Estadística descriptiva

3 de marzo de 2020

Introducción

Algunas definiciones y cuestiones

- *Estadística descriptiva*: buscamos describir y resumir un conjunto de datos para estudiar sus propiedades.

Este conjunto de datos surge a partir de una muestra, que es un conjunto de variables aleatorias asociadas a una magnitud que queremos medir. La muestra es tomada de entre toda la población (por lo que puede haber problemas a la hora de hacer un muestreo adecuado), y no necesariamente conocemos la distribución de las variables de interés, ni sus parámetros correspondientes.

- *Variables*: aquella propiedad que medimos (consumo de carne por habitante por año, si una persona fuma o no, sueldo, edad, altura, a quién va a votar): "lo que medimos".
- *Datos*: los valores que se obtienen al estudiar estas variables sobre la muestra: "lo que da la medición". Son una realización de las variables aleatorias de la muestra. Podemos tener datos categóricos o numéricos, y hoy trabajaremos con numéricos.

Cuando trabajamos con datos, estamos trabajando con números dados, no con variables aleatorias. La distribución de estos datos, evaluada por diferentes métodos, puede permitirnos averiguar información de las variables y parámetros poblacionales, con mayor o menor grado de certeza.

- *Métodos resumen*: los usamos para dar un resumen numérico del conjunto de datos y sus diferentes características. Decimos que se trata de parámetros, si son poblacionales, o estadísticos, si son muestrales.
- *Métodos gráficos*: los usamos para visualizar la información asociada a la muestra. En cuanto a métodos gráficos, haremos histogramas, boxplots y qqplots. No haremos diagramas de hoja y tallo.

Los estadísticos que usaremos nos darán una idea de la posición o la centralidad de los datos (media, mediana, percentiles, etc.) y la dispersión de los datos (desvío muestral, IQR, MAD). Hay otros que se pueden usar para estudiar la forma de la distribución, por ejemplo, si las colas de la distribución son pesadas o no tanto (curtosis), y para evaluar asimetría estadística ("skewness").

Los parámetros poblacionales "verdaderos", que son en general desconocidos, se suelen simbolizar con letras griegas. Las variables aleatorias que conforman la muestra suelen ser letras mayúsculas. En cambio, los estadísticos que usemos suelen ser simbolizados con letras minúsculas, y son números, no variables aleatorias.

Notar, por ejemplo, la diferencia entre el parámetro μ, la variable aleatoria promedio $\overline{X_n}$ (que es el promedio de las variables aleatorias de una muestra) y la media $\overline{x}$, que es el número que da el promedio de los datos.

Nuestro problema de hoy

En bioinformática, se emplea una técnica de simulación llamada "docking" para evaluar fácilmente si moléculas de interés se podrían unir o no a proteínas u otras moléculas "blanco". Esto sirve para seleccionar de entre miles de candidatos posibles para el desarrollo de nuevos fármacos, por ejemplo, sólo aquellas moléculas que sean más promisorias.

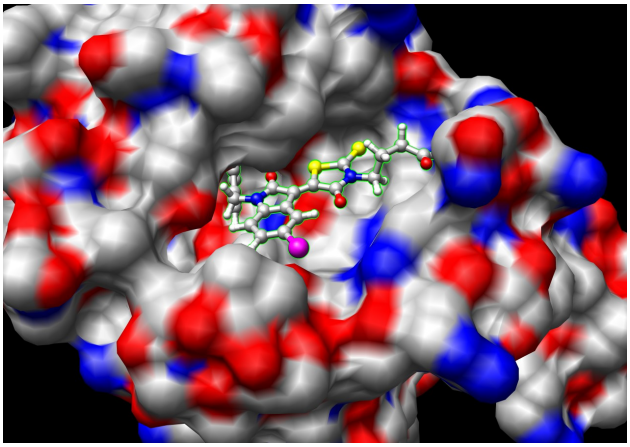


Figure 1: **Simulación computacional de la unión de un compuesto químico a una proteína. Se simula la estructura química del compuesto, la estructura tridimensional de la proteína y las interacciones entre ellos. Fuente: https://commons.wikimedia.org/wiki/File:Docking.jpg**

Ver por ejemplo:

- Introducción al docking (en español)
- Introducción al docking (en inglés)
- Arcon et al, 2017.

Para evaluar esto, se emplean funciones de "scoring", que intentan predecir la afinidad de estas moléculas por una sección blanco en una proteína de interés. Para evaluar si una función de scoring es adecuada o no, se deben calibrar con moléculas de las que ya se conozcan los parámetros adecuados.

En este caso, vamos a hacer esta calibración. Esto es similar a lo que se hace con los métodos de machine learning guiado. Para ello, compararemos tres funciones de scoring, evaluadas sobre una base de datos con 100 moléculas ya previamente ensayadas. Es deseable que:

- en general, para las moléculas de interés, estas funciones tengan un valor que no diste de 1 en más de $\pm 0,1$ (1 representa en este caso una afinidad de una molécula de referencia que es vista como el valor ideal). Los valores mucho menores a 1 indicarían que la molécula se uniría demasiado débilmente, lo que no permitiría que afectara a la proteína de interés. Los valores mucho mayor a 1 indicarían una unión demasiado, por lo que la molécula no se soltaría fácilmente, lo que podría llevar a efectos indeseados en el cuerpo.
- los valores deben tener una distribución aproximadamente normal (que apoyaría la hipótesis de que los valores de scoring sólo tengan un error aleatorio, y no haya ningún sesgo sistemático).

