Entrega 2 - Tests basados en procesos empíricos

Coudet & Czarnievicz

September 2018

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

Reproducir figura 2.1

```
url <- "http://www.stat.cmu.edu/~larry/all-of-statistics/=data/nerve.dat"
nerve <- fread(url)
nerve <- as.matrix(nerve)
dim(nerve) <- NULL
nerve <- as_tibble(nerve) %>% filter(!is.na(value))
```

```
FALSE Warning: Calling `as_tibble()` on a vector is discouraged, because the behavior is
FALSE This warning is displayed once per session.
```

```
n <- dim(nerve)[1]
alpha <- 0.05
epsilon2 <- log(2/alpha) / (2*n)
nerve %>%
      group_by(value) %>%
      summarise(n_i = n()) %>%
      mutate(F.hat = cumsum(n_i)/sum(n_i),
             L=if_else(F.hat-sqrt(epsilon2) > 0, F.hat-sqrt(epsilon2), 0),
             U=if_else(F.hat+sqrt(epsilon2) < 1, F.hat+sqrt(epsilon2), 1),
             f.hat = n_i/sum(n_i)) %>%
      ggplot() +
      geom_line(aes(x=value, y=L), color="grey") +
      geom_line(aes(x=value, y=U), color="grey") +
            geom_line(aes(x=value, y=F.hat)) +
      geom_segment(aes(x=value, y=0, xend=value, yend=0.04, group=value)) +
      labs(x=NULL, y=NULL) +
      ggthemes::theme_economist()
```

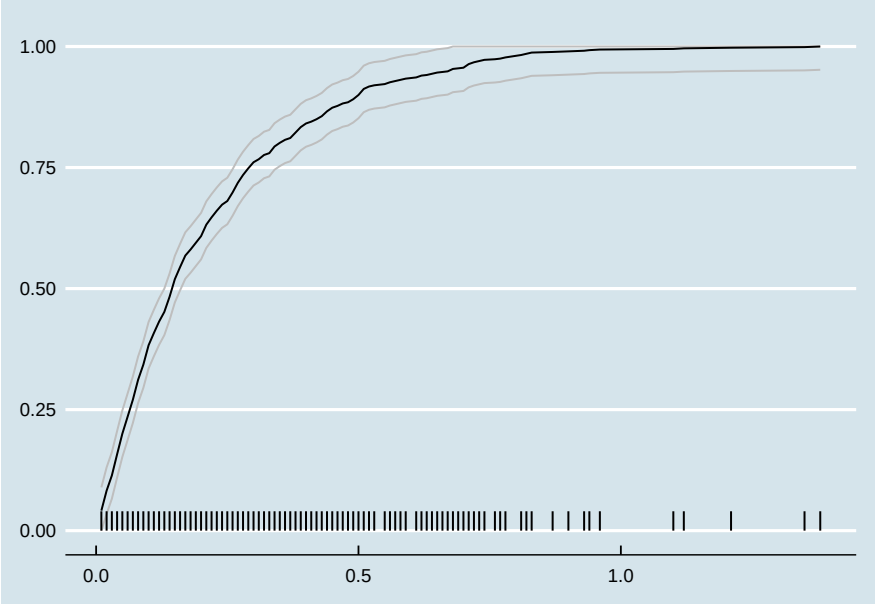

Figure 1: Figura 2.1 de Wasserman, página 14.

