HW_2

Nathan Laroy

2023-11-04

0) prepare data

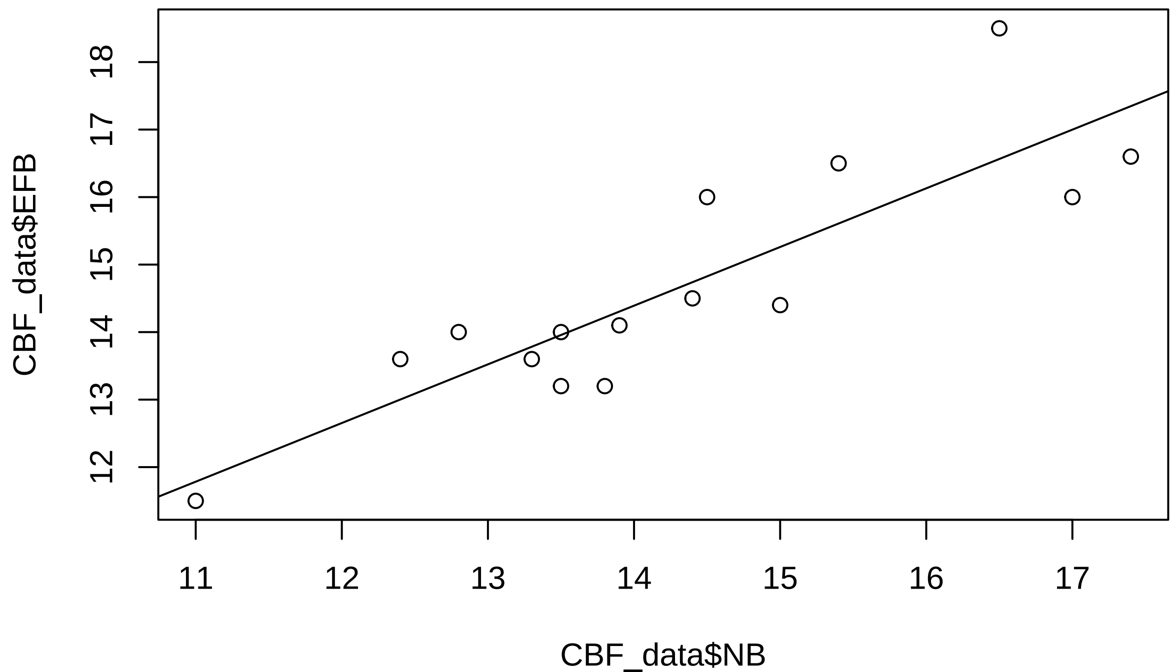
```r
NB <- c(15.4, 13.5, 13.3, 12.4, 12.8, 13.5, 14.5, 13.9,
        11.0, 15.0, 17.0, 13.8, 17.4, 16.5, 14.4)

EFB<- c(16.5, 13.2, 13.6, 13.6, 14.0, 14.0, 16.0, 14.1,
        11.5, 14.4, 16.0, 13.2, 16.6, 18.5, 14.5)

CBF_data <- data.frame("Subject" = c(1:15),
                       "NB" = NB,
                       "EFB" = EFB)
```

