Entrega 3 - Rank tests

Coudet & Czarnievicz

Noviembre 2018

El código de esta entrega puede encontrarse en un repositorio de Github haciendo click aquí.

Ejercicio 4.1 BKN

Sabemos que

$$rs = 1 - \frac{6}{n(n^2-1)} \sum_{i=1}^{n} (R_i - i)^2 = 1 - \frac{12}{n(n^2-1)} \sum_{i<j}^{n} h_{ij}(j-i)$$

Entonces debemos demostrar que:

$$2 \sum_{i<j} h_{ij}(j-i) = \sum_{i=1}^{n} (R_i - i)^2 \Rightarrow$$

$$\Rightarrow 2 \sum_{i<j} h_{ij}(j-i) = \sum_{i=1}^{n} R_i^2 - 2 \sum_{i=1}^{n} R_i\, i + \sum_{i=1}^{n} i^2 \Rightarrow$$

$$\Rightarrow 2 \sum_{i<j} h_{ij}(j-i) = \frac{n(n+1)(2n+1)}{3} - 2 \sum_{i=1}^{n} R_i\, i$$

Ejercicio 4.4 BKN

```r
# Los datos
datos <- matrix(data=c(
   1, 8.0, 5.6, 6, 7.7, 6.1,
   2, 8.4, 7.4, 7, 7.7, 6.6,
   3, 8.0, 7.3, 8, 5.6, 6.0,
   4, 6.4, 6.4, 9, 5.6, 5.5,
   5, 8.6, 7.5, 10, 6.2, 5.5
   ), ncol=6, byrow=TRUE)
i <- datos[,c(1,4)]
dim(i) <- NULL
X <- datos[,c(2,5)]
dim(X) <- NULL
Y <- datos[,c(3,6)]
dim(Y) <- NULL
n <- length(X)

# Los rankings
R_x <- rank(X, ties.method="average")
R_y <- rank(Y, ties.method="average")

# El coef. de correlación de Spearman
numerador_S <- sum((R_x - (n+1)/2) * (R_y - (n+1)/2))
denominador_S <- sqrt(sum((R_x - (n+1)/2)^2) * sum((R_y - (n+1)/2)^2))
r_S <- numerador_S / denominador_S

# n suficientemente grande para una t
t_n <- sqrt(n-2) * r_S / sqrt(1 - r_S^2)
alpha <- 0.05
critical <- qt(1 - alpha/2, df=n-2)
if (abs(t_n) > critical) {
   print("Test de Spearman: Rechazo H0")
} else {
   print("Test de Spearman: No rechazo H0")
}
```

```
[1] "Test de Spearman: Rechazo H0"
```

```r
# El coef. tau_b de Kendall
U <- matrix(0, ncol=n, nrow=n)
for (i in 1:n) {
```

```
   for (j in 1:n) {
      if (X[j] - X[i] > 0 ) {
         U[i,j] <- 1
      } else if (X[j] - X[i] < 0) {
         U[i,j] <- -1
      } else { next }
   }
}
V <- matrix(0, ncol=n, nrow=n)
for (i in 1:n) {
   for (j in 1:n) {
      if (Y[j] - Y[i] > 0 ) {
         V[i,j] <- 1
      } else if (Y[j] - Y[i] < 0) {
         V[i,j] <- -1
      } else { next }
   }
}
numerador_K <- sum(U*V)
denominador_K <- sqrt(sum(U^2)*sum(V^2))
tau_b <- numerador_K / denominador_K

# La prueba de Kendall
S <- sum(U*V) / 2
nu_0 <- n*(n-1)*(2*n+5)
k_X <- n - n_distinct(X)
u_s <- ifelse(as.numeric(table(X)) > 1, as.numeric(table(X)), 0)
nu_u <- sum(u_s * (u_s - 1) * (2 * u_s + 5))
k_Y <- n - n_distinct(Y)
v_r <- ifelse(as.numeric(table(Y)) > 1, as.numeric(table(Y)), 0)
nu_v <- sum(v_r * (v_r - 1) * (2 * v_r + 5))
nu_uv1 <- sum(u_s * (u_s - 1)) * sum(v_r*(v_r - 1))
nu_uv2 <- sum(u_s * (u_s - 1) * (u_s - 2)) * sum(v_r * (v_r - 1) * (v_r - 2))
V_S = (nu_0 - nu_u - nu_v)/18 + nu_uv1 / (2*n*(n-1)) + nu_uv2 / (9*n*(n-1)*(n-2))
critical <- pnorm(1 - alpha / 2)
if (abs(S/sqrt(V_S)) > critical) {
   print("Test de Kendall: Rechazo H0")
} else {
   print("Test de Kendall: No rechazo H0")
}
```

```
[1] "Test de Kendall: Rechazo H0"
```

