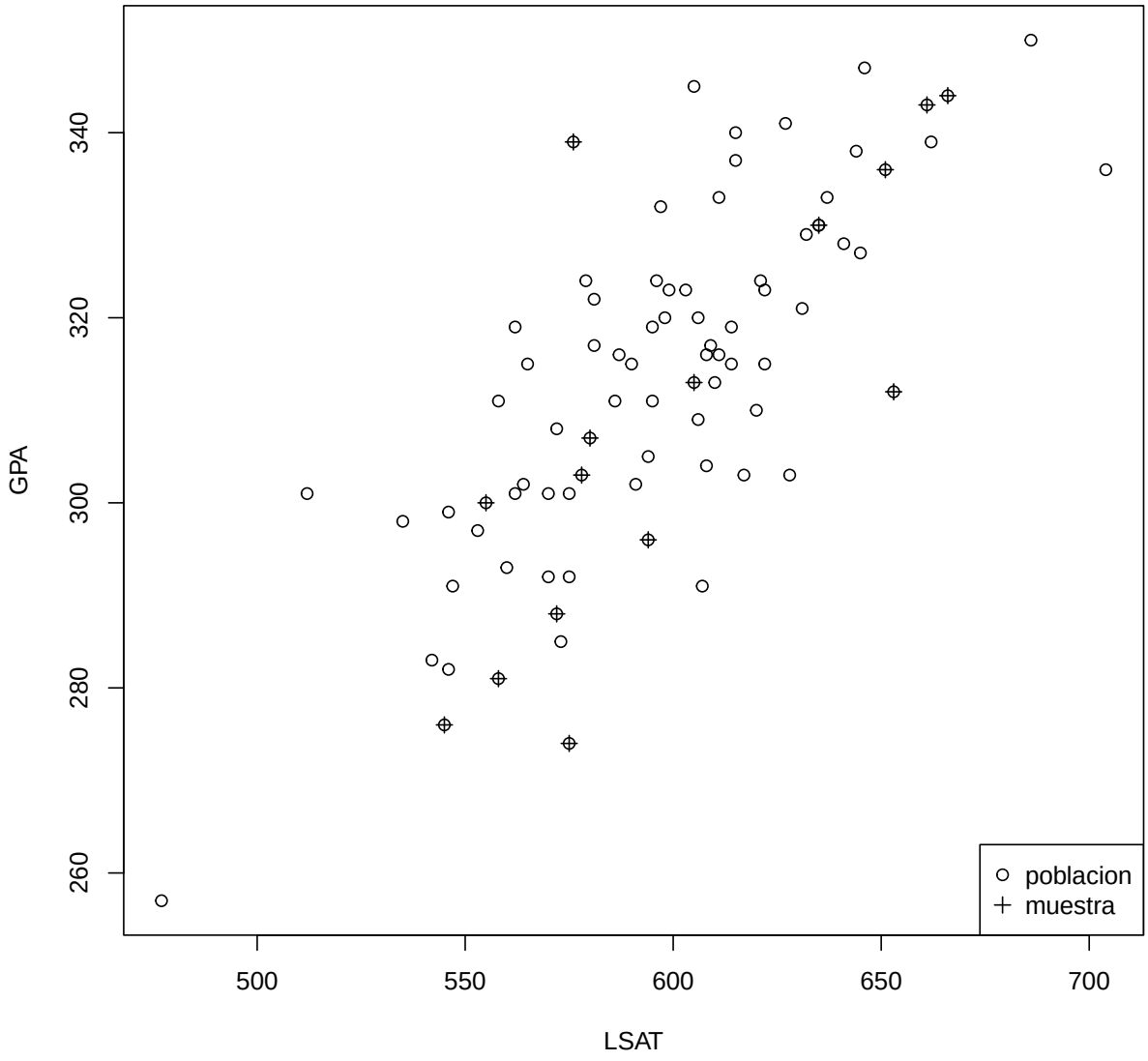
Tema 3. Conceptos relacionados con la Distribución Empírica