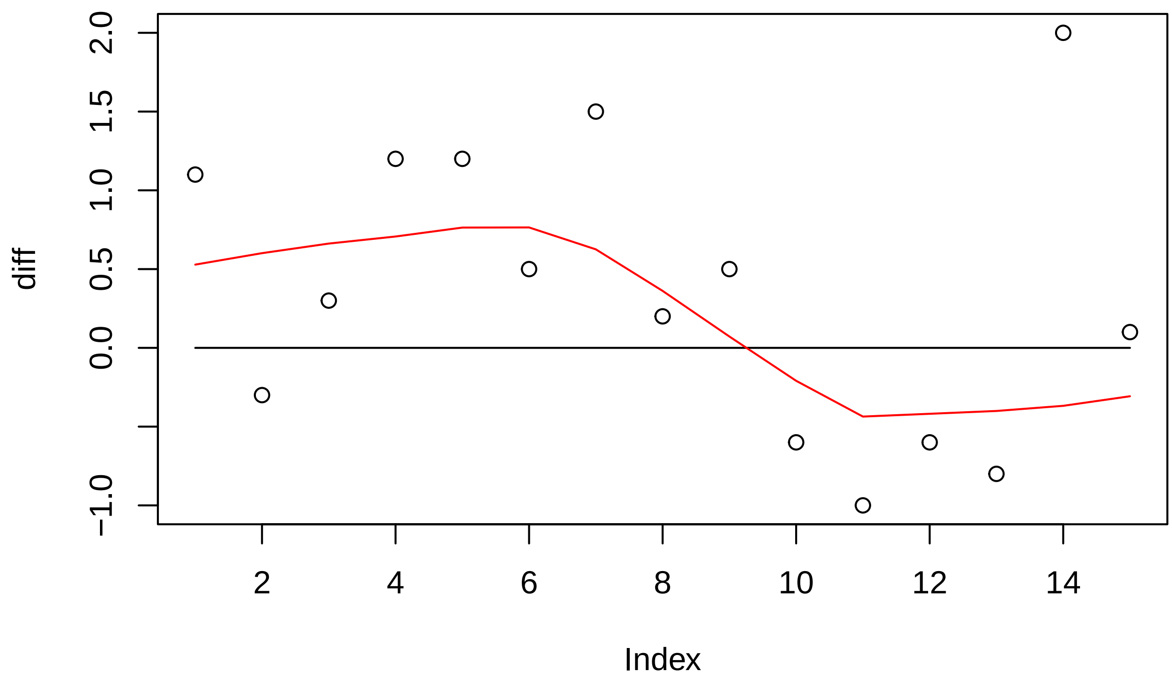
1) graphing and exploratory data analysis

```r
# standard scatterplot with LS line
plot(CBF_data$NB, CBF_data$EFB)
abline(lm(CBF_data$EFB ~ CBF_data$NB))
```

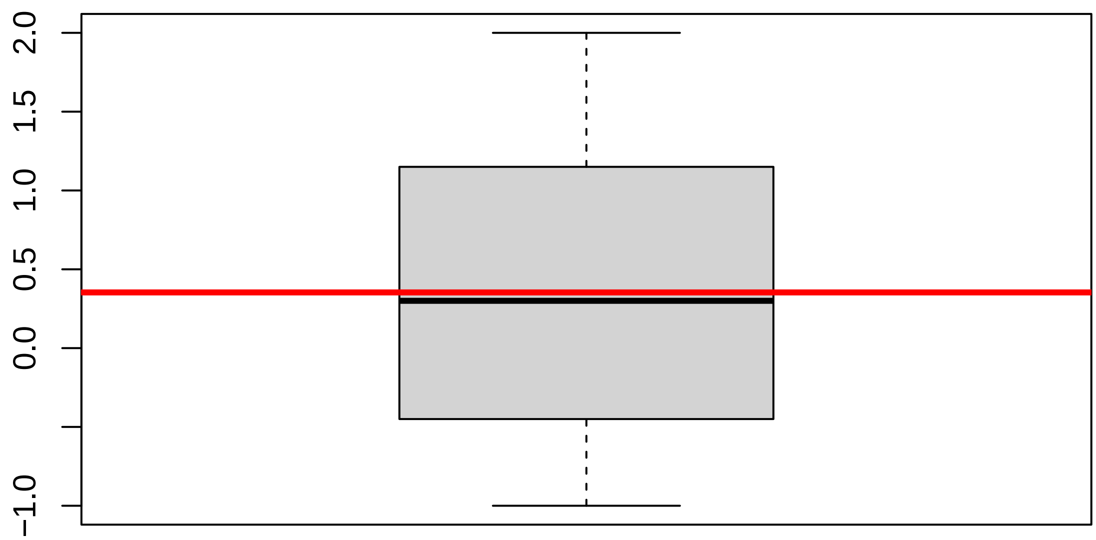


GRAPHS

```
# scatterplot of differences with smoother (ideally about zero)
diff <- CBF_data$EFB - CBF_data$NB
plot(diff)
lines(x = 1:15, y = rep(0, times=15))
lines(lowess(diff), col = "red")
```




```
# boxplot of differences with horizontal about mean
boxplot(diff)
abline(h = mean(diff), col = "red", lwd = 3)
```




```r
#### EXPLORATORY DATA ANALYSIS
# central Q: explore data wrt association
mean(diff)
```

```
## [1] 0.3533333
```

```r
sd(diff)
```

```
## [1] 0.9085834
```

```r
# other: in reference to graphs
```

2) Estimate Spearman Rank Correlation

The Spearman rank correlation is estimated at: 0.8771332.