Ejercicio 4.7 BKN

```
datos <- matrix(data=c(
   1, 03.1, 09, 53.1, 1, 03.3, 09, 56.7,
   2, 09.4, 10, 59.4, 2, 10.0, 10, 63.3,
   3, 15.6, 11, 65.6, 3, 10.7, 11, 70.0,
   4, 21.9, 12, 71.9, 4, 23.3, 12, 76.7,
   5, 28.1, 13, 78.1, 5, 30.0, 13, 83.3,
   6, 34.4, 14, 84.4, 6, 36.7, 14, 90.0,
   7, 40.6, 15, 90.6, 7, 43.3, 15, 96.7,
   8, 46.9, 16, 96.9, 8, 50.0, NA, NA), nrow=8, byrow=TRUE)
datos <- rbind(datos[,c(1,2,6)], datos[,c(3,4,8)])
colnames(datos) <- c("i", "X1", "X2")
datos %>%
   as_tibble() %>%
   gather(key="variable", value="valores", -i) %>%
   filter(!is.na(valores)) %>%
   mutate(ranking = rank(valores, ties.method="average")) %>%
   filter(variable == "X1") -> ranking
W <- sum(ranking$ranking)
n <- sum(!is.na(datos[,"X1"]))
m <- sum(!is.na(datos[,"X2"]))
N <- n + m

# Test de Wilcoxon
E_W <- m * (N + 1) / 2
V_W <- n * m * (N + 1) / 12
Z_nm <- (W - E_W) / sqrt(V_W)
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_nm) > critical) {
   print("Test de Wilcoxon: rechazo H0")
} else {
   print("Test de Wilcoxon: no rechazo H0")
}
```

```
[1] "Test de Wilcoxon: no rechazo H0"
```

```
# Test de Van der Warden
ranking %>%
   select(ranking) %>%
   mutate(v_r = qnorm(ranking / (N+1))) -> v_r
```

```
V <- sum(v_r$v_r)
Q <- (1/N) * sum(v_r$v_r^2)
sigma_V <- (m*n*Q) / (N-1)
Z_mn <- V / sqrt(sigma_V)
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_mn) > critical) {
   print("Test de Van der Warden: rechazo H0")
} else {
   print("Test de Van der Warden: no rechazo H0")
}
```

```
[1] "Test de Van der Warden: no rechazo H0"
```

Ejercicio 5 HWC

```
# Los datos
olympic_kids <- c(12, 44, 34, 14, 9, 19, 156, 23, 13, 11,
                  47, 26, 14, 33, 15, 62, 5, 8, 0, 154, 146)
karate_kids <- c(37, 39, 30, 7, 13, 139, 45, 25, 16, 146,
                 94, 16, 23, 1, 290, 169, 62, 145, 36, 20, 13)
n <- length(olympic_kids) # El largo de X
m <- length(karate_kids) # El largo de Y
N <- n + m

# El estadístico W
cbind(olympic_kids, karate_kids) %>%
   as_tibble() %>%
   gather(key=key, value=value) %>%
   mutate(ranking = rank(value, ties.method="average")) %>%
   filter(key == "karate_kids") %$%
   sum(.$ranking) -> W

# La prueba de Wilcoxon (one sided - upper tail)
alpha <- 0.05
critical <- pwilcox(1 - alpha, n=n, m=m)
if (W >= critical) {
   print("Test de Wilcoxon (upper tail): rechazo H0")
} else {
   print("Test de Wilcoxon (upper tail): no rechazo H0")
}
```

```
[1] "Test de Wilcoxon (upper tail): rechazo H0"
```

```
# La prueba de Wilcoxon (one sided - lower tail)
alpha <- 0.05
critical <- pwilcox(alpha, n=n, m=m)
if (W <= n * (m + n +1) - critical) {
   print("Test de Wilcoxon (lower tail): rechazo H0")
} else {
   print("Test de Wilcoxon (lower tail): no rechazo H0")
}
```

```
[1] "Test de Wilcoxon (lower tail): rechazo H0"
```

```
# La prueba de Wilcoxon (two sided)
alpha <- 0.05
critical <- pwilcox(1 - alpha / 2, n=n, m=m)
if ((W >= critical) | (W <= n * (m + n + 1) - critical)) {
   print("Test de Wilcoxon (two sided): rechazo H0")
} else {
   print("Test de Wilcoxon (two sided): no rechazo H0")
}
```

```
[1] "Test de Wilcoxon (two sided): rechazo H0"
```

