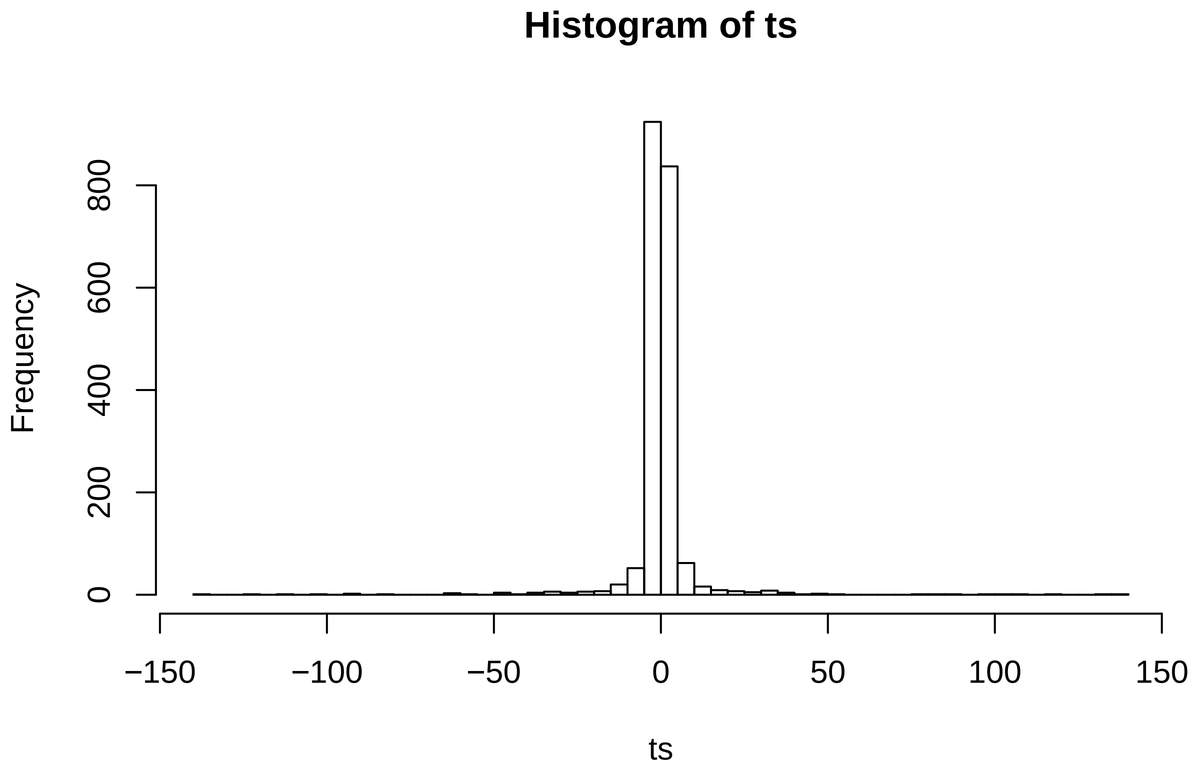
To check whether the lottery is random, it is reasonable to use test statistics

$$T = \frac{\hat{Y}(X_b) - \hat{Y}(X_a)}{X_b - X_a}, \, where \, X_b = argmax_X \hat{Y}(X), \, X_a = argmin_X \hat{Y}(X)$$

If this value is significantly greater than zero, then there should be a trend in the data and the lottery is not random. Estimate the distribution of T by using a non-parametric bootstrap with B = 2000 and comment whether the lottery is random or not. What is the p-value of the test?

```
Y_hat_max <- max(Y_hat)
Y_hat_min <- min(Y_hat)
```

```r
Xb <- which(Y_hat_max == Y_hat)
Xa <- which(Y_hat_min == Y_hat)
T_stat <- (Y_hat_max-Y_hat_min)/(Xb-Xa)

B <- 2000
ts <- c()

set.seed(12345)

for(i in 1:B){
  #Y_hat <- predict(loess1, data = new_sample)
  n <- dim(data)[1]
  id <- sample(1:n, n, replace = T)
  new_data <-data[id,]

  loess1 <- loess(new_data$Draft_No ~ new_data$Day_of_year, data = new_data)
  Y_hat <- predict(loess1, data = new_data)

  Y_hat_max <- max(Y_hat)
  Y_hat_min <- min(Y_hat)
  Xb <- which.max(Y_hat)
  Xa <- which.min(Y_hat)
  ts[i] <- (Y_hat_max-Y_hat_min)/(Xb-Xa)
}

hist(ts, breaks = 50)
```



```
stat0 <- 0

print(c(stat0,mean(ts>stat0)))
```

```
## [1] 0.0000 0.4805
```

4.