Evaluaremos cuál de las tres funciones cumple mejor con estos requerimientos.

Estimación a partir de datos muestrales

1) *Cargar los datos que se encuentran en el archivo scoring.txt. Estos son los valores de cada una de estas funciones de scoring ensayadas sobre las moléculas del banco de datos de referencia.*

Una molécula se llama "lábil" si el puntaje de acuerdo a una función de scoring calibrada es menor a 0.9 (es decir, son moléculas que se unen demasiado débilmente a la proteína de interés). Estimar la probabilidad de que una molécula sea lábil de acuerdo a la función 1.¿Cuál es el mínimo puntaje alcanzado según la función 1?

En cambio, si el puntaje es mayor a 1.1, se dice que la molécula se "une irreversiblmente" con la proteína. Estimar la proporción de moléculas que se unirían irreversiblemente de acuerdo a la función 2. ¿Cuál es el máximo puntaje alcanzado según la función 2?

Estimar la proporción de moléculas con puntajes entre 0.9 y 1.1 (inclusive) para la función 3. ¿Cuál sería la probabilidad de que tome exactamente el valor 1?

```
score <- read.table("scoring.txt", header = TRUE)
```

La tabla de datos es muy larga. Si en la consola escribimos el nombre de la variable en la que la almacenamos, score, nos va a mostrar una tabla de 100 filas y 3 columnas. De ahora en más, no la llamaremos explícitamente, sólo la usaremos como argumento, y R hará las cuentas que le pidamos automáticamente sin mostrarnos la tabla. Por esto es importante contar con medidas que nos permitan resumir los datos, transmitiéndonos la información esencial que contienen.

Podemos estimar una probabilidad usando los datos muestrales. Contamos los casos favorables y los dividimos sobre los casos totales:

```
# Operadores para comparar: <, <= (menor o igual), >, >= (mayor o igual),
# == (es igual?), != (no igual a).
# Operadores lógicos: ! (no), | (o inclusivo), & (y).

# Casos totales
n <- length(score[,1])
n <- length(score$func1)

# Moléculas lábiles, función 1
proba_1 <- sum(score$func1 < 0.9)/n
proba_1
```

```
## [1] 0.22
```

```
minimo <- min(score$func1)
minimo
```

```
## [1] 0.68
```

```
# Moléculas unidas irreversiblemente, función 2
proba_2 <- sum(score$func2 > 1.1)/n
proba_2
```

```
## [1] 0.25
```

```
maximo <- max(score$func2)
maximo
```

```
## [1] 1.22
```

```r
# Moléculas en el rango 0.95-1.05, función 3
proba_3 <- sum(score$func3 >= 0.9 & score$func3 <= 1.1)/n
proba_3
```

```
## [1] 0.11
```

```r
proba_3_b <- sum(score$func3 == 1.00)/n
proba_3_b
```

```
## [1] 0
```

```r
# Qué da esta última probabilidad? Es razonable el valor?
# Puede ser un problema de la muestra, quizás tuvimos mala suerte
# y no hay ningún valor que dé 1. O puede ser que de verdad dé 0.
# Sin conocer la distribución poblacional, es complicado saberlo.
```

Medidas de centralidad

2) *Calcular, para cada conjunto de datos, la:*

- *media*
- *mediana*
- *media α-podada para $\alpha = 0.1, 0.2$.*

Comparar los valores obtenidos para cada función. ¿Qué diferencias observa? Ir anotando si hay alguna función de scoring que ya podríamos considerar peor que las demás.

La tabla es demasiado extensa para trabajar con todos los datos cómodamente, y por eso usamos medidas resumen. Calculamos las medias:

```
# Mean es media en inglés

media_1 <- mean(score$func1)
media_1
```

```
## [1] 0.9868
```

```
media_2 <- mean(score$func2)
media_2
```

```
## [1] 0.9446
```

```
media_3 <- mean(score$func3)
media_3
```

```
## [1] 0.7729
```

La función colMeans calcula todo de una. También podemos aplicar (apply) las funciones mean y median para obtener la media y la mediana, respectivamente (el argumento 2 indica que aplica la función por columnas).

```
colMeans(score)
```

```
##  func1  func2  func3
## 0.9868 0.9446 0.7729
```

```
mediafunc <- apply(score, 2, FUN = "mean")
mediafunc
```

```
##  func1  func2  func3
## 0.9868 0.9446 0.7729
```

```
medianafunc <- apply(score, 2, FUN = "median")
medianafunc
```

```
## func1 func2 func3
## 0.990 0.945 0.730
```

La media α-podada quita el $100\alpha\%$ de los valores a cada extremo. Hacemos la media α-podada para $\alpha = 0, 1$, el caso 0,2 queda para hacer en casa.

```
# Definimos un vector de 3 ceros que almacenará los valores
# de media 0.1-podada
media_01 <- rep(0,3)
```

```r
# Calculamos la media 0.1-podada y la guardamos en el vector
# agregando el parámetro alfa como argumento del comando mean
media_1_01 <- mean(score$func1, 0.1)
media_1_01
```

```
## [1] 0.98525
```

```r
media_01[1] <- media_1_01

media_01[2] <- mean(score$func2, 0.1)
media_01[2]
```

```
## [1] 0.945
```

```r
media_01[3] <- mean(score$func3, 0.1)
media_01[3]
```

```
## [1] 0.753
```

