Advanced Stats HW 3

Joel Vasama — 6012872

2022-07-22

Task 1 — Permutation testing

Independent testing

t-test

```
tTest1 <- t.test(table1$GroupA, table1$GroupB, "two.sided")
tTest1
```

```
##  Welch Two Sample t-test
##
## data:  table1$GroupA and table1$GroupB
## t = -1.1701, df = 67.971, p-value = 0.246
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -10.683434   2.785546
## sample estimates:
## mean of x mean of y
##  72.41553  76.36447
```

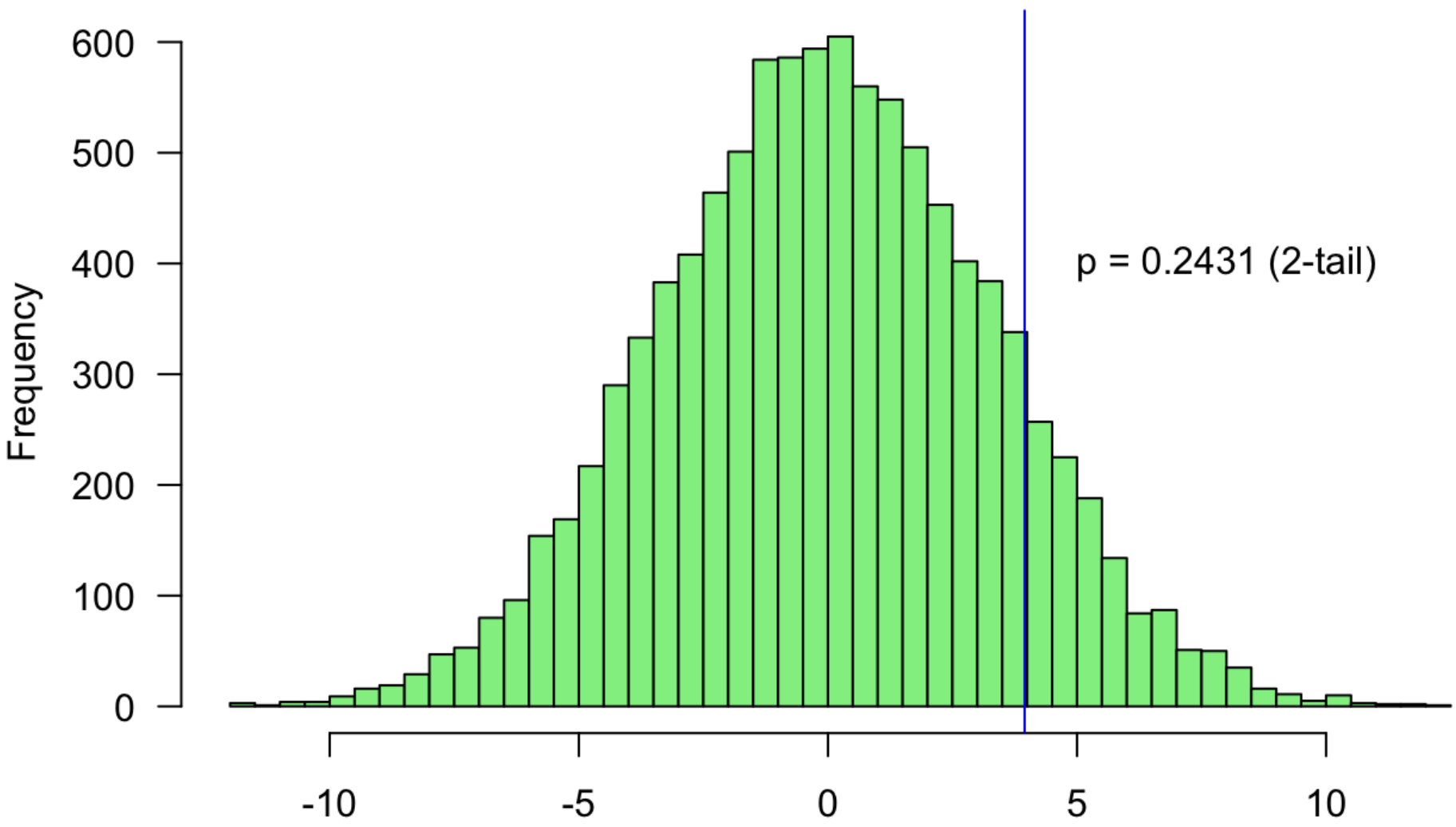
- Non-significant result, finding no evidence for a difference between groups A and B

Permutation Test

```r
# Preparation
stackedTable1 <- stack(table1)
names(stackedTable1) <- c("vals", "group")

vals <- stackedTable1$vals
group <- stackedTable1$group

# Permutation test function
permTest <- function(x, label, n){
  distro <- matrix(NA, n, 1)
  observed <- diff(tapply(x, label, mean))
  for(i in 1:n){
    distro[i] <- diff(by(x, sample(label, length(label), FALSE), mean))
  }
  p <- sum(abs(distro) >= abs(observed))/n
  return(list(p, distro, observed))
}

# Main
indpPerm <- permTest(vals, group, 10000)

# Plot
hist(indpPerm[[2]], breaks = 50, col = 'light green', main = "Permutation Distribu
tion",
     las = 1, xlab = '')
abline(v = indpPerm[[3]], col = "blue")
text(8, 400, paste("p =", indpPerm[[1]], "(2-tail)"))
```

ADD ORIGINAL OBSERVATION TO DISTRO

DO n-1 and add obs at distro(n)



- Similarly to the t-test, after 10,000 permutations a mean difference between groups A and B is found in about 2500 (or 25%) of random samples, providing a independent permutation test p-value of p ≈ 0.25 (two-tailed). This suggests that the mean difference is not significantly different from random variation in the data.