Ejercicio 3 Wasserman

```
n <- 100
alpha <- 0.05
epsilon2 <- log(2/alpha) / (2*n)
m <- 1000
results <- rep(NA, m)
counter <- 1
repeat {
      datos <- rnorm(n)
      datos <- sort(datos)
      as_tibble(datos) %>%
      group_by(value) %>%
      summarise(n_i = n()) %>%
      mutate(F.hat = cumsum(n_i)/sum(n_i),
             L = if_else(F.hat-sqrt(epsilon2) > 0, F.hat-sqrt(epsilon2), 0),
             U = if_else(F.hat+sqrt(epsilon2) < 1, F.hat+sqrt(epsilon2), 1),
             F.norm = pnorm(value),
             contains = if_else((F.norm > L) & (F.norm < U), TRUE, FALSE)) %$%
      table(.$contains)[1] %>%
      as.numeric() -> results[counter]
      if (counter == 1000){
            break
      } else {
```

```
            counter <- counter + 1
      }
}
table(results)
```

```
results
  1   2   3   4   5   6   7   8  10  12  13  14  23  28  30  31 100
  6   3   7   7   1   3   3   1   1   2   1   1   1   1   1   1 960
```

```
n <- 100
alpha <- 0.05
epsilon2 <- log(2/alpha) / (2*n)
m <- 1000
results <- rep(NA, m)
counter <- 1
repeat {
      datos <- rcauchy(n)
      datos <- sort(datos)
      as_tibble(datos) %>%
      group_by(value) %>%
      summarise(n_i = n()) %>%
      mutate(F.hat = cumsum(n_i)/sum(n_i),
             L = if_else(F.hat-sqrt(epsilon2) > 0, F.hat-sqrt(epsilon2), 0),
             U = if_else(F.hat+sqrt(epsilon2) < 1, F.hat+sqrt(epsilon2), 1),
             F.cau = pnorm(value),
             contains = if_else( (F.cau > L) & (F.cau < U), TRUE, FALSE)) %$%
      table(.$contains)[1] %>%
      as.numeric() -> results[counter]
      if (counter == 1000){
            break
      } else {
            counter <- counter + 1
      }
}
table(results)
```

```
results
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
 11  19  42  55  46  49  34  41  44  29  36  17  35  34  29  33  25  31
 19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
 22  23  26  22  31  18  19  24  18  14  21  16  19  10  14  16   9  12
 37  38  39  40  41  42  43  46  48  50  51  52  54  59  60 100
  5   6   9   9   4   1   5   3   2   2   3   1   1   1   1   3
```

Ejercicio 6 Wasserman

$$\mathrm{COV}\Big(\hat{F}_n(x), \hat{F}_n(y)\Big) = \mathrm{COV}\left(\frac{1}{n}\sum_{i=1}^{n} \mathrm{I}_{(-\infty,x]}(X_i), \frac{1}{n}\sum_{i=1}^{n} \mathrm{I}_{(-\infty,y]}(X_i)\right) =$$

$$= \frac{1}{n^2}\ \mathrm{COV}\left(\sum_{i=1}^{n} \mathrm{I}_{(-\infty,x]}(X_i), \sum_{i=1}^{n} \mathrm{I}_{(-\infty,y]}(X_i)\right) =$$

$$= \frac{1}{n^2}\ \sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{COV}\left(\mathrm{I}_{(-\infty,x]}(X_i), \mathrm{I}_{(-\infty,y]}(X_j)\right) =$$

Ahora bien, dado que las X_i son independientes, la covarianza de los sucesos $\left\{\mathrm{I}_{(-\infty,x]}(X_i)\right\}$, $\left\{\mathrm{I}_{(-\infty,y]}(X_j)\right\}$ vale 0. Por lo tanto, en la sumatoria anterior sobreviven los términos con $i = j$

$$\frac{1}{n^2}\sum_{i=1}^{n} \mathrm{COV}(\mathrm{I}_{(-\infty,x]}(X_i), \mathrm{I}_{(-\infty,y]}(X_i)) + \frac{1}{n^2}\sum_{i\neq j}\sum \underbrace{\mathrm{COV}(\mathrm{I}_{(-\infty,x]}(X_i), \mathrm{I}_{(-\infty,y]}(X_j))}_{=0} =$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} \mathrm{COV}(\mathrm{I}_{(-\infty,x]}(X_i), \mathrm{I}_{(-\infty,y]}(X_i)) =$$

$$= \frac{1}{n^2}\mathrm{E}\left[\mathrm{I}_{(-\infty,x]}(X_i)\mathrm{I}_{(-\infty,y]}(X_i)\right] - \frac{1}{n^2}\underbrace{\mathrm{E}\left[\mathrm{I}_{(-\infty,x]}(X_i)\right]}_{P(X\leq x)=\mathrm{F}(x)}\underbrace{\mathrm{E}\left[\mathrm{I}_{(-\infty,y]}(X_i)\right]}_{P(X\leq y)=\mathrm{F}(y)} =$$