Comparemos los valores de estas medidas de centralidad. ¿Qué podemos observar?

```r
# Armamos un data frame con los vectores de media, mediana
# y la media 0.1-podada.

centralidadfunc <- data.frame(
                          Media = c(mediafunc),
                          Mediana = c(medianafunc),
                          Podada01 = c(media_01))
centralidadfunc
```

```
##        Media Mediana Podada01
## func1 0.9868   0.990  0.98525
## func2 0.9446   0.945  0.94500
## func3 0.7729   0.730  0.75300
```

La media es sensible frente a outliers (también llamados valores atípicos), mientras que la mediana y las medias podadas no lo son tanto (son medidas más robustas). Esto quiere decir que si hay un valor extremo (muy bajo o muy alto), el valor de la media cambia mucho, mientras que el de las demás no mucho. Eso se debe a que la mediana no se fija explícitamente en los valores de los datos, si no, si se acumulan en la mitad de la derecha o de la izquierda. Y para calcular la media α-podada cortamos los valores de las puntas, que serían los posibles valores atípicos.

Conclusiones (2)

Probablemente, los datos para la función 3 sean asimétricos, dado que la mediana es menor a la media (esto se llama asimetría positiva o a derecha). En los demás casos, media y mediana son bastante similares, indicando que probablemente los datos estén distribuidos en forma simétrica.

En los casos de las funciones 1 y 2 la media 0.1-podada es prácticamente igual a la media, mientras que en el caso 3 varía un poco más. Eso podría indicar la presencia de outliers en los datos asociados a la función 3.

Medidas de posición: percentiles, cuantiles

3) *Obtener los percentiles 10, 25, 50, 75 y 90 y los valores máximos y mínimos, para cada una de las funciones de scoring. Comparar los valores obtenidos.*

En general, $x_{P\%}$ es el valor tal que el P % de los datos medidos son menores a $x_{P\%}$ (o tal que $100 - P$ % son mayores a él). Percentiles famosos:

$x_{50\%}$ es la mediana.

$x_{25\%} \rightarrow Q_1$, primer cuartil.

$x_{75\%} \rightarrow Q_3$, tercer cuartil.

Si en en vez de indicar el percentil con un porcentaje lo hacemos con un número entre 0 y 1, hablamos de cuantiles. Es decir, el cuantil 0,20 es el percentil 20 %.

El comando a usar en R es quantile, indicando el conjunto de datos y qué cuantiles le pedimos evaluar.

```
quantile(score$func1, c(0.1,0.25,0.5,0.75,0.9))
```

```
##    10%    25%    50%    75%    90%
## 0.8600 0.9075 0.9900 1.0525 1.1110
```

Una forma rápida de calcular algunos de estos valores es pedirle a R que haga un summary (resumen) de los datos:

```
# Summary de un conjunto de datos
summary(score$func1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.6800  0.9075  0.9900  0.9868  1.0525  1.2100
```

```
# Pedimos el summary para toda la tabla, lo hace por columnas
summary(score)
```

```
##      func1            func2            func3
##  Min.   :0.6800   Min.   :0.6800   Min.   :0.6800
##  1st Qu.:0.9075   1st Qu.:0.7900   1st Qu.:0.7000
##  Median :0.9900   Median :0.9450   Median :0.7300
##  Mean   :0.9868   Mean   :0.9446   Mean   :0.7729
##  3rd Qu.:1.0525   3rd Qu.:1.1025   3rd Qu.:0.8100
##  Max.   :1.2100   Max.   :1.2200   Max.   :1.2100
```

Los valores informados por el summary son el mínimo, Q_1, la mediana, la media, Q_3 y el máximo. Sirven para tener a primera vista información sobre el rango y la centralidad de los datos, es recomendable correrlo cuando se empieza un análisis exploratorio de datos.

Conclusiones (3)

Los valores mínimos y máximos son prácticamente los mismos para las tres funciones. Comparando los valores de los cuartiles, se observa que los datos de la función 1 y la 3 están menos dispersos que los de la 2. Los datos de la función 3 parecen estar más concentrados en valores menores.

Como ya dijimos antes, media y mediana son similares para 1 y 2, pero los datos de la función 3 tienen una mediana menor a la media (señal de asimetría). En 1 y en 2 la mediana está aproximadamente equidistante de los cuartiles, mientras que en 3 está más cerca del primer cuartil. Esto también indicaría una posible asimetría.

Medidas de dispersión

4) *Calcular medidas de dispersión para estos tres conjuntos de datos:*

- *desvío estándar,*
- *rango intercuartil o intercuartílico (IQR),*
- *MAD (mediana de la desviación absoluta).*

Comparar los valores de dispersión obtenidos. ¿Cuál de las funciones parece tener valores menos dispersos?

Algunas definiciones:

La varianza muestral (insesgada) se calcula como:

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2$$

El n-1 va a ser explicado en el futuro, pero tiene que ver con que estamos trabajando con el promedio muestral y no la verdadera media poblacional. Restarle al número de datos que usamos apunta a corregir ese error.

El desvío estandar muestral es $s = \sqrt{s^2}$. Observemos que s tiene las mismas unidades que los datos (a diferencia de la varianza, que tendría las unidades elevadas al cuadrado). Tanto s como s^2 son sensibles a outliers.

El rango intercuartil, IQR, se define como $Q_3 - Q_1$, donde Q_1 y Q_3 son el primer y tercer cuartil, respectivamente.