Pairwise testing

t-test

```
tTest2 <- t.test(table1$GroupA, table1$GroupB, paired = TRUE)
tTest2
```

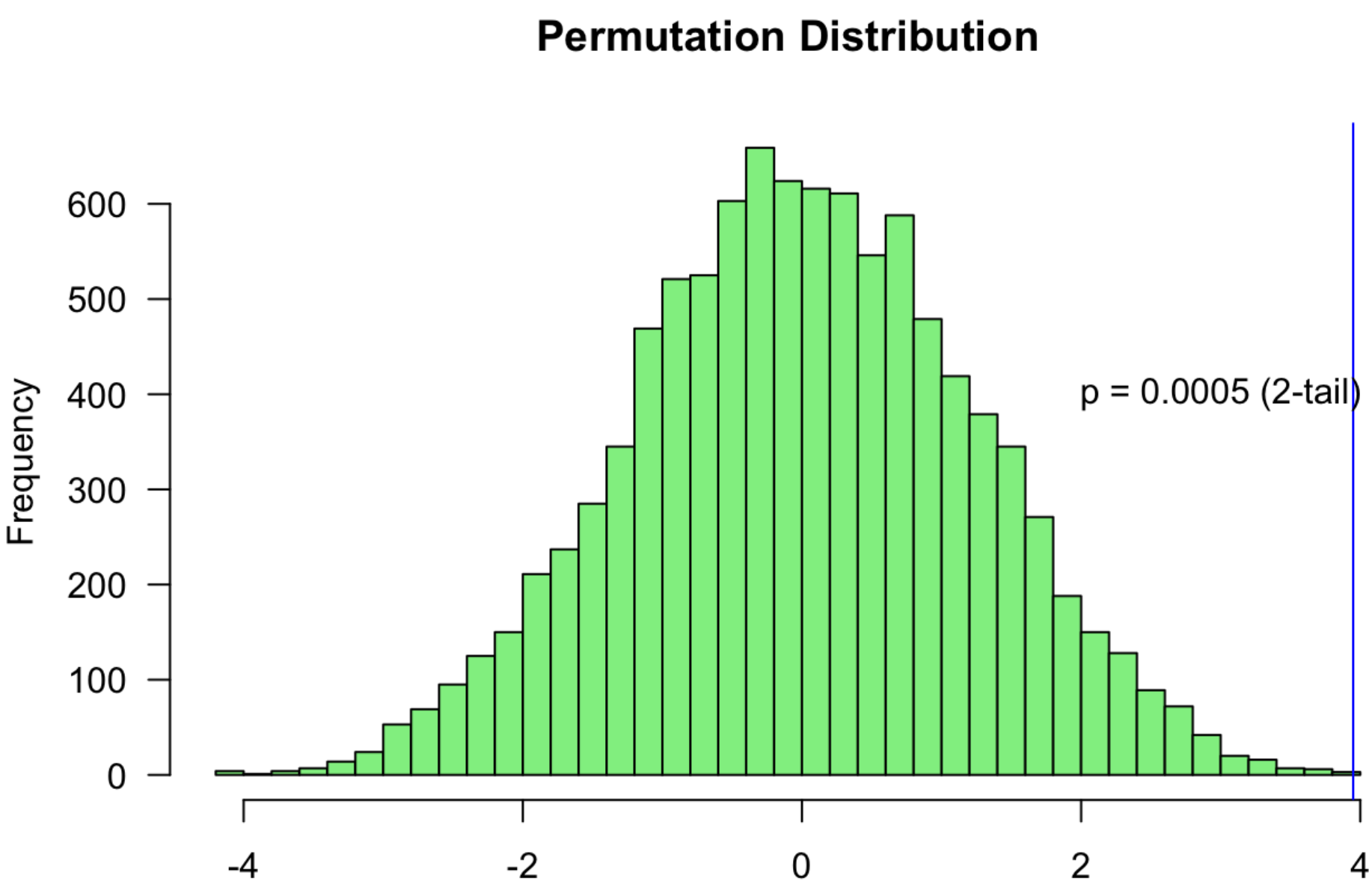
```
##
##  Paired t-test
##
## data:  table1$GroupA and table1$GroupB
## t = -3.7651, df = 34, p-value = 0.0006313
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -6.080395 -1.817493
## sample estimates:
## mean difference
##       -3.948944
```

- Significant result, suggesting that there is a mean difference between Morning and Evening tests; specifically, a lower test result in Evening tests for the same subjects by about 3.94