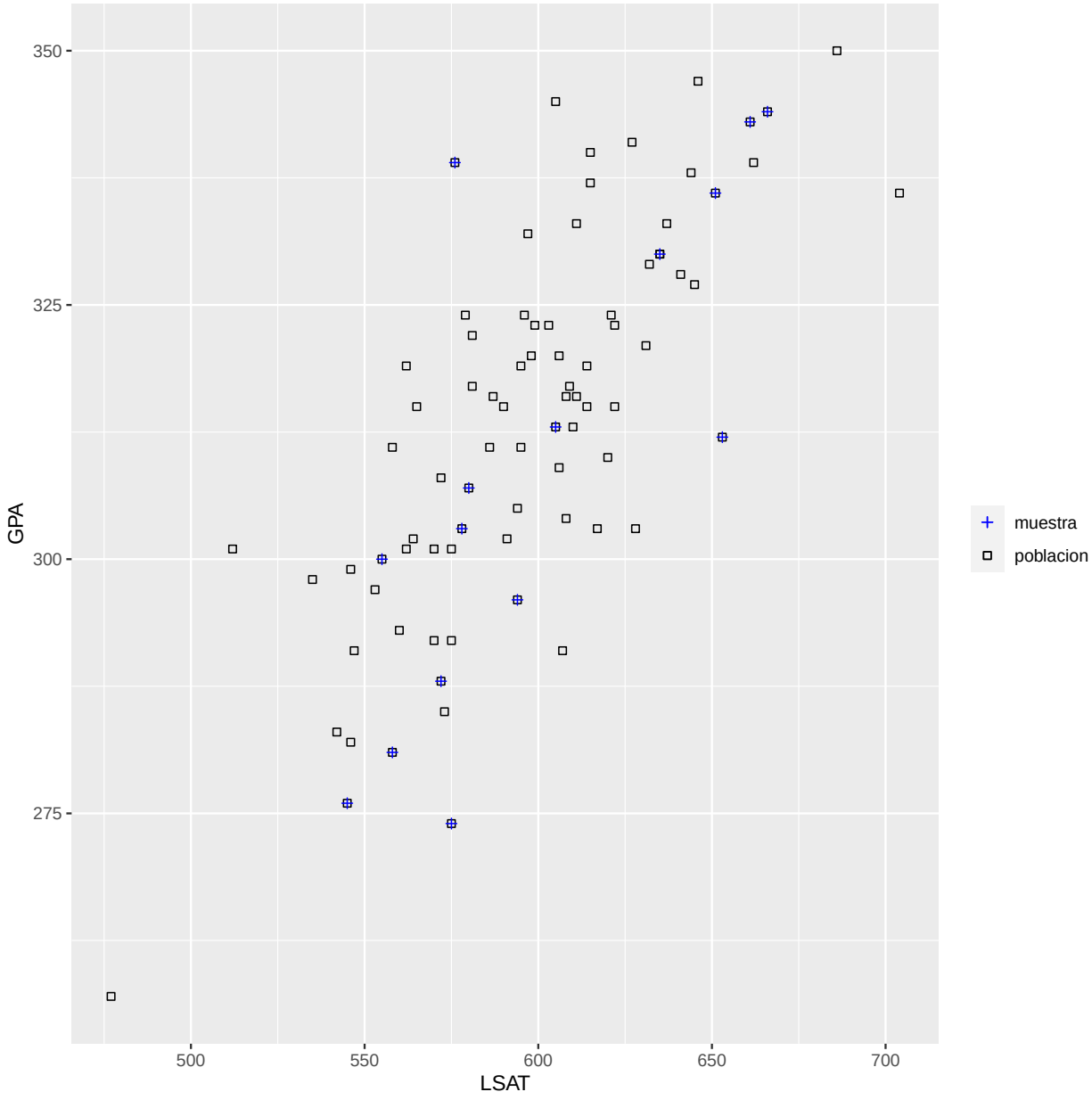
Ejemplo sobre estudios de máster

Se toma el ejemplo de las universidades con máster en leyes que está incluido en el libro de Efron y Tibshirani (1993).

```
library(bootstrap)

with(law82, plot(100*GPA ~ LSAT, ylab="GPA"))
with(law, points(100*GPA ~ LSAT, pch=3))
legend("bottomright", c("poblacion", "muestra"),
pch=c(1,3))
```




```
library(ggplot2)

df1 = data.frame(LSAT = law82$LSAT, GPA = 100*law82$GPA)
df2 = data.frame(LSAT = law$LSAT, GPA = 100*law$GPA)

ggplot() +
    geom_point(data = df1, aes(x = LSAT, y = GPA, color = "poblacion"), shape = 0) +
    geom_point(data = df2, aes(x = LSAT, y = GPA, color = "muestra"), shape = 3) +
    labs(x = "LSAT", y = "GPA") +
    scale_color_manual(name = "", values = c("poblacion" = "black", "muestra" = "blue"),
                       guide = guide_legend(override.aes = list(shape = c(3, 0))))
```



Calculo la correlación entre GPA (la puntuación media en los cursos de grado) y LSAT (calificación de admisión).

La correlación poblacional es

```
with(law82, cor(GPA,LSAT))
```

```
[1] 0.7599979
```

La correlación muestral (estimador *plug-in*) es

```
with(law, cor(GPA,LSAT))
```

```
[1] 0.7763745
```

Función de distribución empírica

```
# Simulo datos de calificaciones

mu = 6.5
sigma = 0.5

y = rnorm(n=20, mean=mu, sd=sigma)
y = round(y,3)
```

La muestra ordenada es

```
sort(y)
```

```
 [1] 5.473 5.780 5.782 5.986 5.993 6.013 6.024 6.151 6.162 6.240 6.341 6.361
[13] 6.438 6.440 6.654 6.694 7.037 7.176 7.203 7.773
```

Alternativamente, en forma tabular:

```
# Instalar la versión development
# devtools::install_github("kupietz/kableExtra")
library(kableExtra)
kable(list(sort(y)[1:10],sort(y)[11:20]), linesep = "", col.names = "datos") %>%
  kable_styling(latex_options = "HOLD_position")
```

datos	datos
5.473	6.341
5.780	6.361
5.782	6.438
5.986	6.440
5.993	6.654
6.013	6.694
6.024	7.037
6.151	7.176
6.162	7.203
6.240	7.773

Algunos valores la función de distribución empírica son

```
Fn = ecdf(y)

Fn(6)
```

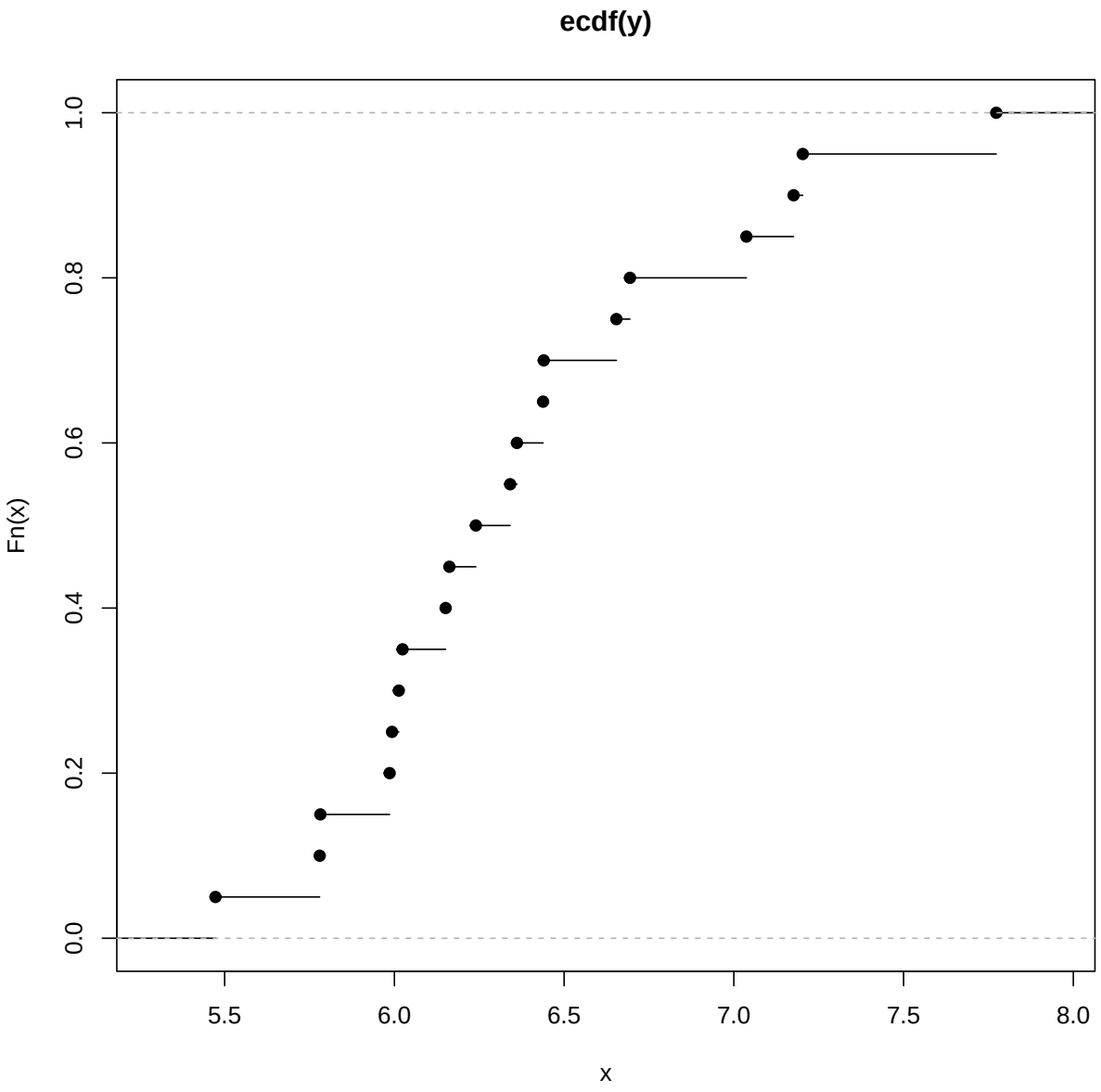
```
[1] 0.25
```

```
Fn(7)
```

```
[1] 0.8
```