Ejercicio 9 HWC

```
# Los datos
olympic_kids <- c(12, 44, 34, 14, 9, 19, 156, 23, 13, 11,
                  47, 26, 14, 33, 15, 62, 5, 8, 0, 154, 146)
karate_kids <- c(37, 39, 30, 7, 13, 139, 45, 25, 16, 146,
                 94, 16, 23, 1, 290, 169, 62, 145, 36, 20, 13)
n <- length(olympic_kids) # El largo de X
m <- length(karate_kids) # El largo de Y
N <- n + m

# El estadísto de VdW
cbind(olympic_kids, karate_kids) %>%
   as_tibble() %>%
   gather(key=key, value=value) %>%
```

```
   mutate(ranking = rank(value, ties.method="average"),
          r_s = qnorm(ranking / (N + 1))) %>%
   filter(key == "karate_kids") -> ranking
v <- sum(ranking$r_s)

# La distribución del estadístico
Q <- (1/N) * sum(ranking$r_s^2)
sigma_V <- (m * n * Q) / (N - 1)
Z_mn <- V / sqrt(sigma_V)

# Test de Van der Warden
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_mn) > critical) {
   print("Test de Van der Warden: rechazo H0")
} else {
   print("Test de Van der Warden: no rechazo H0")
}
```

```
[1] "Test de Van der Warden: no rechazo H0"
```

Ejercicio 4.9 BKN

```
X <- c(09, 09, 08, 10, 12, 13, 10, 11, NA)
Y <- c(15, 16, 17, 23, 22, 20, 21, 24, 27)
n <- sum(!is.na(X))
m <- sum(!is.na(Y))
N <- m + n

# Siegel-Tukey test
s_r <- rep(NA_integer_, N)
s_r[1] <- 1
s_r[N] <- 2
i <- 1
while (s_r[i] < N) {
   if ((i+1) %% 2 == 0) {
      s_r[i+1] <- s_r[i] + 3
      s_r[N-i] <- s_r[i+1] - 1
      i <- i + 1
   } else {
```

```
      s_r[i+1] <- s_r[i] + 1
      s_r[N-i] <- s_r[i+1] + 1
      i <- i + 1
   }
   if (i == ceiling(N/2)) {
      s_r[ceiling(N/2)] <- N
   }
}
cbind(X,Y) %>%
   as_tibble() %>%
   gather() %>%
   filter(!is.na(value)) %>%
   mutate(ranking = rank(value, ties.method="average")) %>%
   arrange(ranking) %>%
   mutate(s_r = s_r) %>%
   filter(key == "X") %$%
   sum(.$s_r) -> S
E_S <- m * (N + 1) / 2
V_S <- m * n * (N + 1) / 12
Z_mn <- (S - E_S) / sqrt(V_S)
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_mn) > critical) {
   print("Test de Siegel-Tukey: rechazo H0")
} else {
   print("Test de Siegel-Tukey: no rechazo H0")
}
```

```
[1] "Test de Siegel-Tukey: rechazo H0"
```

```
# Ansari-Bradley
s_r <- rep(NA_integer_, N)
s_r[1] <- s_r[N] <- 1
i <- 1
while (s_r[i] < ceiling(N/2)) {
   s_r[i+1] <- s_r[i] + 1
   s_r[N-i] <- s_r[i+1]
   i = i + 1
}
cbind(X,Y) %>%
   as_tibble() %>%
   gather() %>%
   filter(!is.na(value)) %>%
```

```
   mutate(ranking = rank(value, ties.method="average")) %>%
   arrange(ranking) %>%
   mutate(s_r = s_r) %>%
   filter(key == "X") %$%
   sum(.$s_r) -> S
E_S <- m * (N + 1) / 4
V_S <- m * n * (N + 1)^2 / (48 * N)
Z_mn <- (S - E_S) / sqrt(V_S)
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_mn) > critical) {
   print("Test de Ansari-Bradley: rechazo H0")
} else {
   print("Test de Ansari-Bradley: no rechazo H0")
}
```

```
[1] "Test de Ansari-Bradley: rechazo H0"
```

```
# Test de Mood
cbind(X,Y) %>%
   as_tibble() %>%
   gather() %>%
   filter(!is.na(value)) %>%
   mutate(ranking = rank(value, ties.method="average"),
          s_r = (ranking - (N+1)/2)^2) %>%
   filter(key == "X") %$%
   sum(.$s_r) -> S
E_S <- m * (N^2 - 1) / 12
V_S <- m * n * (N + 1) * (N^2 - 4) / 180
Z_mn <- (S - E_S) / sqrt(V_S)
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_mn) > critical) {
   print("Test de Mood: rechazo H0")
} else {
   print("Test de Mood: no rechazo H0")
}
```

```
[1] "Test de Mood: no rechazo H0"
```

```
# Test de Klotz
s_r <- cbind(X,Y) %>%
```

```
   as_tibble() %>%
   gather() %>%
   filter(!is.na(value)) %>%
   mutate(ranking = rank(value, ties.method="average"),
          s_r = (qnorm(ranking / (N+1)))^2)
S <- s_r %>% filter(key == "X") %$% sum(.$s_r)
E_S <- (m/N) * sum(s_r$s_r)
V_S <- ((m*n) / (N * (N-1))) * sum(s_r$s_r^2) - n / (m*(N-1)) * E_S^2
Z_mn <- (S - E_S) / sqrt(V_S)
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (abs(Z_mn) > critical) {
   print("Test de Klotz: rechazo H0")
} else {
   print("Test de Klotz: no rechazo H0")
}
```

```
[1] "Test de Klotz: no rechazo H0"
```