La MAD, mediana de las desviaciones absolutas, es

$$\text{mediana}|x_i - \tilde{x}|$$

. En espíritu es similar a la cuenta que se hace para calcular la varianza muestral, dado que también sumamos las distancias de los datos a un valor central (la mediana en este caso), y a eso lo promediamos (medianizamos en este caso). Después, a la varianza hay que tomarle la raíz para que quede con la misma dimensión que los datos, y trabajar con el desvío (mientras que la MAD ya tiene las mismas unidades).

Tanto el IQR como la MAD son medidas de dispersión más robustas frente a outliers que el desvío estándar, dado que para el IQR no estamos tomando en cuenta explícitamente los valores de los extremos, y para la MAD trabajamos con mediana y no con media.

```
sd(score$func1)
```

```
## [1] 0.1018404
```

```
IQR(score$func1)
```

```
## [1] 0.145
```

```
# Da lo mismo que calcularlo a mano:
quantile(score$func1,0.75)-quantile(score$func1,0.25)
```

```
##   75%
## 0.145
```

```
mad(score$func1)
```

```
## [1] 0.111195
```

```r
#Calculamos sd, IQR y MAD usando el comando apply por columnas sobre los datos:

apply(score, 2, "sd")
```

```
##     func1     func2     func3
## 0.1018404 0.1674552 0.1058329
```

```r
apply(score, 2, "IQR")
```

```
##  func1  func2  func3
## 0.1450 0.3125 0.1100
```

```r
apply(score, 2, "mad")
```

```
##    func1    func2    func3
## 0.111195 0.229803 0.059304
```

```r
# Lo copiamos en un data frame para visualizarlo mejor
# round(..., 3) es para redondear a tres cifras decimales

dispersionfunc <- data.frame(
                      SD = c(round(apply(score, 2, "sd"),3)),
                      IQR = c(round(apply(score, 2, "IQR"),3)),
                      MAD = c(round(apply(score, 2, "mad"),3)))

dispersionfunc
```

```
##          SD   IQR   MAD
## func1 0.102 0.145 0.111
## func2 0.167 0.312 0.230
## func3 0.106 0.110 0.059
```

Conclusiones (4)

Comparamos una medida de dispersión para los tres conjuntos de datos. Las funciones 1 y 3 parecen tener la menor dispersión de datos (dependiendo de qué medida usemos para comparar). El desvío estándar para ambas es similar, y está en torno a lo que buscamos (queríamos que los valores de las funciones se hallarán en el rango $1 \pm 0,1$). Los datos de 2 son siempre los más dispersos.

Podemos hacer algunas observaciones entre las diferentes medidas de dispersión. El IQR indica el intervalo donde se halla el 50 % central de los datos (de 25 % a 75 %), mientras que el desvío estándar incluye la información de todos los datos.

En los tres casos el IQR aumenta respecto al desvío (esto pasa en general, el IQR suele ser mayor), pero el aumento en 2 es mucho mayor que para las funciones 1 y 3. Eso podría indicar que los datos centrales de 2 están mucho más dispersos que los de 1 y 3. Pero como los valores de desvío no son tan distintos, probablemente no haya valores tan extremos para 2.

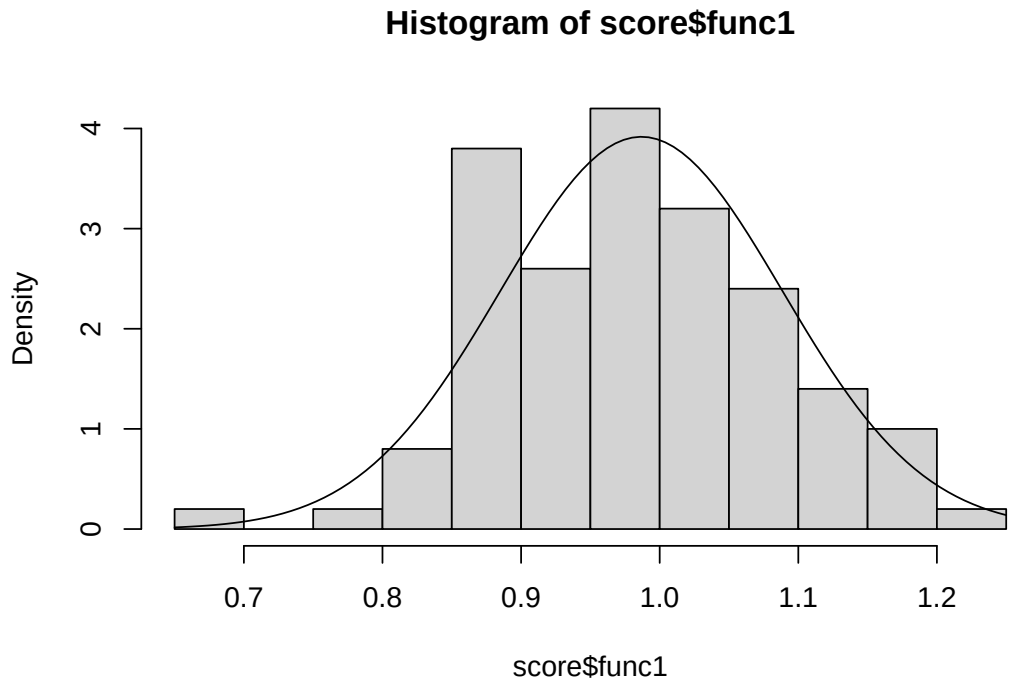
La MAD de 3 disminuye mucho respecto al desvío estándar de los datos de la función 3 (mientras que para 1 y 2 cambia poco o aumenta). Eso apoyaría lo que ya sospechábamos: los datos de 3 presentan outliers.