Asumiendo, sin pérdida de generalidad, que $x < y$:

$$= \frac{1}{n^2}\mathrm{E}\left[\mathrm{I}_{(-\infty,x]}(X_i)\right] - \frac{1}{n^2}\mathrm{F}(x)\mathrm{F}(y) = \frac{1}{n^2}\mathrm{F}(x)(1-\mathrm{F}(y))$$

Por lo tanto,

$$\boxed{\mathrm{COV}\Big(\hat{F}_n(x), \hat{F}_n(y)\Big) = \frac{1}{n^2}\mathrm{F}(x)(1-\mathrm{F}(y))}$$

Ejercicio 3.10 BKN

```r
# Los datos
x <- c(338, 336, 312, 322, 381, 302, 296, 360, 342, 334, 348, 304, 323, 310,
       368, 341, 298, 312, 322, 350, 304, 302, 336, 334, 304, 292, 324, 331,
       324, 334, 314, 338, 324, 292, 298, 342, 338, 331, 325, 324, 326, 314,
       312, 362, 368, 321, 352, 304, 302, 332, 314, 304, 312, 381, 290, 322,
       326, 316, 328, 340, 324, 320, 364, 304, 340, 290, 318, 332, 354, 324,
       304, 321, 356, 366, 328, 332, 304, 282, 330, 314, 342, 322, 362, 298,
       316, 298, 332, 342, 316, 326, 308, 321, 302, 304, 322, 296, 322, 338,
       324, 323)
x <- sort(x)
n <- length(x)
alpha <- 0.05
```

```r
# KS test
Dmas <- max((1/n)*seq(1, n, 1) - pnorm(x, mean=mean(x), sd=sd(x)))
Dmenos <- max(pnorm(x, mean=mean(x), sd=sd(x)) - (1/n)*seq(0, n-1, 1))
Dn <- max(Dmas,Dmenos)
k <- seq(1, 10000, 1)
K_x <- 1 + 2 * sum((exp(-2 * (sqrt(n) * Dn)^2 * k^2)) %*% ((-1)^k))
pv_a <- 1 - K_x
if (pv_a > alpha) {
      print("Rechazo H0")
} else {
      print("No Rechazo H0")
}
```

```
[1] "Rechazo H0"
```

```r
# AD test
w <- seq(1, 2*n-1, 2)
F0 <- pnorm(x, mean=mean(x), sd=sd(x))
ad <- -n - (1/n)*sum(w*(log(sort(F0)) + log(1 - sort(F0, decreasing=TRUE))))
if(ad.test(F0, null="punif")$p.value > alpha) {
      print("Rechazo H0")
} else {
      print("No Rechazo H0")
}
```

```
[1] "Rechazo H0"
```

```r
# CvM test
F0 <- pnorm(x, mean=mean(x), sd=sd(x))
u <- seq(1, 2*n-1, 2)
cvm <- (1/(12*n)) + sum((sort(F0) - (u/(2*n)))^2)
if(cvm.test(F0, null="punif")$p.value > alpha) {
      print("Rechazo H0")
} else {
      print("No Rechazo H0")
}
```

```
[1] "Rechazo H0"
```