Permutation test

This method only works if values are symmetrically distributed

```
# Preparations
pairedPermTest <- function(x1, x2, n){
  observed <- x1 - x2
  m0 <- mean(observed)
  distro <- replicate(n, mean((rbinom(length(observed), 1, 0.5)*2-1)*observed))
  p <- sum(abs(distro) >= abs(m0))/n
  return(list(p, distro, abs(m0)))
}

# Main
pairedPerm <- pairedPermTest(table1$GroupA, table1$GroupB, 10000)

# Plot
hist(pairedPerm[[2]], breaks = 50, col = 'light green', main = "Permutation Distri
bution",
     las = 1, xlab = '')
abline(v = pairedPerm[[3]], col = "blue")
text(3, 400, paste("p =", pairedPerm[[1]], "(2-tail)"))
```



Similarly to p-value from the t-test, after 10,000 permutations only about 1 test value is as extreme as the observed difference, suggesting that the difference is not a result of random variation in the data. Provides a pairwise permutation test p-value of about p = 0.0009 (two-tailed).

Task 2 — Correlation and Bootstrapping

Correlation test

```
corTest1 <- cor.test(table2$x, table2$y)
corTest1
```

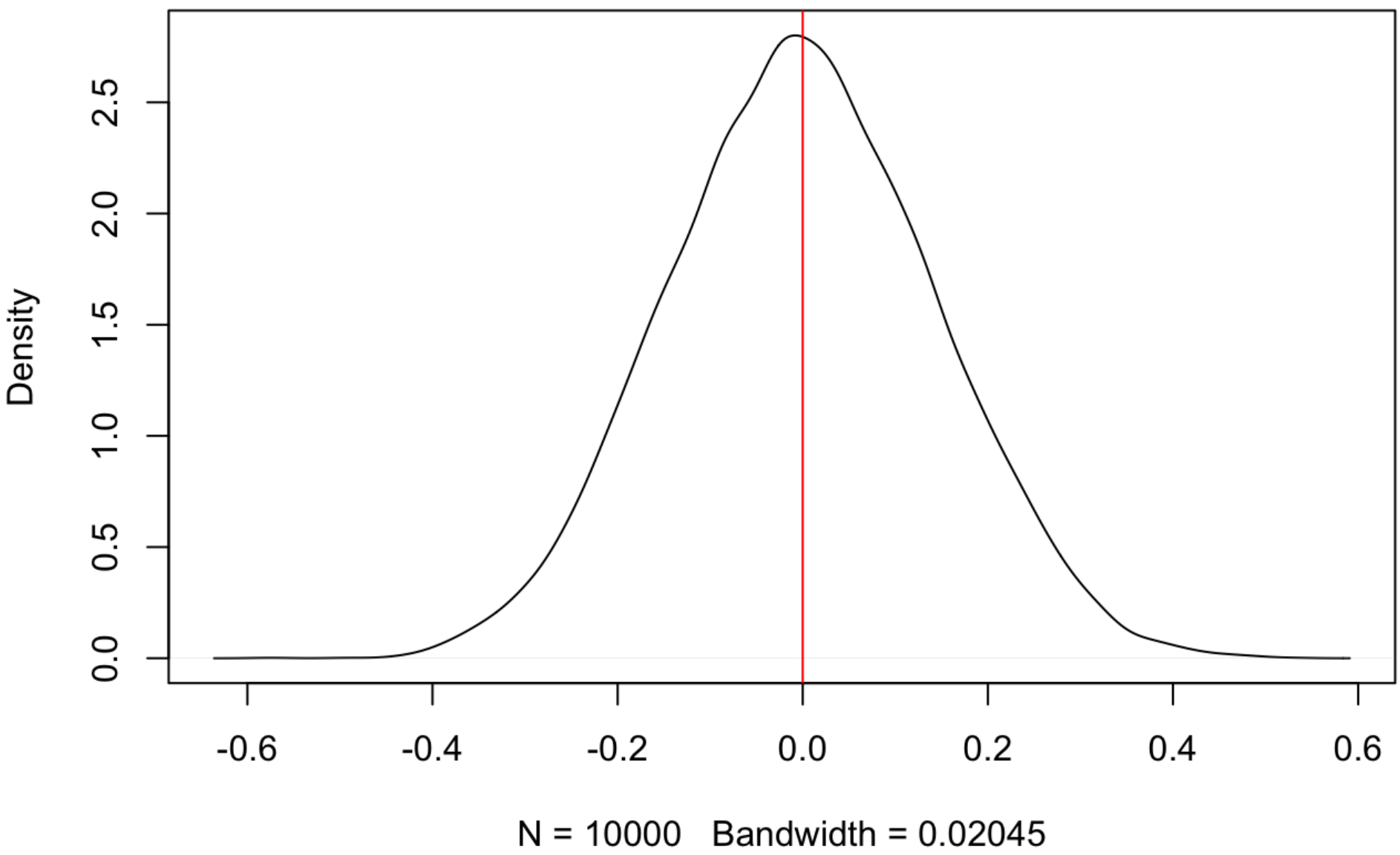
```
##
##  Pearson's product-moment correlation
##
## data:  table2$x and table2$y
## t = 4.1056, df = 48, p-value = 0.0001558
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2697256 0.6902055
## sample estimates:
##       cor
## 0.5097989
```

- A significant correlation between the two variables (r ≈ 0.5).