Implement a function depending on data and B that tests the hypothesis H0: Lottery is random versus H1: Lottery is non-random, by using a permutation test with statistics T. The function is to return the p-value of this test. Test this function on our data with B = 2000.

```
Y <- data$Draft_No
X <- data$Day_of_year
data_transformed <- data.frame(Y, X)

permu_test <- function(data, B){
  ts <- c()

  for(i in 1:B){
    #Y_hat <- predict(loess1, data = new_sample)
    n <- dim(data)[1]
    id <- sample(1:n, n, replace = T)
    new_data <-data[id,]

    loess1 <- loess(new_data$Y ~ new_data$X, data = new_data)
    Y_hat <- predict(loess1, data = new_data)
    Y_hat_max <- max(Y_hat)
    Y_hat_min <- min(Y_hat)
    Xb <- which.max(Y_hat)
    Xa <- which.min(Y_hat)
    ts[i] <- (Y_hat_max-Y_hat_min)/(Xb-Xa)
  }

  stat0 <- 0

  print(c(stat0,mean(ts>stat0)))
}

permu_test(data_transformed, 2000)
```

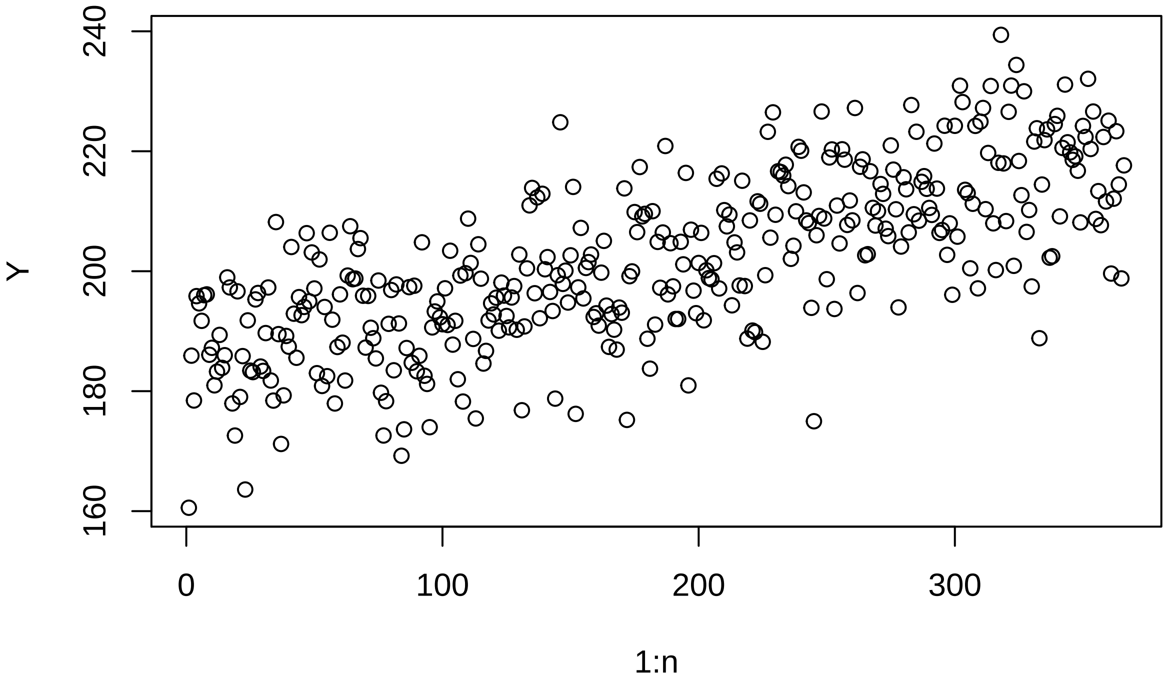
```
## [1] 0.000 0.487
```

5.

a.

Generate (an obviously non-random) dataset with n = 366 observations by using same X as in the original data set and Y(x) = max(0, min(ax + b, 366)), where a = 0.1 and b ∼ N(183, sd = 10).

```
n <- 366
Y <- c()

for(i in 1:n){
  Y[i] <- max(0,min(0.1*i + rnorm(1, mean = 183, sd = 10)))
}
plot(1:n,Y)
```



```
gen_data <- data.frame(Y,1:n)
```

b.

Plug these data into the permutation test with B = 200 and note whether it was rejected.

```
permu_test(gen_data, B = 200)
```

```
## [1] 0.000 0.495
```

c.