Métodos gráficos: Histogramas

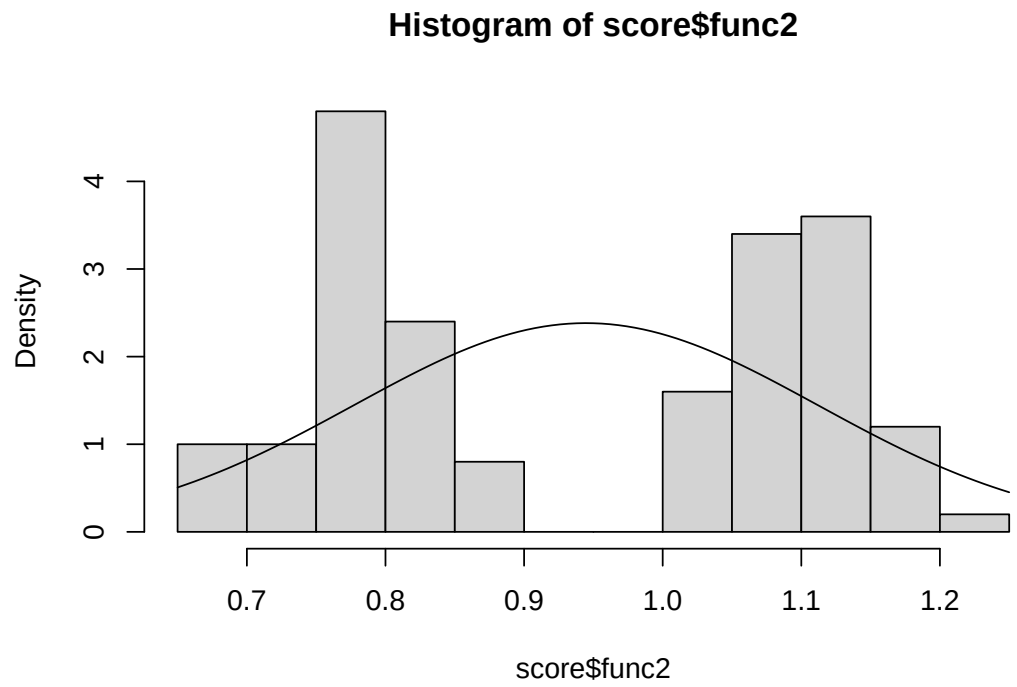
5) *Construir histogramas que permitan visualizar los valores de scoring para cada función. ¿Qué observaciones haría sobre la distribución de estos valores?. ¿Alguna de ellas parece bimodal? ¿En alguna de ellas parece haber valores atípicos o outliers?*

¿Los valores de scoring se hallan en el rango deseado? ¿Hay alguna asimetría en la distribución de los valores de una función? ¿En algún caso el ajuste normal parece razonable? Verificarlo superponiendo una curva de densidad normal con los parámetros correspondientes.

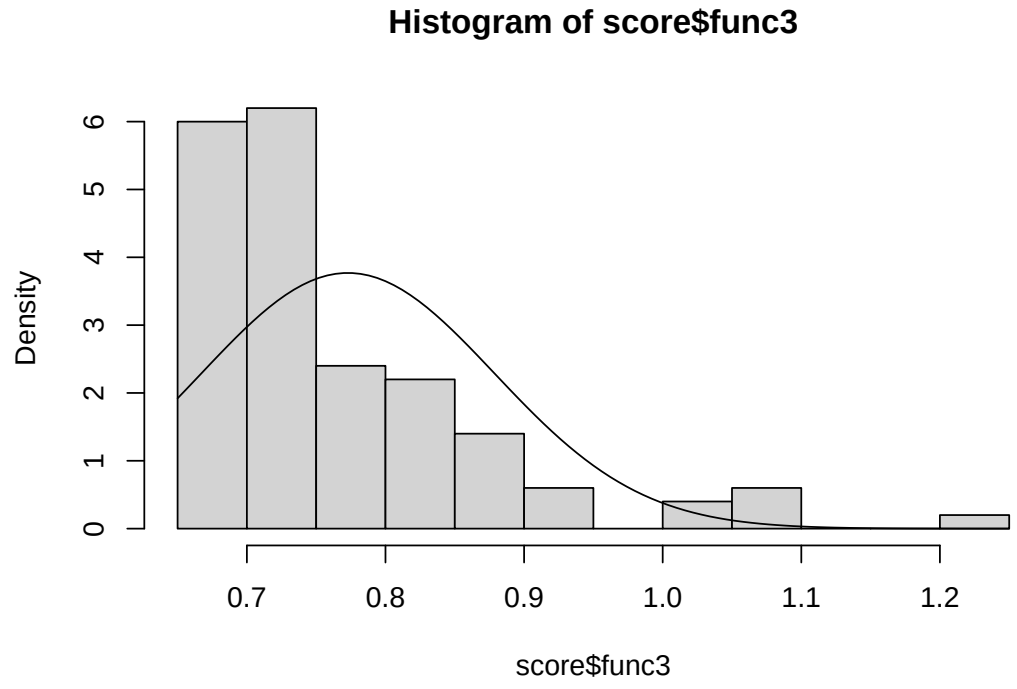
```
# Calculamos medias y desvíos para representar las curvas normales.

media_1<-mean(score$func1);desvio_1<-sd(score$func1)
media_2<-mean(score$func2);desvio_2<-sd(score$func2)
media_3<-mean(score$func3);desvio_3<-sd(score$func3)

# Hacemos los histogramas con el comando hist
# (prob = TRUE es para normalizar los histogramas, si
# fuera FALSE los haría en forma absoluta)
# y superponemos curvas de densidad normales

hist(score$func1, prob=TRUE)
curve(dnorm(x, mean = media_1, sd= desvio_1), add=TRUE)
```




```
hist(score$func2, prob=TRUE)
curve(dnorm(x, mean = media_2, sd= desvio_2), add=TRUE)
```




```
hist(score$func3, prob=TRUE)
curve(dnorm(x, mean = media_3, sd= desvio_3), add=TRUE)
```