Bootstrapping (failed, don’t know why)

```
n <- 10000

bootCor <- matrix(NA, n, 6)
colnames(bootCor) <- c("stat", "param", "p", "cor", "ciLow", "ciUp")

for(i in 1:n){
  x <- sample(table2$x, length(table2$x), replace = TRUE)
  y <- sample(table2$y, length(table2$y), replace = TRUE)

  test <- cor.test(x, y)

  bootCor[i,] <- c(test$statistic, test$parameter, test$p.value,
                   test$estimate, test$conf.int[1], test$conf.int[2])
}

plot(density(bootCor[,"cor"]), main = "Distribution of Correlation Coefficients")
abline(v = 0, col = "red")
```

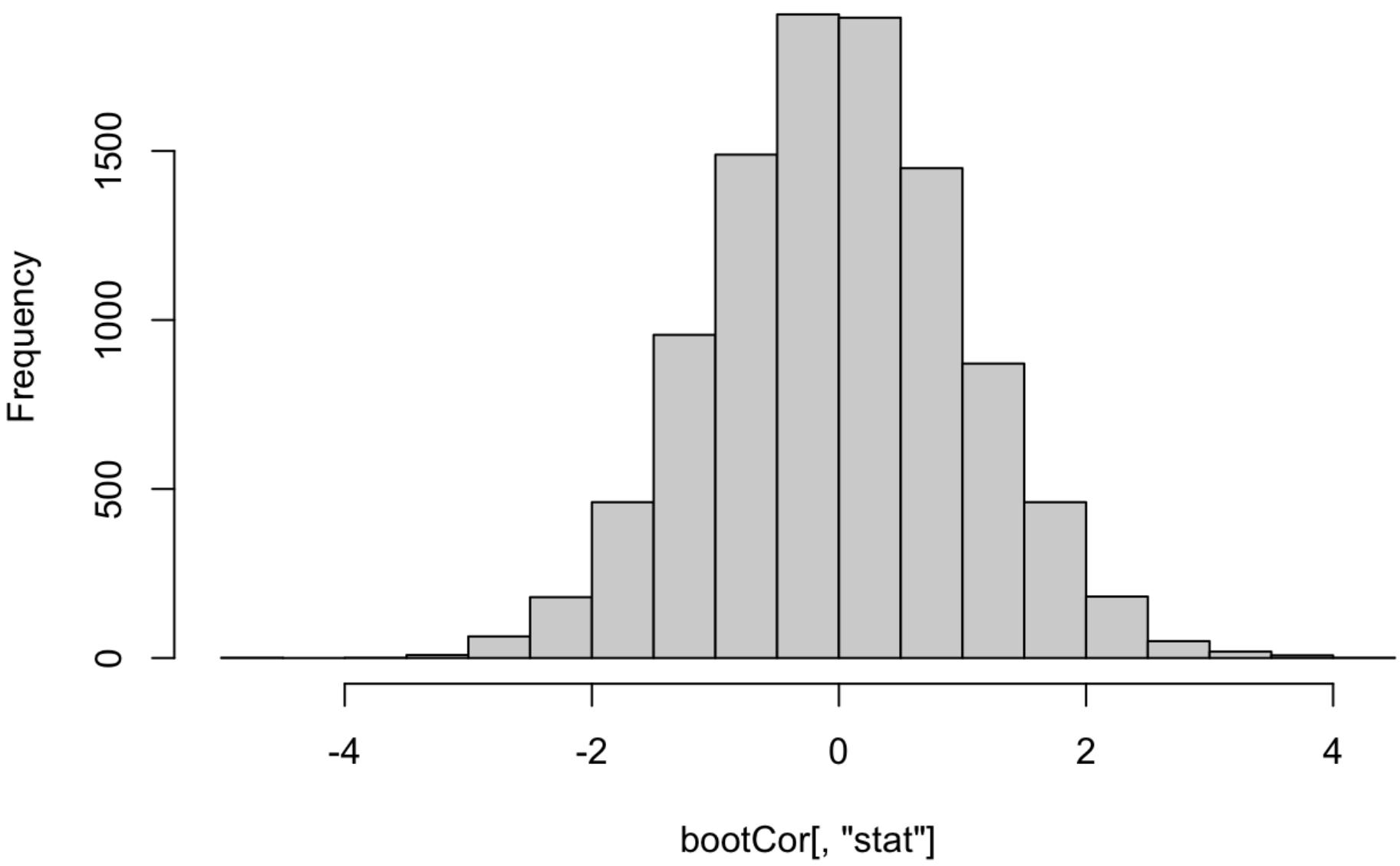
DON’T RANDOMIZE BOTH, ONLY THE INDEX FOR BOTH




```
hist(bootCor[,"stat"])
```

95CI =
sortedCorrDistro[95/100*n]

p-value is where 0 lies
(remember to multiply
by 2 and divide by n)