Repeat Steps 5a{5b for a = 0.2, 0.3, . . . , 10.

```
n <- 366
Y <- c()
alpha <- seq(0.2, 10, by = 0.1)

for(j in 1:length(alpha)){
  for(i in 1:n){
    Y[i] <- max(0,min(alpha[j]*i + rnorm(1, mean = 183, sd = 10)))
  }

  gen_data <- data.frame(Y,1:n)
  print(alpha[j])
  permu_test(gen_data, B = 200)
  Y <- c()
}
```

```
## [1] 0.2
## [1] 0.000 0.495
## [1] 0.3
## [1] 0.00 0.51
## [1] 0.4
## [1] 0.00 0.47
## [1] 0.5
## [1] 0.000 0.425
## [1] 0.6
## [1] 0.000 0.565
## [1] 0.7
## [1] 0.0 0.5
## [1] 0.8
## [1] 0.000 0.515
## [1] 0.9
## [1] 0.0 0.5
## [1] 1
## [1] 0.00 0.54
## [1] 1.1
## [1] 0.000 0.535
## [1] 1.2
## [1] 0.00 0.47
## [1] 1.3
## [1] 0.00 0.51
## [1] 1.4
## [1] 0.000 0.575
## [1] 1.5
## [1] 0.000 0.485
## [1] 1.6
## [1] 0.000 0.515
## [1] 1.7
## [1] 0.000 0.495
## [1] 1.8
## [1] 0.00 0.44
## [1] 1.9
## [1] 0.000 0.525
## [1] 2
## [1] 0.00 0.56
## [1] 2.1
## [1] 0.000 0.435
## [1] 2.2
## [1] 0.000 0.465
## [1] 2.3
## [1] 0.00 0.48
## [1] 2.4
## [1] 0.000 0.525
## [1] 2.5
## [1] 0.000 0.485
## [1] 2.6
## [1] 0.000 0.495
## [1] 2.7
## [1] 0.000 0.525
## [1] 2.8
## [1] 0.00 0.51
## [1] 2.9
## [1] 0.00 0.49
## [1] 3
## [1] 0.00 0.52
## [1] 3.1
## [1] 0.00 0.54
## [1] 3.2
## [1] 0.00 0.44
## [1] 3.3
## [1] 0.000 0.525
## [1] 3.4
## [1] 0.000 0.515
## [1] 3.5
## [1] 0.00 0.53
## [1] 3.6
## [1] 0.00 0.46
## [1] 3.7
## [1] 0.000 0.505
## [1] 3.8
## [1] 0.000 0.485
## [1] 3.9
## [1] 0.00 0.53
## [1] 4
## [1] 0.00 0.52
## [1] 4.1
## [1] 0.0 0.5
## [1] 4.2
## [1] 0.00 0.48
## [1] 4.3
## [1] 0.00 0.56
## [1] 4.4
## [1] 0.000 0.565
## [1] 4.5
## [1] 0.0 0.6
## [1] 4.6
## [1] 0.000 0.495
## [1] 4.7
## [1] 0.00 0.46
## [1] 4.8
## [1] 0.000 0.525
## [1] 4.9
## [1] 0.00 0.49
## [1] 5
## [1] 0.00 0.45
## [1] 5.1
## [1] 0.000 0.535
## [1] 5.2
## [1] 0.000 0.465
## [1] 5.3
## [1] 0.00 0.46
## [1] 5.4
## [1] 0.00 0.57
## [1] 5.5
## [1] 0.00 0.47
## [1] 5.6
## [1] 0.00 0.47
## [1] 5.7
## [1] 0.00 0.49
## [1] 5.8
## [1] 0.000 0.445
## [1] 5.9
## [1] 0.00 0.55
## [1] 6
## [1] 0.00 0.55
## [1] 6.1
## [1] 0.00 0.48
## [1] 6.2
## [1] 0.00 0.48
## [1] 6.3
## [1] 0.000 0.505
## [1] 6.4
## [1] 0.00 0.48
## [1] 6.5
## [1] 0.000 0.535
## [1] 6.6
## [1] 0.00 0.45
## [1] 6.7
## [1] 0.000 0.505
## [1] 6.8
## [1] 0.000 0.495
## [1] 6.9
## [1] 0.00 0.51
## [1] 7
## [1] 0.000 0.505
## [1] 7.1
## [1] 0.000 0.525
## [1] 7.2
## [1] 0.000 0.535
## [1] 7.3
## [1] 0.000 0.495
## [1] 7.4
## [1] 0.00 0.49
## [1] 7.5
## [1] 0.00 0.47
## [1] 7.6
## [1] 0.00 0.45
## [1] 7.7
## [1] 0.000 0.475
## [1] 7.8
## [1] 0.00 0.42
## [1] 7.9
## [1] 0.00 0.55
## [1] 8
## [1] 0.00 0.51
## [1] 8.1
## [1] 0.00 0.49
## [1] 8.2
## [1] 0.00 0.47
## [1] 8.3
## [1] 0.00 0.47
## [1] 8.4
## [1] 0.000 0.475
## [1] 8.5
## [1] 0.00 0.46
## [1] 8.6
## [1] 0.00 0.46
## [1] 8.7
## [1] 0.000 0.525
## [1] 8.8
## [1] 0.000 0.505
## [1] 8.9
## [1] 0.000 0.515
## [1] 9
## [1] 0.00 0.49
## [1] 9.1
## [1] 0.00 0.49
## [1] 9.2
## [1] 0.00 0.46
## [1] 9.3
## [1] 0.000 0.495
## [1] 9.4
## [1] 0.00 0.52
## [1] 9.5
## [1] 0.00 0.46
## [1] 9.6
## [1] 0.00 0.56
## [1] 9.7
## [1] 0.000 0.485
## [1] 9.8
## [1] 0.00 0.48
## [1] 9.9
## [1] 0.000 0.505
## [1] 10
## [1] 0.00 0.51
```