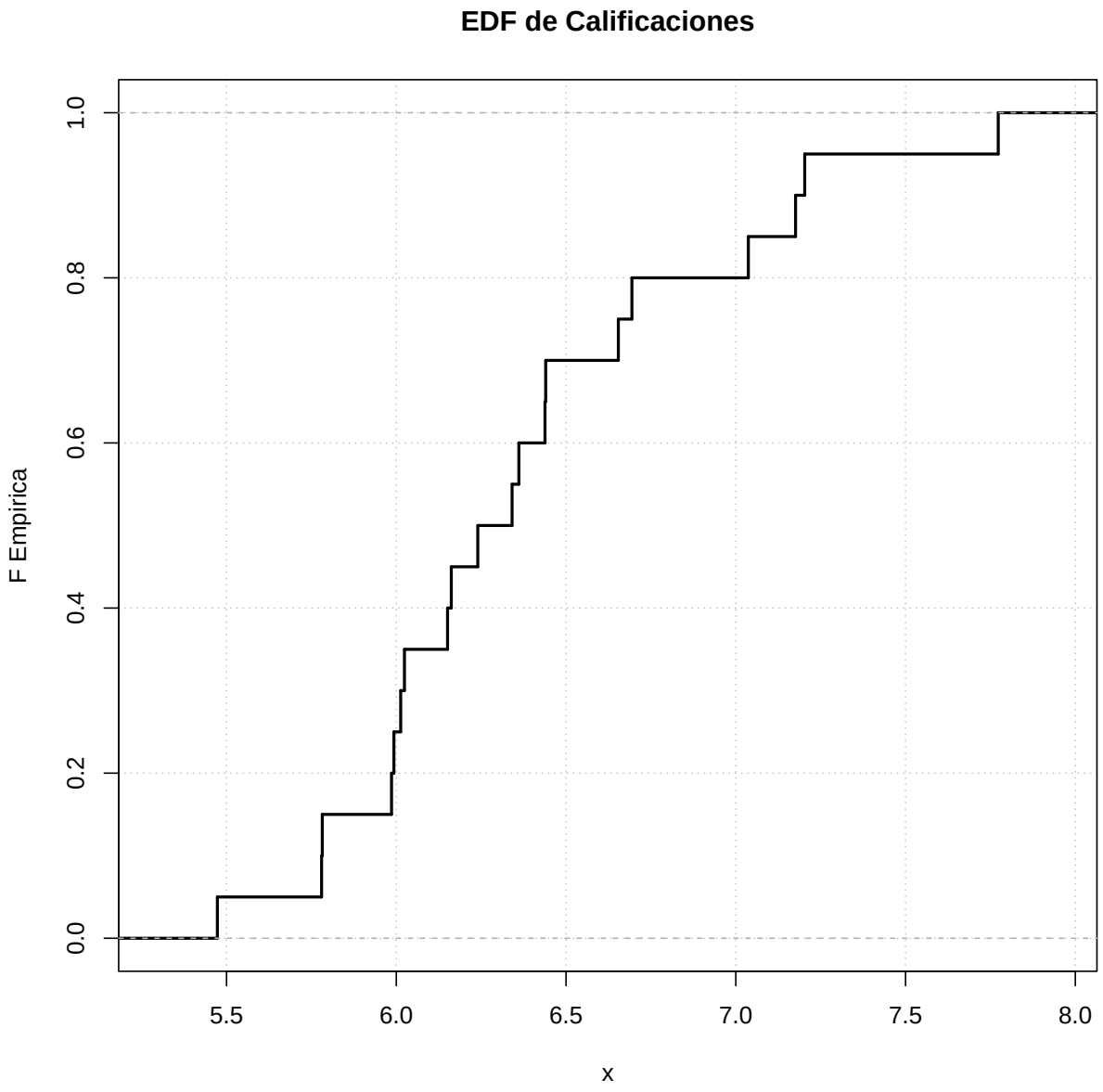
La gráfica de la función de distribución empírica es

```
plot(Fn)
```



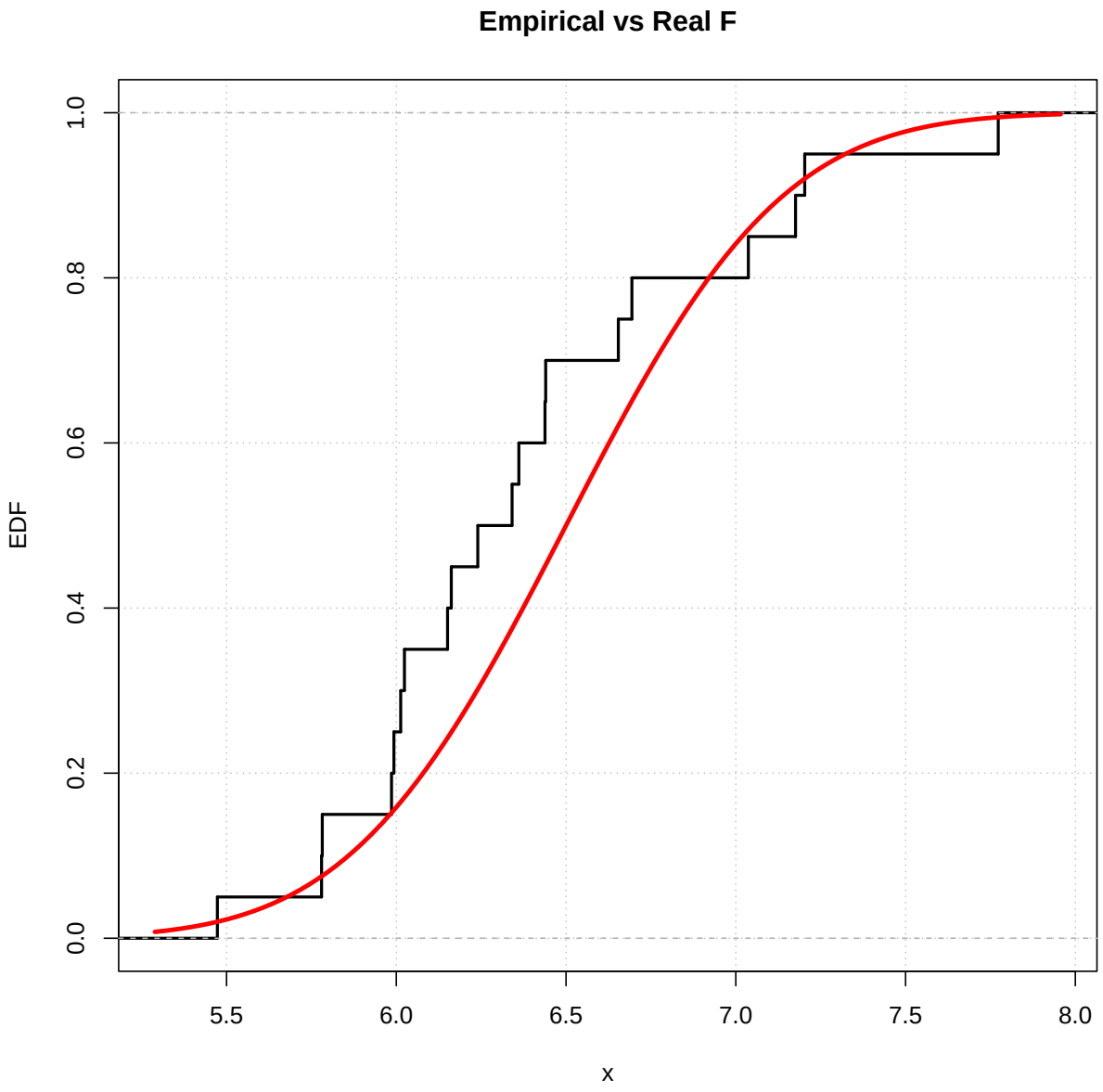
O bien

```
plot.ecdf(x=y, verticals=TRUE, do.p=FALSE,
main="EDF de Calificaciones", lwd=2,
panel.first=grid(col="gray", lty="dotted"),
ylab="F Empirica")
```