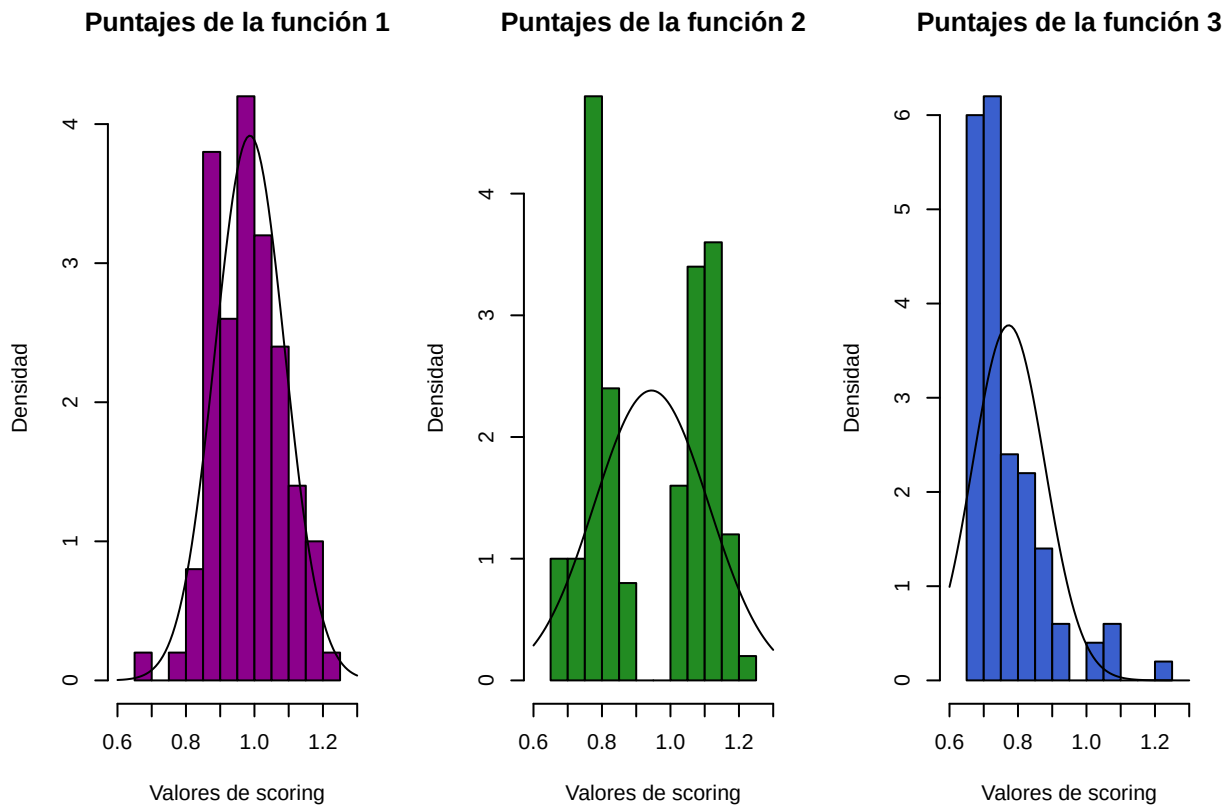
Conclusiones (5)

- La moda de un conjunto de datos es el valor que más veces se repite. Es otra medida de centralidad, que en general se usa para datos categóricos. Se dice que un conjunto de datos es unimodal si están concentrados alrededor de un valor (si pensamos que los histogramas son como montañas, sería que se observa un sólo pico). Los conjuntos de datos 1 y 3 son unimodales. En cambio, los datos de la función 2 serían bimodales (hay dos picos). Eso hasta ahora no lo habíamos podido observar.

- Los datos de la función 3 son asimétricos (esto ya lo habíamos observado cuando hablábamos de media y mediana, y de los cuantiles), y parece haber varios outliers. La cola derecha de los datos es bastante larga (o sea, aparecen muchos datos a derecha, alejados del centro). Pueden ser outliers o no (dependiendo del valor exacto de SD o IQR y la media o la mediana decidiremos esto; con los boxplots veremos una forma de saberlo), pero casi seguramente que el valor que aparece por arriba de 1.1 lo es.
- Los datos de la función 1 son los que mejor ajustan a una distribución normal.