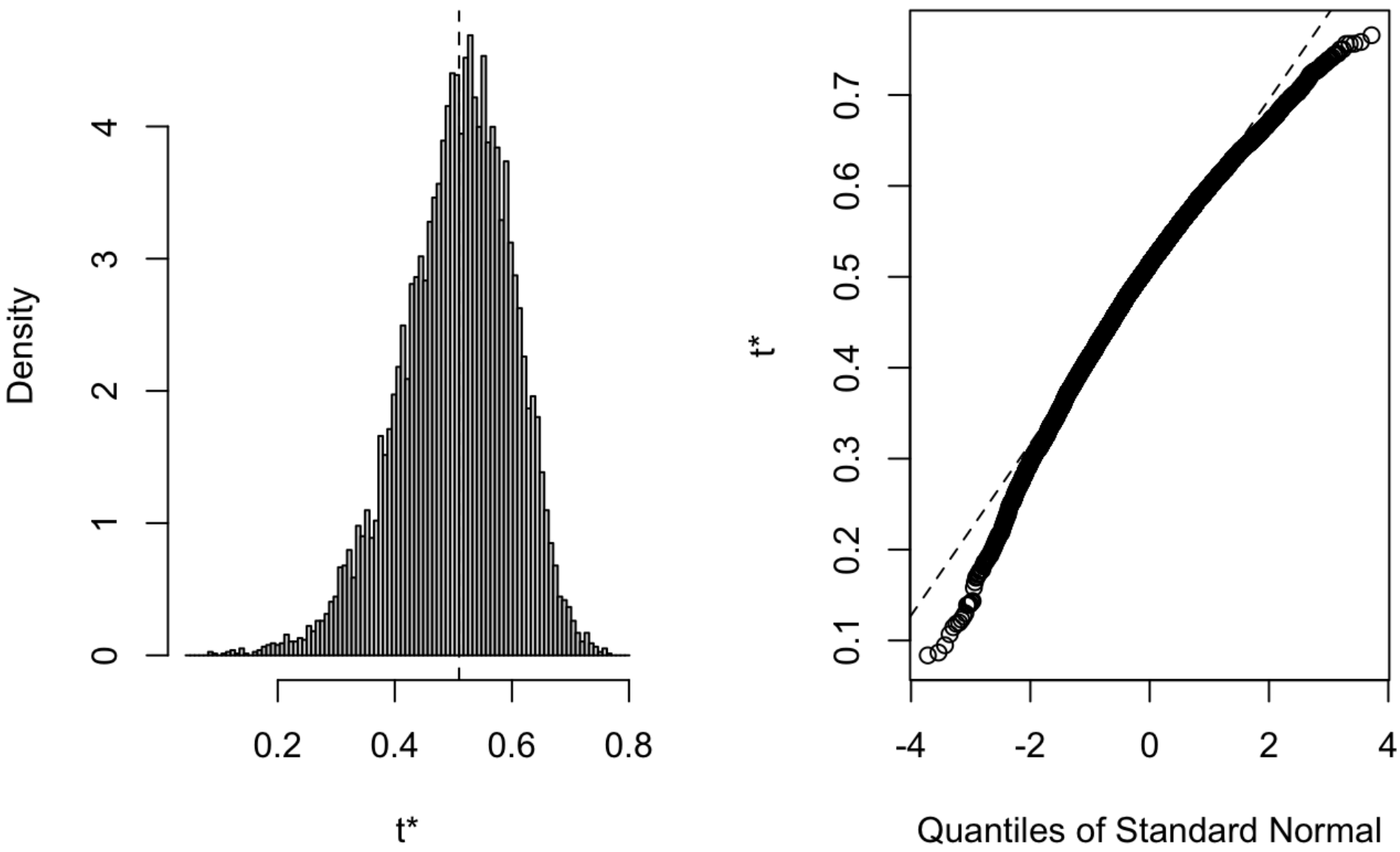


Bootstrapping (success using “boot” library)

```
corFun <- function(data, i)
{
  df <- data[i, ]

  c(cor(df[, 1], df[, 2], method = "pearson"))
}

bootCor1 <- boot(table2, corFun, 10000)
bootCor1
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = table2, statistic = corFun, R = 10000)
##
##
## Bootstrap Statistics :
##      original       bias    std. error
## t1* 0.5097989 -0.003884642  0.09457968
```

```
plot(bootCor1)
```




```
boot.ci(boot.out = bootCor1, type = c("norm", "basic", "perc", "bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = bootCor1, type = c("norm", "basic", "perc",
##     "bca"))
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 0.3283,  0.6991 )   ( 0.3532,  0.7182 )
##
## Level     Percentile            BCa
## 95%   ( 0.3014,  0.6664 )   ( 0.2982,  0.6650 )
## Calculations and Intervals on Original Scale
```

- Bootstrapping suggests also a significant correlation between variables x and y, with the 95%CI for the correlation coefficient (Pearson) being between 0.32 and 0.7.

Task 3 — Regression and Permutation

Functions to be used

```r
# Only preparations

# F-test
f_test <- function(n, p, r, SSE_h0, SSE){
  np_diff <- n - p
  SSE_diff <- SSE_h0 - SSE
  f_val <- (np_diff/r) * (SSE_diff/SSE)
  return(f_val)
}

# Regression function
regByHand <- function(y, varList){

  X <- matrix(c(unlist(varList)), ncol = length(varList))

  Beta <- solve(t(X) %*% X) %*% t(X) %*% y
  y_hat <- X %*% Beta
  error <- y - y_hat

  SSE <- t(error) %*% error

  outList <- list(rankMatrix(X)[1], X, Beta, y_hat, error, SSE)

  names(outList) <- c("rank", "X", "beta", "y_hat", "error", "SSE")

  return(outList)
}
```