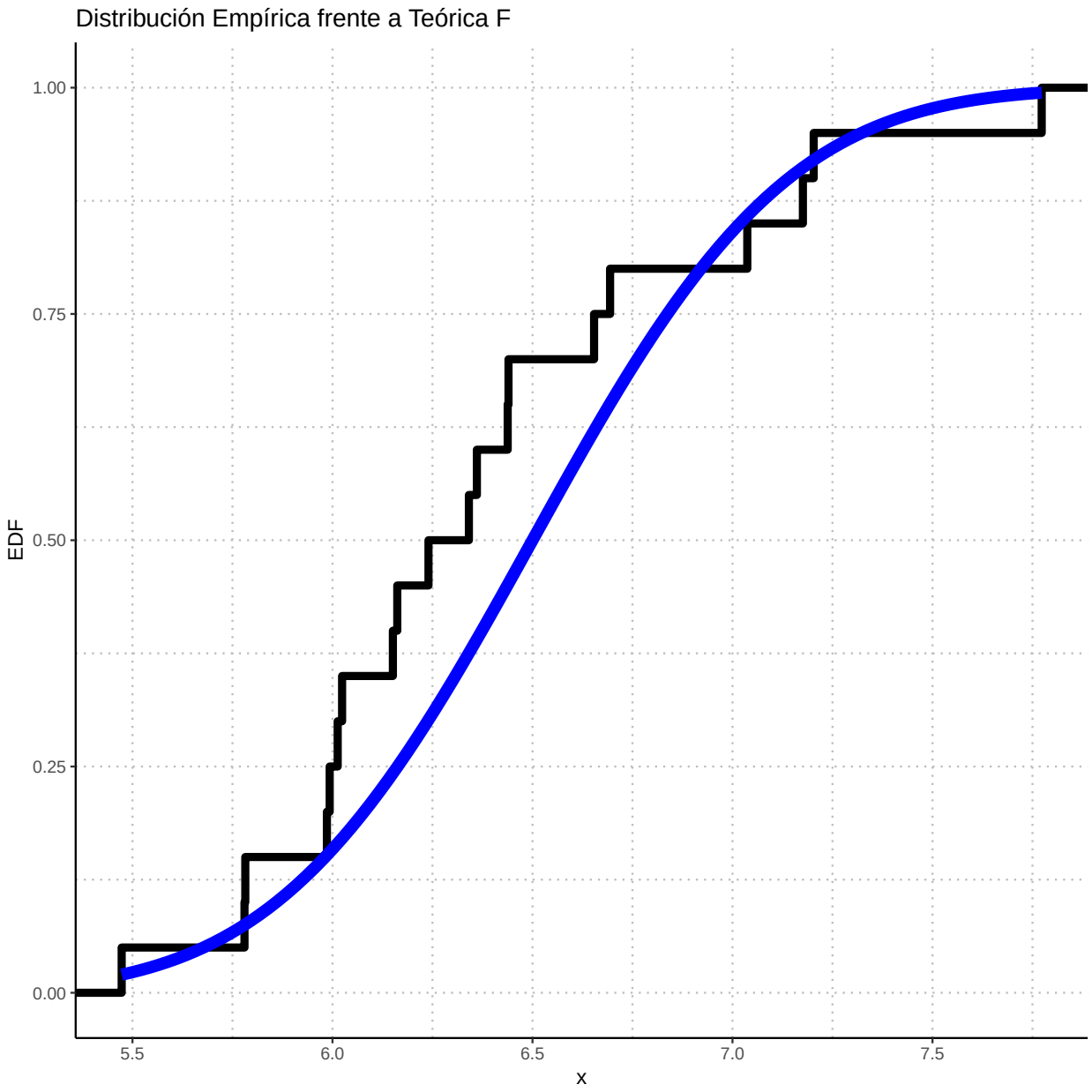
Se puede dibujar la correspondiente función de distribución empírica junto con la curva de la función de distribución real.

```
plot.ecdf(x=y, verticals=TRUE, do.p=FALSE,
main="Empirical vs Real F", lwd=2, xlab="x",
panel.first = grid(nx=NULL, ny=NULL,
col="gray", lty="dotted"), ylab="EDF")

curve(expr=pnorm(x, mean=mu, sd=sigma), col="red",
add=TRUE, lw=3)
```




```r
library(ggplot2)

# Se crea un dataframe
df = data.frame(x = y)

p = ggplot(df, aes(x)) +
  stat_ecdf(geom = "step", color = "black", size = 2) +
  stat_function(fun = pnorm, args = list(mean = mu, sd = sigma),
                color = "blue", size = 3) +
  labs(title = "Distribución Empírica frente a Teórica F", x = "x", y = "EDF") +
  theme_classic() +
  theme(panel.grid.minor = element_line(colour = "gray", linetype = "dotted"),
```

```
        panel.grid.major = element_line(colour = "gray", linetype = "dotted"))

print(p)
```



Simulaciones de la función de distribución empírica

Por ejemplo se toma una m.a.s de una distribución de Poisson

```r
x = rpois(20,3)
P = ecdf(x)
P(3)
```

```
[1] 0.6
```

```r
acumula.dist = function(muestra, z){
    cuento = 0
    for(t in muestra){ if(t<=z) cuento = cuento+1 }
    return(cuento/length(muestra))
}

acumula.dist(x, 3)
```

```
[1] 0.6
```

Para simular de la función de distribución empírica una vez observado vector x, se puede usar la función `sample`.