Ejercicio 4.12 BKN

```
datos <- data.frame(
   expert=c("First", "Second", "Third", "Fourth","Fifth"),
   A=c(10, 5, 6, 3, 9),
   B=c(7, 2, 8, 4, 8),
   C=c(8, 4, 6, 6, 10)
)
k <- n_distinct(datos$expert)
n <- 3
datos %>%
   as_tibble() %>%
   gather(key=key, value=value, -expert) %>%
   group_by(expert) %>%
   mutate(ranking = rank(value)) %>%
   ungroup() %>%
   group_by(key) %>%
   summarise(Ri. = sum(ranking)) %$%
   sum((12 / (k^2 * n * (n^2 - 1))) * (.$Ri. - k * (n+1) / 2)^2) -> W

alpha <- 0.05
```

```
gamma <- (n-1)/2
eta <- (k-1)*gamma
critical <- pbeta(1-alpha, shape1=gamma, shape2=eta)

if (W > critical){
   print("Kendall's concordance test: Rechazo H0")
} else {
   print("Kendall's concordance test: No rechazo H0")
}
```

```
[1] "Kendall's concordance test: No rechazo H0"
```

Ejercicio 4.14 BKN

```
# Los datos
datos <- matrix(data=c(
   1, 8.0, 5.6, 6, 7.7, 6.1,
   2, 8.4, 7.4, 7, 7.7, 6.6,
   3, 8.0, 7.3, 8, 5.6, 6.0,
   4, 6.4, 6.4, 9, 5.6, 5.5,
   5, 8.6, 7.5, 10, 6.2, 5.5
   ), ncol=6, byrow=TRUE)
datos <- as_tibble(rbind(datos[,1:3], datos[,4:6]))
```

```
Warning: `as_tibble.matrix()` requires a matrix with column names or a `.name_repair` ar
This warning is displayed once per session.
```

```
colnames(datos) <- c('i','X','Y')
n <- dim(datos)[1]

# Wilcoxon signed rank test
datos %>%
   mutate(D_i = X - Y,
          abs_D_i = abs(D_i),
          ranking = rank(abs_D_i),
          signo = if_else(D_i < 0, '-', '+' )) %>%
   group_by(signo) %>%
   summarise(T_ = sum(ranking)) %>%
   filter(signo == '+') %>%
```

```
   select(T_) -> T_mas
T_mas <- as.numeric(T_mas)
E_T_mas <- n * (n + 1) / 4
V_T_mas <- n * (n + 1) * (2*n + 1) / 24
Z_n <- (T_mas - E_T_mas) / sqrt(V_T_mas)
datos %>%
   gather(key=key, value=value, -i) %>%
   group_by(value) %>%
   summarise(conteo = n()) %>%
   filter(conteo > 1) %$%
   sum(.$conteo^3-.$conteo) -> Te
Z_n_ast <- Z_n / sqrt(1 - Te / (2 * n * (n + 1) * (2 * n + 1)))
alpha <- 0.05
critical <- pnorm(1 - alpha / 2)
if (Z_n_ast > critical){
   print("Wilcoxon signed rank test: Rechazo H0")
} else {
   print("Wilcoxon signed rank test: No rechazo H0")
}
```

```
[1] "Wilcoxon signed rank test: Rechazo H0"
```