Preparations

```r
table3$Clinic <- as.factor(table3$Clinic)
table3$Treatment <- as.factor(table3$Treatment)

n <- nrow(table3)

intercept <- rep(1, n)
clinGroup <- recode(table3$Clinic, "1" = 1, "2" = -1)
treatment <- recode(table3$Treatment, "1" = -1, "2" = 1)
interaction <- clinGroup * treatment

y <- table3$Outcome
```

Regressions

```r
h1 <- regByHand(y, list(intercept, clinGroup, treatment, interaction))
h0a <- regByHand(y, list(intercept, clinGroup, treatment))
h0b <- regByHand(y, list(intercept, clinGroup, interaction))
h0c <- regByHand(y, list(intercept, treatment, interaction))
```

F-tests

```
# F-tests
f_test(n, p = 4, r = 1, h0a$SSE, h1$SSE)
```

```
##           [,1]
## [1,] 0.7142146
```

```
f_test(n, p = 4, r = 1, h0b$SSE, h1$SSE)
```

```
##          [,1]
## [1,] 14.98498
```

```
f_test(n, p = 4, r = 1, h0c$SSE, h1$SSE)
```

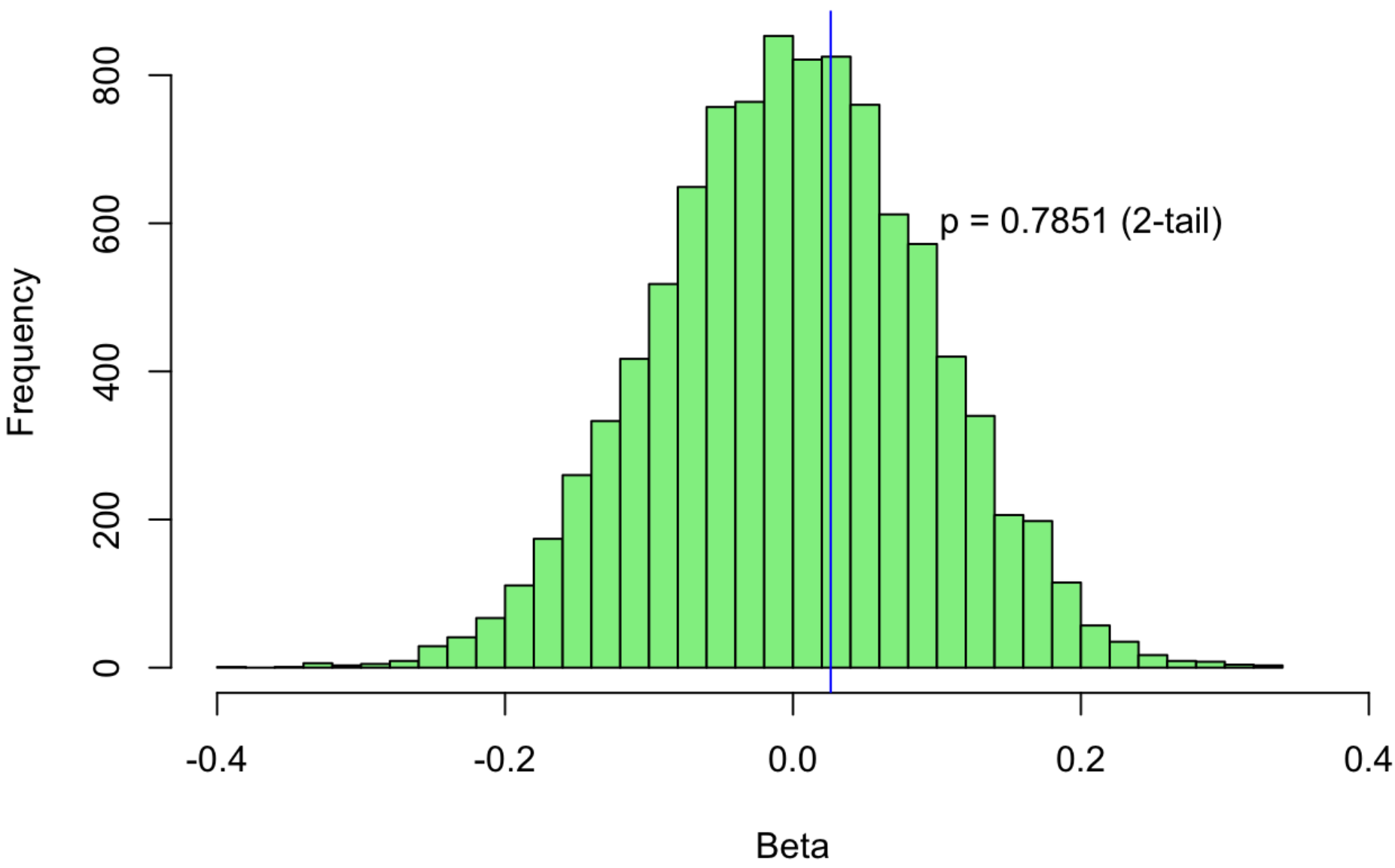
```
##            [,1]
## [1,] 0.07902128
```