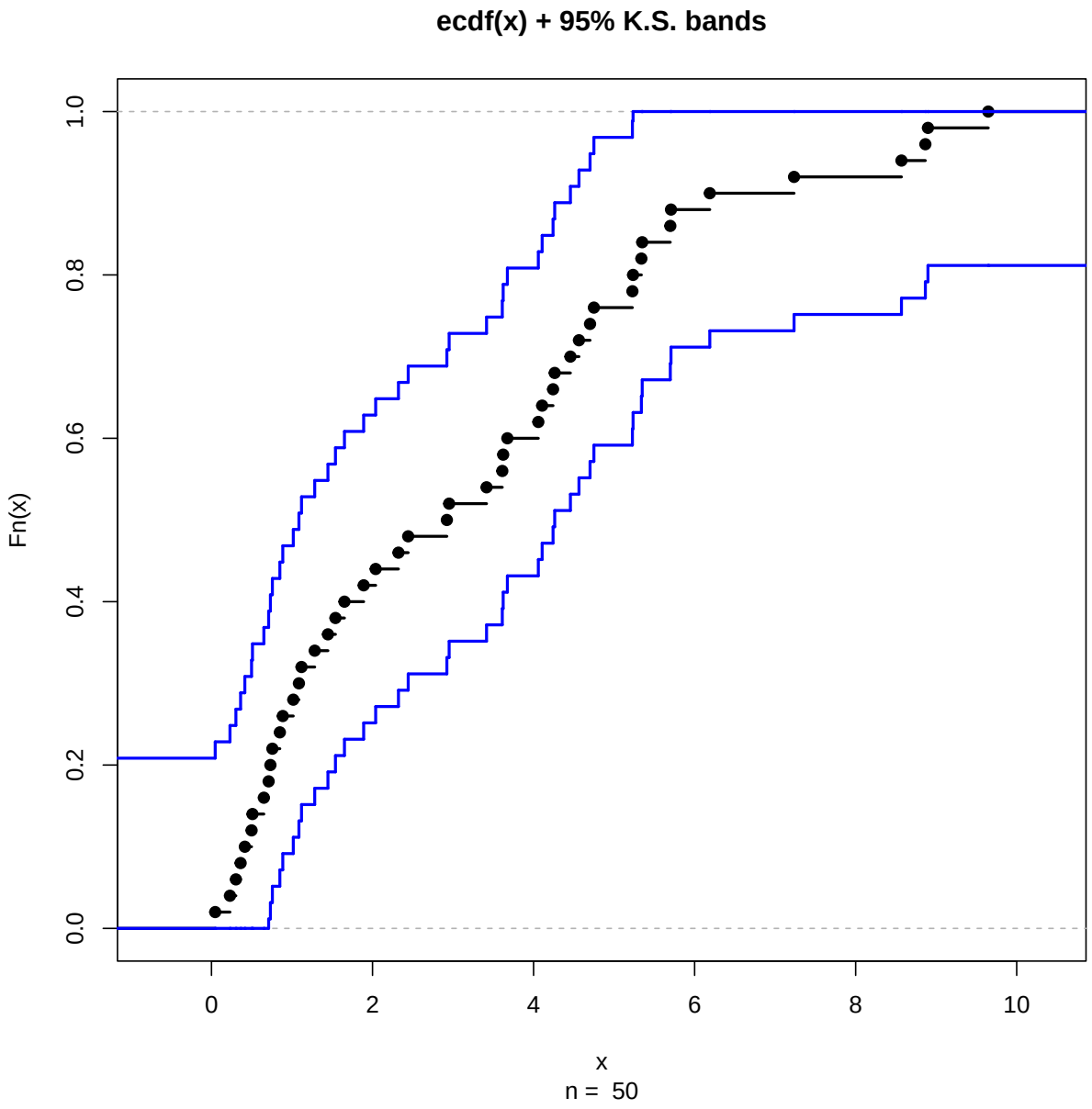
```r
sample(x, size=20, replace=TRUE)
```

```
 [1] 4 3 4 3 3 3 2 2 3 4 2 3 5 3 6 2 3 4 4 2
```

Intervalos de confianza basados en la función de distribución empírica

Simulas datos de una v.a. χ^2 con 3 grados de libertad.

```r
library(sfsmisc)

x = rchisq(50,3)

ecdf.ksCI(x, ci.col="blue", lwd=2)
```