The Spearman rank correlation coefficient is calculated as follows: first, both variables are ordered from least to greatest and ranked accordingly. Potential ties for each batch of observations are resolved via mid-ranks (i.e., the 'normal' ranks of the ties are averaged across the number of constituents of the respective tie). The coefficient is then mathematically defined as:

$$r_s = \frac{12\sum_{i=1}^{n}\left(\left[R_i - \frac{n+1}{2}\right]\left[S_i - \frac{n+1}{2}\right]\right)}{n(n^2-1)}$$

with R_i the ranks of the first variable's observations, S_i the ranks of the second variable's observations, and n the number of paired observations.

3) Permutation test based on Spearman's rank statistic

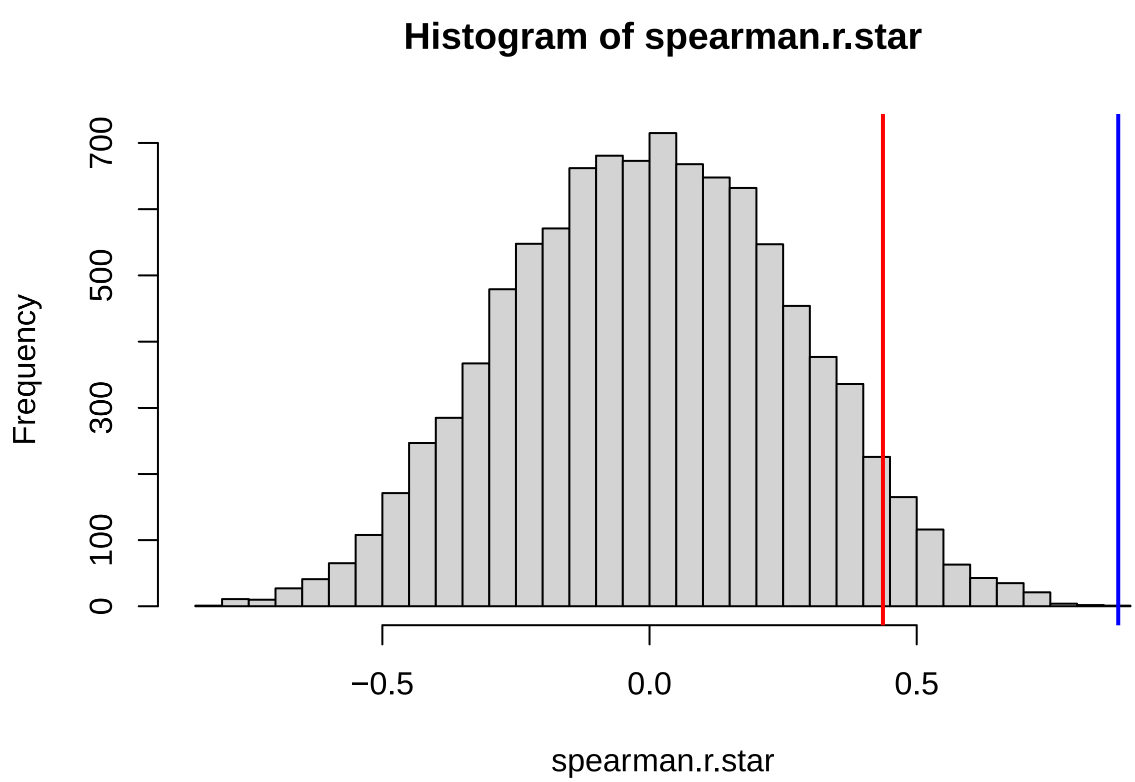
What does H_0 imply?

Under H_0, if there is no correlation between the NB and EFB measurements, then any observed value of NB can co-occur with any observed value of EFB.