Ejercicio 3.11 BKN

```r
x <- c(10, 51, 08, 47, 08, 05, 56, 12, 04, 05, 04, 04, 07, 06, 09, 30, 25,
       12, 03, 22, 05, 15, 04, 04, 29, 15, 04, 02, 18, 41, 03, 05, 54, 110,
       24, 16, 02, 37, 20, 02, 06, 07, 16, 02, 14, 68, 10, 16, 11, 78, 06,
       17, 07, 11, 21, 15, 24, 06, 32, 08, 11, 04, 14, 45, 17, 10, 15, 20,
       04, 65, 10, 03, 05, 11, 13, 35, 11, 34, 03, 04, 12, 07, 06, 62, 13,
       36, 26, 06, 11, 06, 13, 01, 04, 36, 18, 10, 37, 28, 04, 12, 31, 14,
       03, 11, 06, 04, 10, 38, 06, 11, 24, 09, 04, 05, 08, 22, 06, 16, 135,
       18, 49, 17, 09, 32, 27, 02, 12, 08, 93, 03, 09, 10, 03, 14, 33, 72,
       14, 04, 09, 10, 19, 02, 05, 21, 08, 25, 30, 20, 12, 19)
x <- log(sort(x))
n <- length(x)
```

```r
# KS test
Dmas <- max((1/n)*seq(1, n, 1) - pnorm(x, mean=mean(x), sd=sd(x)))
Dmenos <- max(pnorm(x, mean=mean(x), sd=sd(x)) - (1/n)*seq(0, n-1, 1))
Dn <- max(Dmas,Dmenos)
k <- seq(1, 10000, 1)
K_x <- 1 + 2 * sum((exp(-2 * (sqrt(n) * Dn)^2 * k^2)) %*% ((-1)^k))
pv_a <- 1 - K_x
if (pv_a > alpha) {
      print("Rechazo H0")
} else {
      print("No Rechazo H0")
}
```

```
[1] "Rechazo H0"
```

```r
# AD test
w <- seq(1, 2*n-1, 2)
F0 <- pnorm(x, mean=mean(x), sd=sd(x))
ad <- -n - (1/n)*sum(w*(log(sort(F0)) + log(1 - sort(F0, decreasing=TRUE))))
if(ad.test(F0, null="punif")$p.value > alpha) {
      print("Rechazo H0")
} else {
      print("No Rechazo H0")
}
```

```
[1] "Rechazo H0"
```

```r
# CvM test
F0 <- pnorm(x, mean=mean(x), sd=sd(x))
u <- seq(1, 2*n-1, 2)
cvm <- (1/(12*n)) + sum((sort(F0) - (u/(2*n)))^2)
if(cvm.test(F0, null="punif")$p.value > alpha) {
      print("Rechazo H0")
} else {
      print("No Rechazo H0")
}
```

```
[1] "Rechazo H0"
```

Ejercicio 3.15 BKN

```r
# KS test for two samples
x <- c(338, 336, 312, 322, 381, 302, 296, 360, 342, 334, 348, 304, 323, 310,
       368, 341, 298, 312, 322, 350, 304, 302, 336, 334, 304, 292, 324, 331,
       324, 334, 314, 338, 324, 292, 298, 342, 338, 331, 325, 324, 326, 314,
       312, 362, 368, 321, 352, 304, 302, 332, 314, 304, 312, 381, 290, 322,
       326, 316, 328, 340, 324, 320, 364, 304, 340, 290, 318, 332, 354, 324,
       304, 321, 356, 366, 328, 332, 304, 282, 330, 314, 342, 322, 362, 298,
       316, 298, 332, 342, 316, 326, 308, 321, 302, 304, 322, 296, 322, 338,
       324, 323)
dim(x) <- c(10, 10)
x <- t(x)
y <- x[, 6:10]
x <- x[, 1:5]
```

```r
dim(x) <- NULL
dim(y) <- NULL
x <- sort(x)
y <- sort(y)
m <- length(x)
n <- length(y)
F.hat.y <- as_tibble(y) %>%
      group_by(value) %>%
      summarise(n_i = n()) %>%
      mutate(F.hat = cumsum(n_i)/sum(n_i)) %>%
      dplyr::select(-n_i) %>%
      add_row(.,value=-Inf, F.hat=0, .before=1) %>%
      rename(inferior = value) %>%
      mutate(superior = lead(inferior, default=+Inf)) %>%
      dplyr::select(inferior, superior, F.hat)
F.hat.x <- NULL
for (i in 1:length(x)){
      for (j in 1:dim(F.hat.y)[1]) {
            if ((x[i] >= F.hat.y$inferior[j]) & (x[i] < F.hat.y$superior[j])) {
                  F.hat.x[i] <- F.hat.y$F.hat[j]
            } else {
                  next
            }
      }
}
Dmas <- max((1/m)*seq(1, m, 1) - F.hat.x)
Dmenos <- max(F.hat.x - (1/m)*seq(0, m-1, 1))
Dn <- max(Dmas,Dmenos)
# k <- seq(1, 10000, 1)
# K_x <- 1 + 2 * sum((exp(-2 * (sqrt(n) * Dn)^2 * k^2)) %*% ((-1)^k))
# pv_a <- 1 - K_x
# if (pv_a > alpha) {
#       print("Rechazo H0")
# } else {
#       print("No Rechazo H0")
# }
```

Ejercicio 3.16 BKN