Simulas observaciones de una distribución t de Student

```
datos = rt(20,3)
```

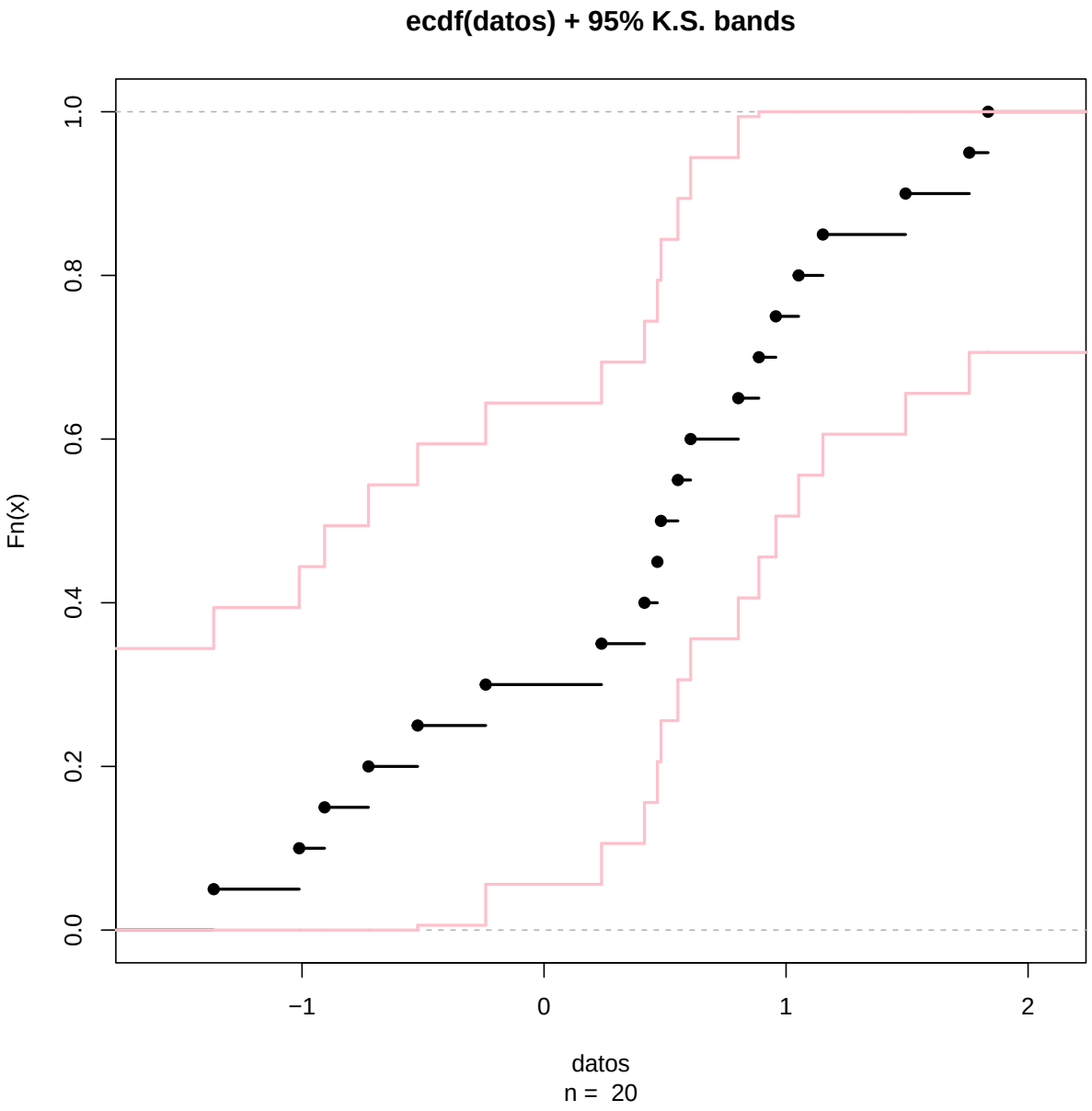
```
dkw_cota = function(datos, x, alfa){
    P = ecdf(datos)
    F_boina = P(x)
    epsilon = sqrt(log(2/alfa)/(2*length(datos)))
```

```
    inf_cota = pmax(F_boina - epsilon, 0)
    sup_cota = pmin(F_boina + epsilon, 1)
    return(c(inf_cota, sup_cota))
}

dkw_cota(datos, -0.5, 0.05)
```

```
[1] 0.0000000 0.5536807
```

Calculas los intervalos

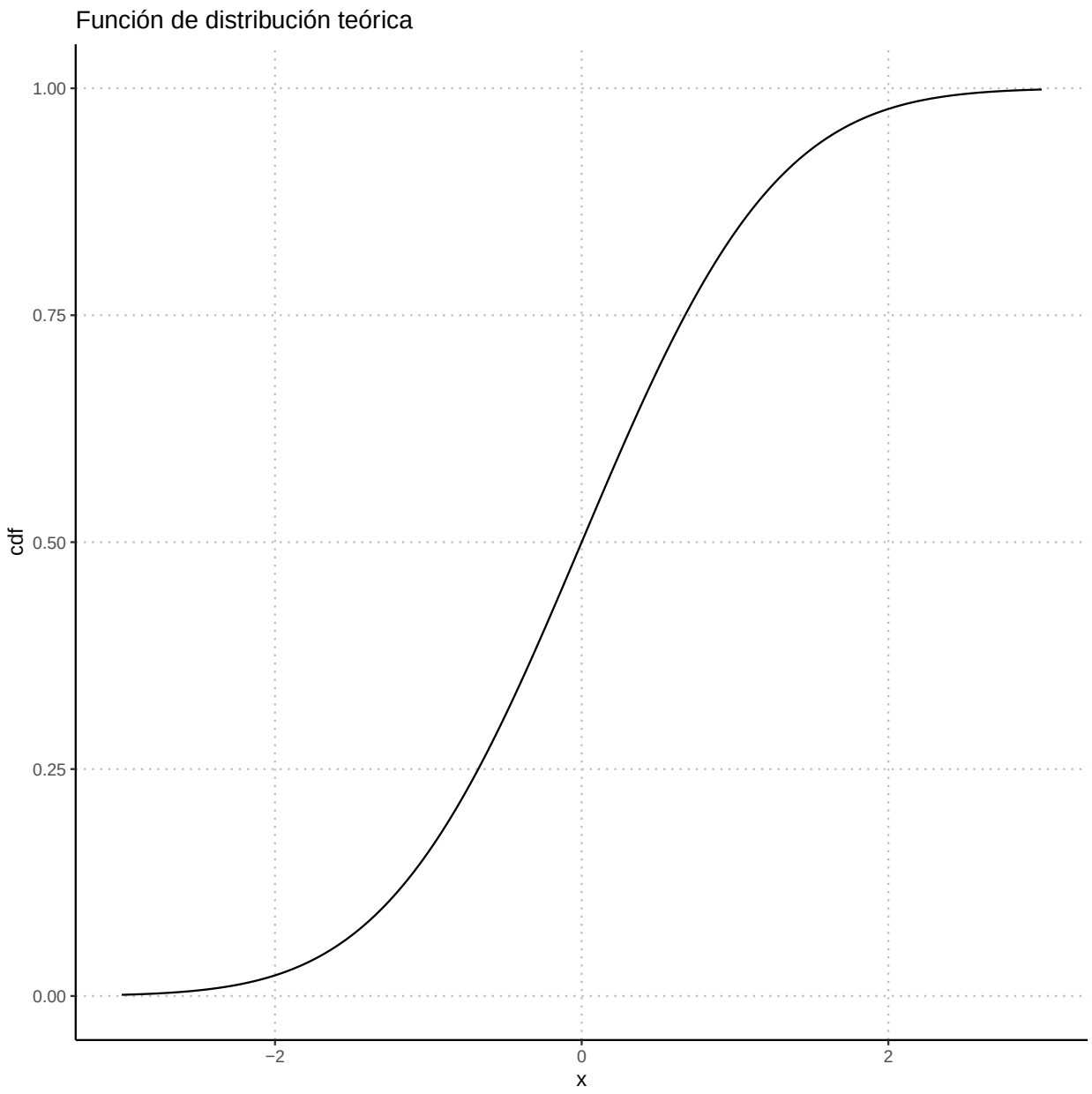
```
ecdf.ksCI(datos, ci.col="pink", lwd=2)
```


ecdf(datos) + 95% K.S. bands
Fn(x)
0.0
0.2
0.4
0.6
0.8
1.0
−1
0
1
2
datos
n = 20

Ejemplo con una distribución normal

En la primera gráfica se muestra la función de distribución teórica de una $N(0, l)$.