Si queremos todos los gráficos juntos

```
par(mfrow=c(1,3))

hist(score$func1,
     main = "Puntajes de la función 1",
     xlab = "Valores de scoring",
     ylab = "Densidad",
     col="darkmagenta",
     xlim = c(0.6,1.3),
     prob=TRUE)
curve(dnorm(x, mean = media_1, sd= desvio_1), add=TRUE)

hist(score$func2, main = "Puntajes de la función 2",
     xlab = "Valores de scoring",
     ylab = "Densidad",
     col="forestgreen",
     xlim = c(0.6,1.3),
     prob=TRUE)
curve(dnorm(x, mean = media_2, sd= desvio_2), add=TRUE)

hist(score$func3, main = "Puntajes de la función 3",
     xlab = "Valores de scoring",
     ylab = "Densidad",
     col="royalblue3",
     xlim = c(0.6,1.3),
     prob=TRUE)
curve(dnorm(x, mean = media_3, sd= desvio_3), add=TRUE)
```



El comando par(...) permite ajustar las opciones de la ventana de visualización de los gráficos que hace R. par(mfrow = c(1,3)) le dice a R que queremos armar una grilla donde a medida que se van armando los gráficos los va ubicando en los huecos de la grilla. En este caso, la grilla es de 1 fila y 3 columnas (por eso el (1,3)).

Además, le pusimos algunas opciones extras a los gráficos. main es el título. xlab e ylab son las etiquetas de los ejes x e y (lab es de label, etiqueta en inglés). col es el color de las barras, y xlim es para indicar los bordes máximo y mínimo del eje x (así todos muestran la misma región y es más fácil comparar). Estos argumentos son completamente opcionales, y se hay más opciones disponibles. En general, para averiguar las opciones de los comandos y cómo usarlos, es recomendable buscar en Google "nombre del comando" + opciones/ayuda (o en inglés, "command" + options/help).

Métodos gráficos: Box-plots

6) *Graficar los box-plots correspondientes. ¿Cómo se compara la información que dan estos gráficos con la obtenida con los histogramas? En base a los gráficos obtenidos, discutir simetría, presencia de outliers y comparar dispersiones.*

Figure 2: **Mi diagrama es una caja!**

Un box-plot (o gráfico/diagrama de caja) es un gráfico que rápidamente nos permite representar conjuntos de datos a partir de la información brindada por sus cuartiles. Se pueden hacer indistintamente en forma vertical u horizontal (R por defecto los hace verticalmente), y sirven para comparar rápidamente las distribuciones de diferentes series de datos.

Se comienza dibujando una caja, donde se toman como borde izquierdo (o inferior) el valor del primer cuartil, Q_1, y como borde derecho (o superior), el tercer cuartil, Q_3. Una línea en el medio de la caja representa la mediana. Dentro de la caja se hallarán el 50 % central de los datos. La posición de la mediana respecto de los cuartiles puede indicar la (a)simetría de los datos (si está equidistante serían simétricos, si se acerca a un cuartil, se estarían acumulando más datos de ese lado).

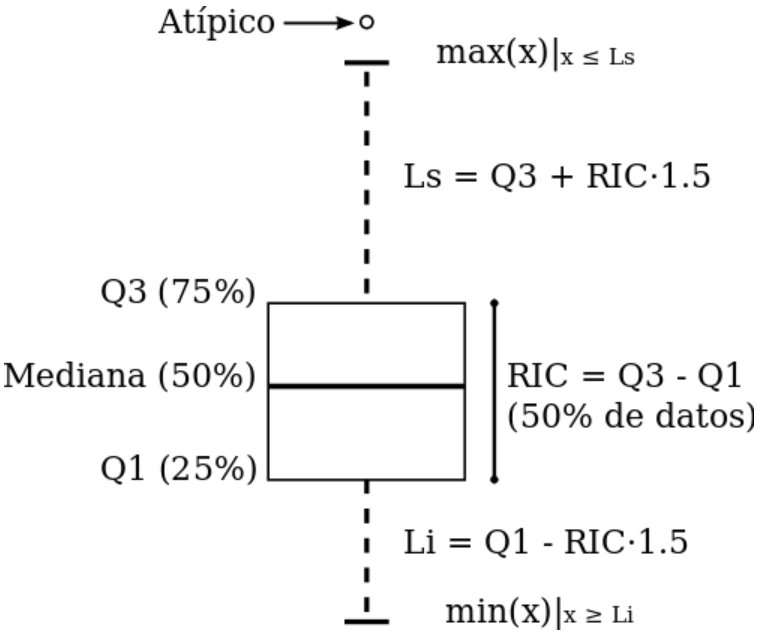
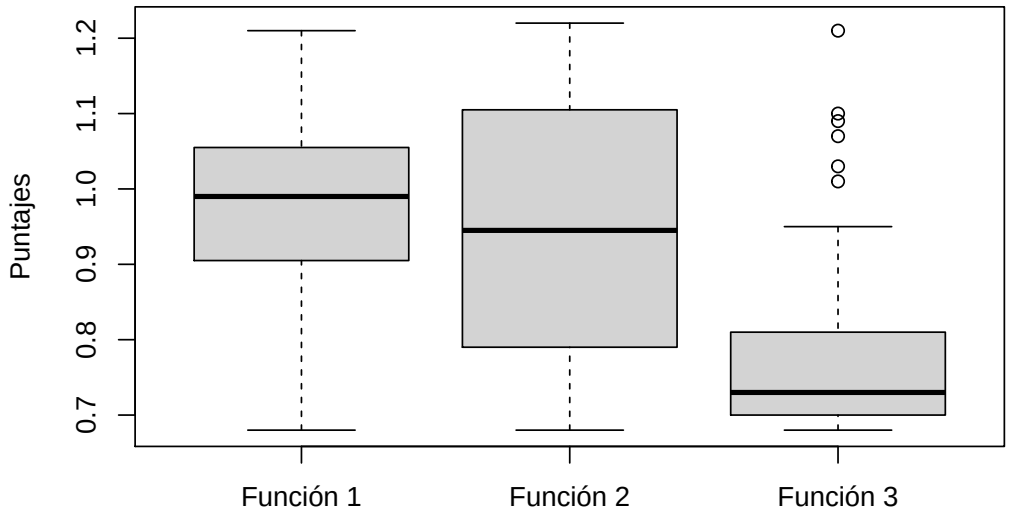


Figure 3: Elementos de un boxplot (Wikipedia). Li y Ls son los bigotes inferior y superior, respectivamente. RIC es el rango intercuartil (que nosotros llamamos IQR).