Permutation

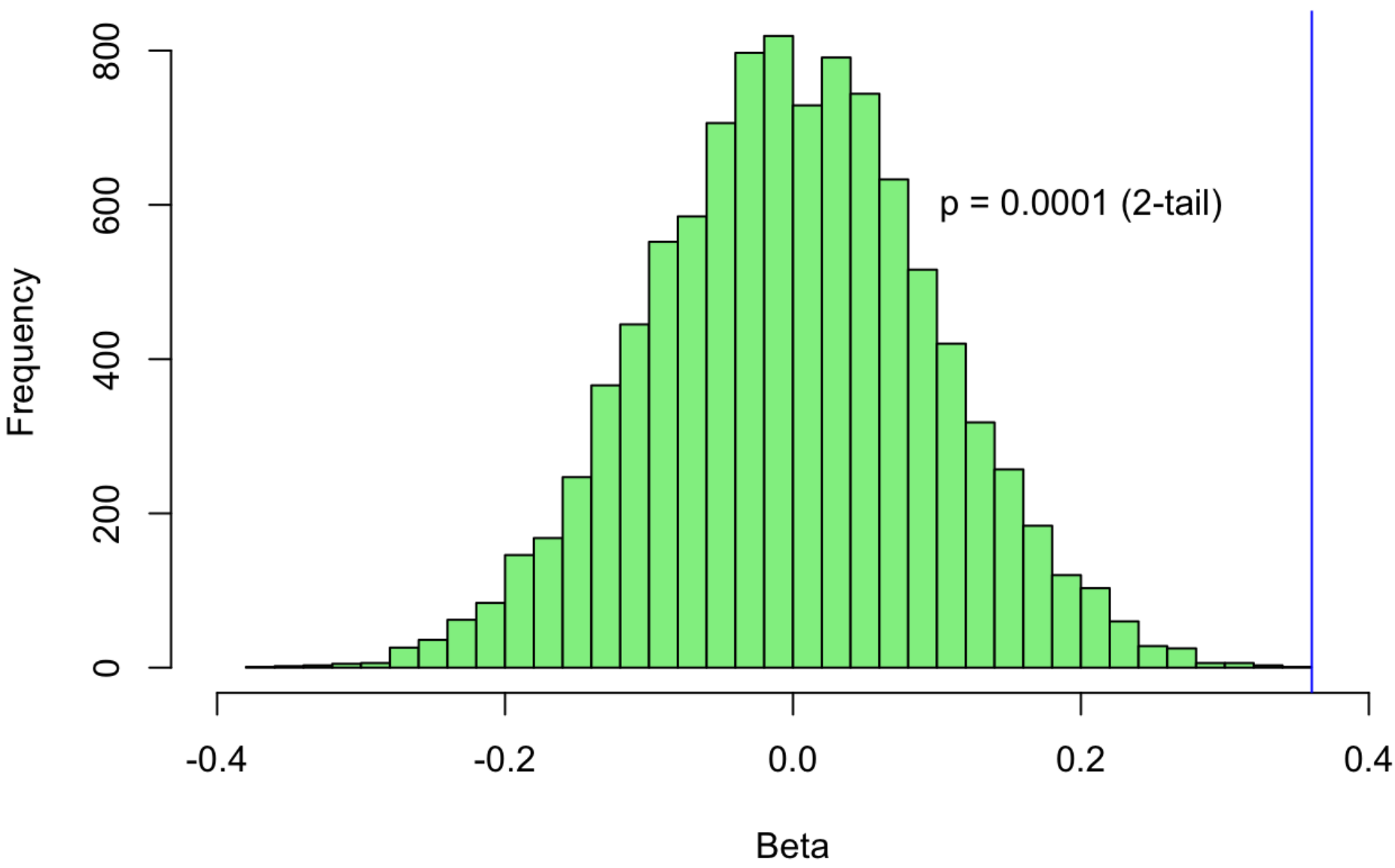
```
nBperms <- 10000

betaDistro <- matrix(NA, nBperms, 3)
for(i in 1:nBperms){
  clinPerm <- regByHand(y, list(intercept, sample(clinGroup), treatment, interacti
on))
  trtmPerm <- regByHand(y, list(intercept, clinGroup, sample(treatment), interacti
on))
  intrPerm <- regByHand(y, list(intercept, clinGroup, treatment, sample(interactio
n)))
  betaDistro[i,] <- c(clinPerm$beta[2], trtmPerm$beta[3], intrPerm$beta[4])
}
```