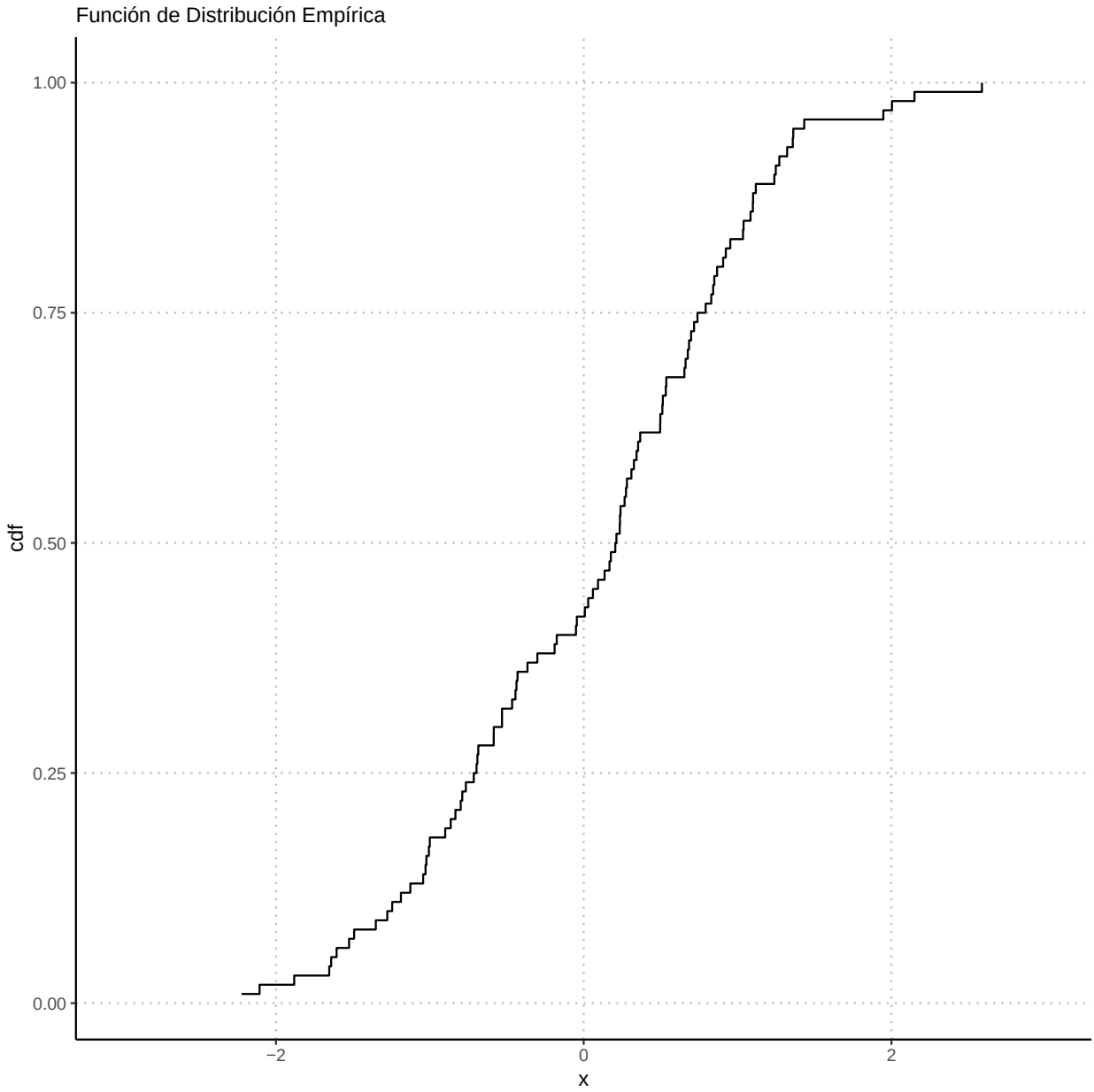
```
grid = seq(-3, 3, length = 1000)
cdf = pnorm(grid)
df_cdf = data.frame(grid, cdf)

n = 100
x = sort(rnorm(n))
cdf.hat = (1:n)/n
df_ecdf = data.frame(x, cdf.hat)

p1 = ggplot(df_cdf, aes(grid, cdf)) +
  geom_line() +
  labs(x = "x", y = "cdf", title = "Función de distribución teórica") +
  theme_classic() +
  theme(panel.grid.major = element_line(colour = "gray", linetype = "dotted"))
print(p1)
```



Se generan 100 observaciones de una $N(0, l)$ y se muestra la función de Distribución Empírica.

```
n = 100
x = sort(rnorm(n))
cdf.hat = (1:n)/n
df_ecdf = data.frame(x, cdf.hat)

p2 = ggplot(df_ecdf, aes(x, cdf.hat)) +
  geom_step() +
  labs(x = "x", y = "cdf", subtitle = "Función de Distribución Empírica") +
  xlim(-3, 3) +
```

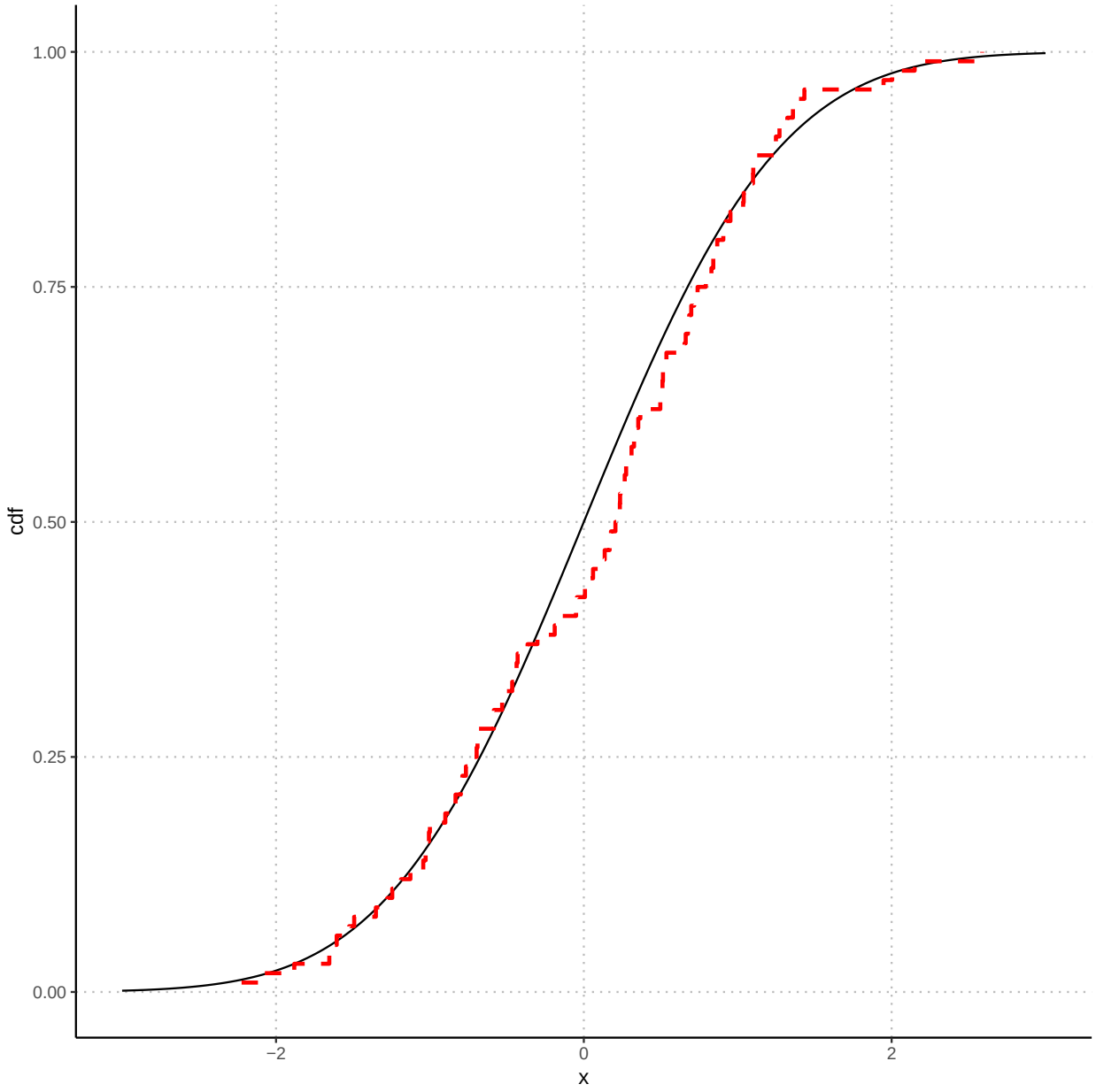
```
  theme_classic() +
  theme(panel.grid.major = element_line(colour = "gray", linetype = "dotted"))
print(p2)
```