To perform a permutation test, we first construct a permutation null distribution. What is to be permuted exactly? Given that under H_0 any NB value may co-occur with any EFB value, it is easiest to permute the NB vector, and calculate iteratively the Spearman correlation coefficient for each combination of EFB and permuted NB.

```r
set.seed(44)
obs.cor <- cor(NB, EFB, method = "spearman")

N <- 10000
spearman.r.star <- replicate(N, {
  NB_new <- sample(NB)
  spearman.r <- cor(NB_new, EFB, method = "spearman")
  return(spearman.r)
})
CritValues <- quantile(spearman.r.star, probs = c(0.95))
hist(spearman.r.star, breaks = 40)
abline(v = c(CritValues,obs.cor), col = c("red","blue")
                              , lwd = 2)
```



```r
print(c(max(spearman.r.star), obs.cor))
```

```
## [1] 0.8923799 0.8771332
```

```r
sum(spearman.r.star >= obs.cor)
```

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

```r
mean(spearman.r.star >= obs.cor)
```

```
## [1] 1e-04
```

4) Asymptotic approximation

standardized coefficients (with ties!)

```r
n <- 15
NB_ranked <- rank(NB) # 1 tie: 5.5 (2)
EFB_ranked <- rank(EFB) # 4 ties:
#                             2.5 (2)
#                             4.5 (2)
#                             6.5 (2)
#                             11.5 (2)
Diff_ranked <- NB_ranked - EFB_ranked
g <- 1
t_i <- 2
h <- 4
u_i <- 2
## r.star == obs.cor, because R calculates tie-corrected by default
r.star <- (n*(n^2 - 1) - 6*sum(Diff_ranked^2) - 0.5*(g*(t_i*(t_i^2 - 1)) +
                                                      h*(u_i*(u_i^2 - 1)))) /
  (((n*(n^2 - 1) - g*(t_i*(t_i^2 - 1))) * (n*(n^2 - 1)
                                            - h*(u_i*(u_i^2 - 1))))^0.5)

r.standard <- (n-1)^(0.5) * r.star
(p.value <- 1 - pnorm(r.standard))
```

```
## [1] 0.0005154923
```

```r
# can be doublechecked via perm.cor.test from "jmuOutlier" package)
```

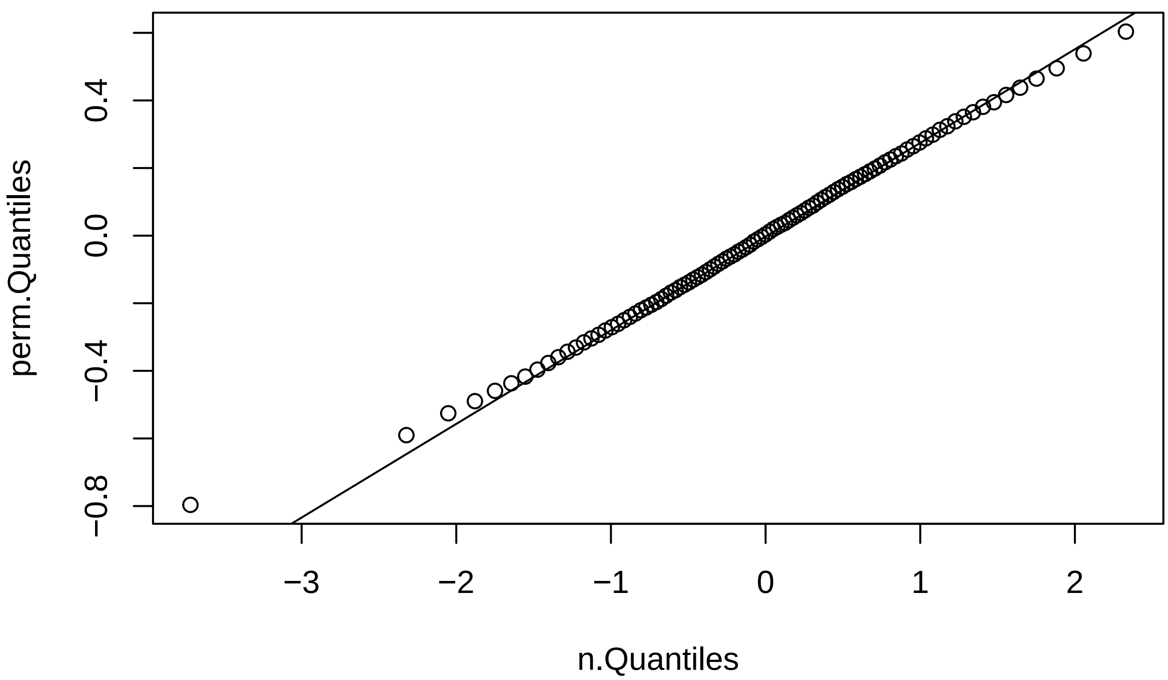
5) Compare two distributions

```r
probabilities <- seq(0.0001, 0.9999, by = 0.01)
probabilities[1:5]
```