Plots

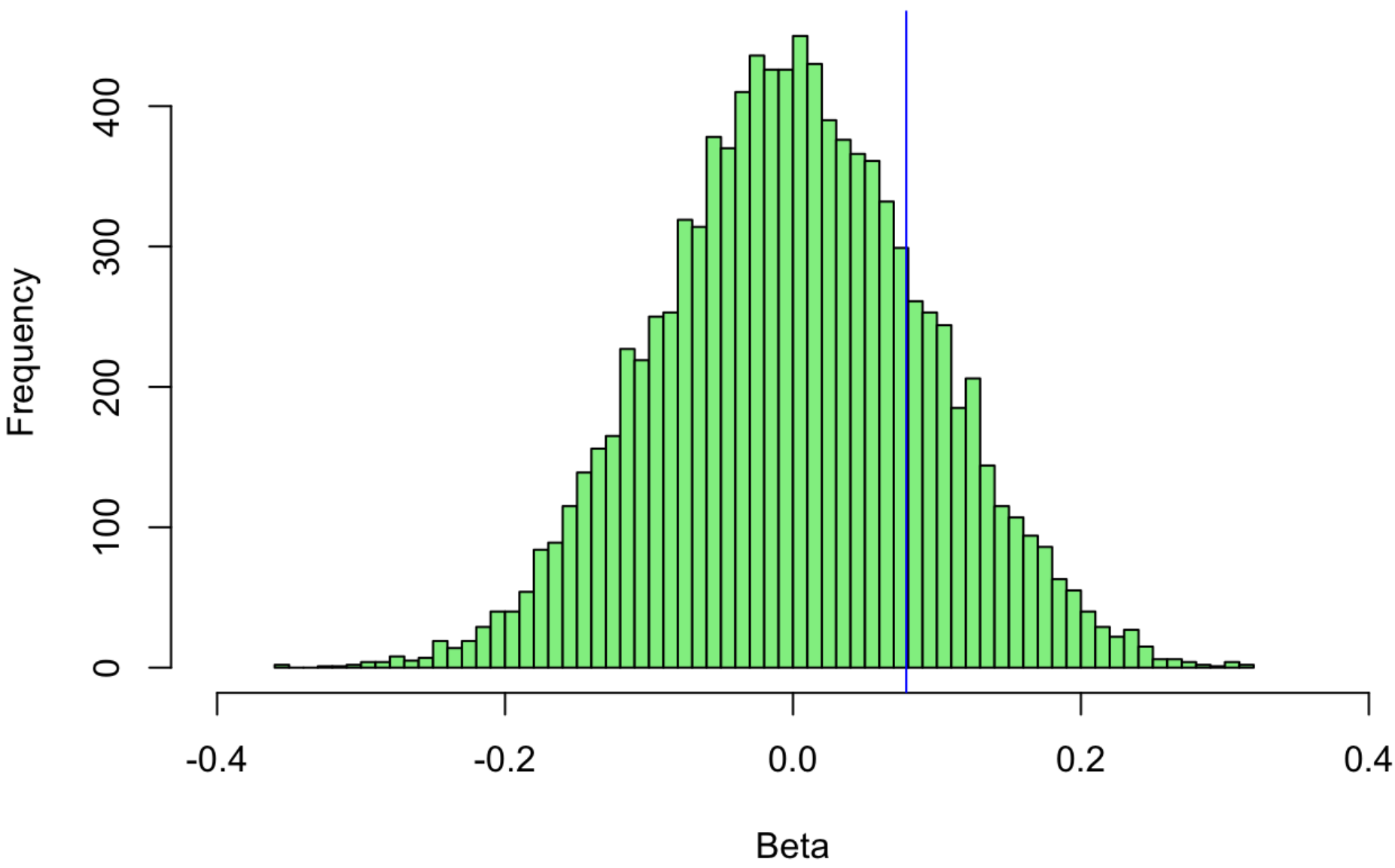
```
hist(betaDistro[,1], breaks = 50, col = "light green", main = "Permutation of Grou
p",
     xlab = "Beta", xlim = c(-0.4, 0.4))
abline(v = h1$beta[2], col = "blue")
text(0.2, 600, paste("p =", sum(abs(betaDistro[,1]) >= abs(h1$beta[2]))/nBperms, "
(2-tail)"))
```




```
hist(betaDistro[,2], breaks = 50, col = "light green", main = "Permutation of Trea
tment",
     xlab = "Beta", xlim = c(-0.4, 0.4))
abline(v = h1$beta[3], col = "blue")
text(0.2, 600, paste("p =", sum(abs(betaDistro[,2]) >= abs(h1$beta[3]))/nBperms, "
(2-tail)"))
```




```
hist(betaDistro[,3], breaks = 50, col = "light green", main = "Permutation of Inte
raction",
     xlab = "Beta", xlim = c(-0.4, 0.4))
abline(v = h1$beta[4], col = "blue")
text(0.2, 600, paste("p =", sum(abs(betaDistro[,3]) >= abs(h1$beta[4]))/nBperms, "
(2-tail)"))
```



- Permutation of each variable (including interaction) independently suggests treatment has the most influence, it's beta being the only one extreme enough to not be accounted for in random permutations

R regression

```
reg1 <- lm(Outcome ~ Clinic * Treatment, data = table3)
summary(reg1)
```

```
##
## Call:
## lm(formula = Outcome ~ Clinic * Treatment, data = table3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6573 -0.7177  0.0745  0.6055  2.3210
##
## Coefficients:
##                   Estimate Std. Error t value             Pr(>|t|)
## (Intercept)         4.9090     0.1862  26.369 < 0.0000000000000002 ***
## Clinic2             0.1050     0.2633   0.399              0.69077
## Treatment2          0.8780     0.2633   3.335              0.00115 **
## Clinic2:Treatment2 -0.3147     0.3723  -0.845              0.39979
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.02 on 116 degrees of freedom
## Multiple R-squared:  0.1197, Adjusted R-squared:  0.09697
## F-statistic: 5.259 on 3 and 116 DF,  p-value: 0.001944
```

If there is a three level factor, after permuting two columns of X, note down R-squared