Se superponen las dos funciones de distribución.

```
p3 = ggplot() +
geom_line(data = df_cdf, aes(grid, cdf)) +
 geom_step(data = df_ecdf, aes(x, cdf.hat), linetype="dashed", color="red", size=1) +
```

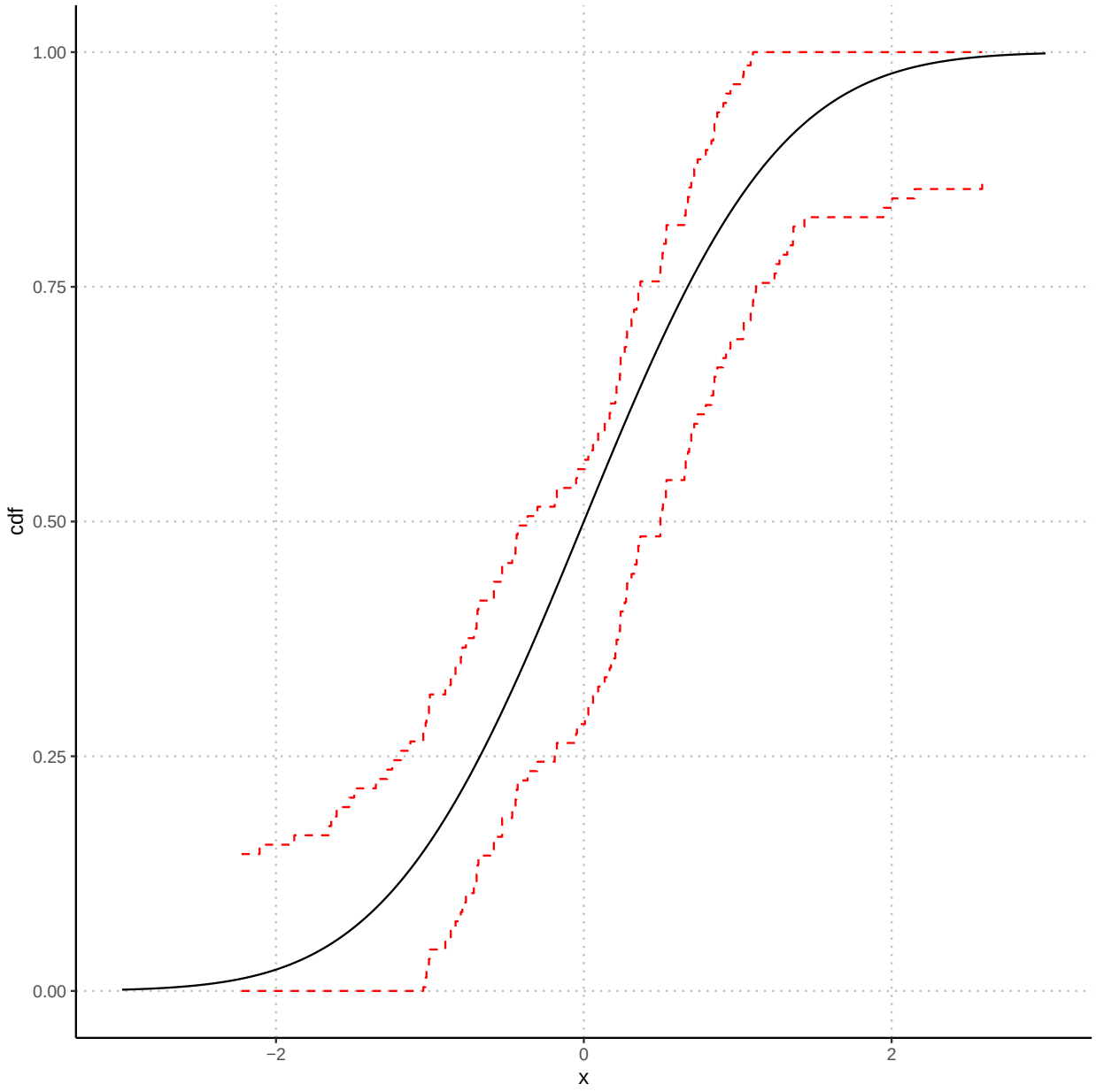
```
 labs(x="x", y="cdf") +
 theme_classic() +
 theme(panel.grid.major = element_line(colour="gray", linetype="dotted"))
print(p3)
```



Se muestran las funciones de distribución verdadera, la empírica y la banda de confianza del 95 por ciento usando 100 observaciones de una $N(0, l)$.

```r
alfa = 0.05
eps = sqrt(log(2/alfa) / (2*n))
l = pmax(cdf.hat - eps, 0)
u = pmin(cdf.hat + eps, 1)
df_band = data.frame(x, l, u)
```

```r
p4 = ggplot(df_cdf, aes(grid, cdf)) +
  geom_line() +
  geom_step(data = df_band, aes(x, l), linetype = "dashed", color = "red") +
  geom_step(data = df_band, aes(x, u), linetype = "dashed", color = "red") +
  labs(x = "x", y = "cdf") +
  theme_classic() +
  theme(panel.grid.major = element_line(colour = "gray", linetype = "dotted"))
print(p4)
```


1.00
0.75
0.50
0.25
0.00
cdf
−2
0
2
x