What can you say about the quality of your test statistics considering the value of the power?

It remains approximately the same.

Q2 - Bootstrap, jackknife and confidence intervals

The data you are going to continue analyzing is the database of home prices in Albuquerque, 1993. The variables present are Price, SqFt, the area of a house, FEATS, number of features such as dishwasher, refrigerator and so on, Taxes, annual taxes paid for the house. Explore the file prices1.xls.

1.

Plot the histogram of Price. Does it remind any conventional distribution? Compute the mean price.

Appendix

```
RNGversion(min(as.character(getRversion()), "3.6.2"))
knitr::opts_chunk$set(echo = TRUE)
library(RMaCzek)
library(knitr)
library(tidyr)
library(tidyverse)
library(tinytex)
library(dplyr)
library(readxl)
library(stats)
library(coda)
library(gdata)
data <- read.csv2("lottery.csv")
plot(data$Day_of_year, data$Draft_No)
loess1 <- loess(data$Draft_No ~ data$Day_of_year, data = data)
Y_hat <- predict(loess1, data = data)
plot(data$Day_of_year, data$Draft_No)
points(Y_hat, col = "red")
Y_hat_max <- max(Y_hat)
Y_hat_min <- min(Y_hat)
Xb <- which(Y_hat_max == Y_hat)
Xa <- which(Y_hat_min == Y_hat)
T_stat <- (Y_hat_max-Y_hat_min)/(Xb-Xa)

B <- 2000
ts <- c()

set.seed(12345)

for(i in 1:B){
  #Y_hat <- predict(loess1, data = new_sample)
  n <- dim(data)[1]
  id <- sample(1:n, n, replace = T)
  new_data <-data[id,]

  loess1 <- loess(new_data$Draft_No ~ new_data$Day_of_year, data = new_data)
  Y_hat <- predict(loess1, data = new_data)

  Y_hat_max <- max(Y_hat)
  Y_hat_min <- min(Y_hat)
  Xb <- which.max(Y_hat)
```

```r
  Xa <- which.min(Y_hat)
  ts[i] <- (Y_hat_max-Y_hat_min)/(Xb-Xa)
}

hist(ts, breaks = 50)

stat0 <- 0

print(c(stat0,mean(ts>stat0)))
Y <- data$Draft_No
X <- data$Day_of_year
data_transformed <- data.frame(Y, X)

permu_test <- function(data, B){
  ts <- c()

  for(i in 1:B){
    #Y_hat <- predict(loess1, data = new_sample)
    n <- dim(data)[1]
    id <- sample(1:n, n, replace = T)
    new_data <-data[id,]

    loess1 <- loess(new_data$Y ~ new_data$X, data = new_data)
    Y_hat <- predict(loess1, data = new_data)

    Y_hat_max <- max(Y_hat)
    Y_hat_min <- min(Y_hat)
    Xb <- which.max(Y_hat)
    Xa <- which.min(Y_hat)
    ts[i] <- (Y_hat_max-Y_hat_min)/(Xb-Xa)
  }

  stat0 <- 0

  print(c(stat0,mean(ts>stat0)))
}

permu_test(data_transformed, 2000)
n <- 366
Y <- c()

for(i in 1:n){
  Y[i] <- max(0,min(0.1*i + rnorm(1, mean = 183, sd = 10)))
}
plot(1:n,Y)

gen_data <- data.frame(Y,1:n)
permu_test(gen_data, B = 200)
n <- 366
Y <- c()
alpha <- seq(0.2, 10, by = 0.1)

for(j in 1:length(alpha)){
  for(i in 1:n){
    Y[i] <- max(0,min(alpha[j]*i + rnorm(1, mean = 183, sd = 10)))
  }

  gen_data <- data.frame(Y,1:n)
  print(alpha[j])
  permu_test(gen_data, B = 200)
  Y <- c()
}
```