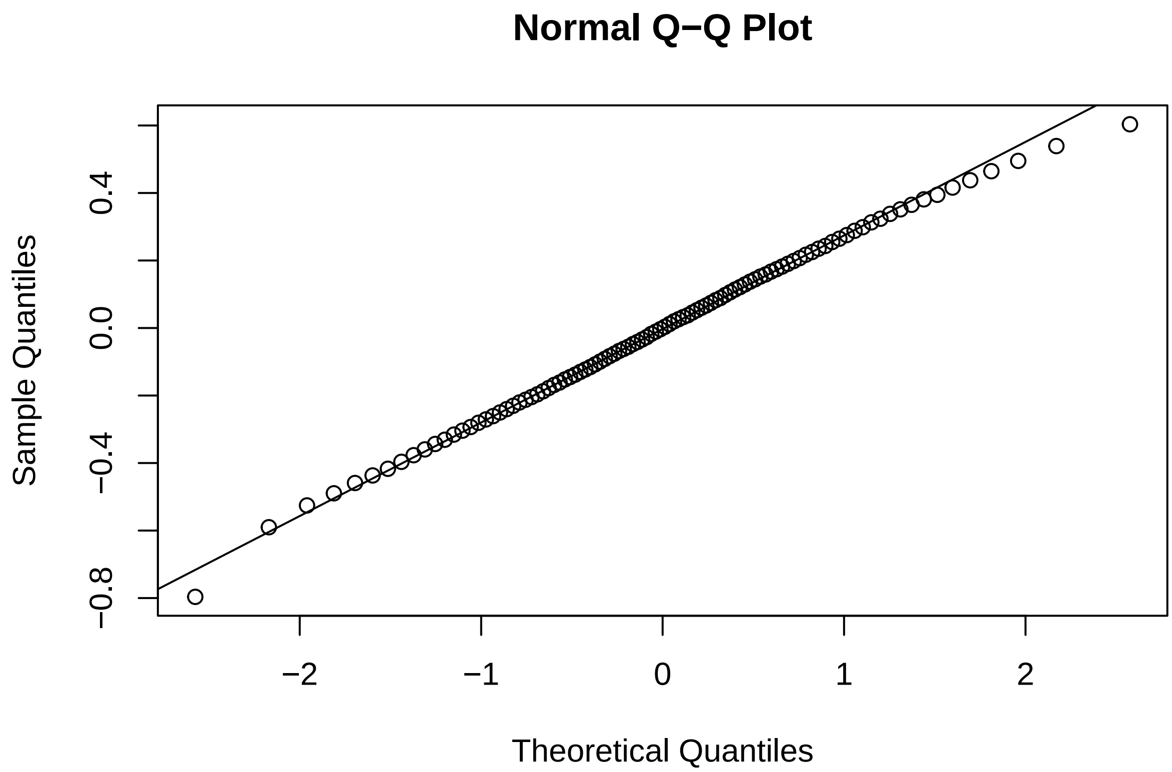
```
## [1] 0.0001 0.0101 0.0201 0.0301 0.0401
```

```
perm.Quantiles <- quantile(spearman.r.star, probabilities)
n.Quantiles <- qnorm(probabilities)

plot(n.Quantiles, perm.Quantiles)
qqline(perm.Quantiles)
```




```
# or
qqnorm(perm.Quantiles)
qqline(perm.Quantiles)
```


Normal Q−Q Plot
Sample Quantiles
−0.8
−0.4
0.0
0.4
−2
−1
0
1
2
Theoretical Quantiles