Por fuera de la caja (que tendrá longitud igual a IQR, el rango intercuartil) se dibujan los "bigotes" que abarcan los demás datos: tienen longitud menor o igual a $1,5 \times IQR$ (en general, a veces se usan longitudes diferentes. R usa este criterio), y se extienden a ambos lados. Si se alcanzan el máximo o el mínimo de los datos antes de $1,5 \times IQR$, se cortan ahí los bigotes. Si no, alcanzan la longitud máxima. Si todavía hay valores por fuera de ese rango (más chicos o más grandes), se los representa por fuera de los bigotes con un símbolo especial (círculo, cruces, asteriscos), y se los llama valores atípicos o "outliers".

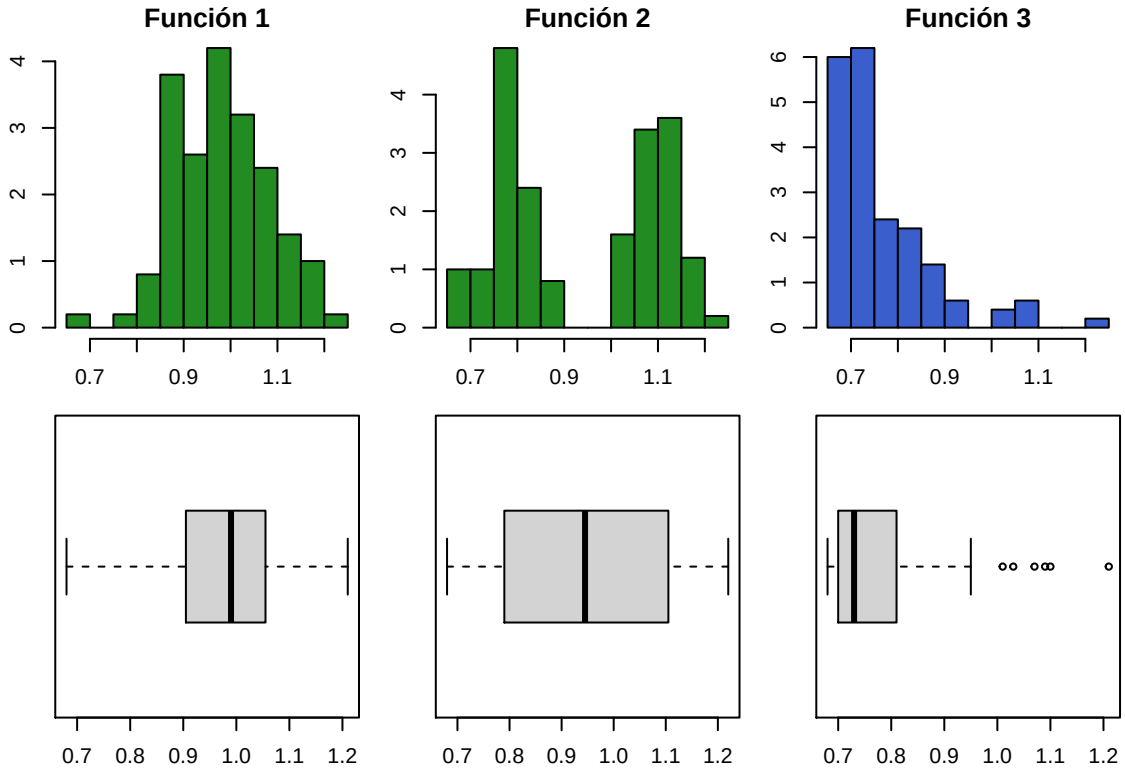
```
#Se puede hacer de dos formas:
#with(data=score, boxplot(func1, func2, func3))

#O así, y le ponemos los nombres

boxplot(score$func1, score$func2, score$func3,
        ylab = "Puntajes",
        names=c("Función 1","Función 2", "Función 3"))
```



Comparando las longitudes de las cajas se puede visualizar la dispersión de los datos, mientras que la posición de la mediana (y de la caja) da una indicación de centralidad. La posición de la mediana dentro de la caja muestra la asimetría en la distribución (si no está centrada). Esto también puede observarse en algunos casos con la longitud relativa de los bigotes y la presencia de outliers.

No toda la información que veíamos con los histogramas se puede observar con los boxplots. Por ejemplo, la distribución bimodal de los datos de la función 2 no se observa en el boxplot. En general, los boxplots son útiles para comparar a simple vista varios conjuntos de datos, pero los histogramas dan un poco más de información de la distribución de cada conjunto de datos en particular. Comparemos los boxplots de los datos de las funciones 1, 2 y 3 con sus histogramas correspondientes.

```r
# Estas opciones son para ajustar los gráficos y que la imagen quede mejor
par(mfrow=c(2,3), oma = c(2, 1, 2, 3) + 0.1, mai = c(0.2, 0.2, 0.2, 0.2))

hist(score$func1,xlab = NULL, ylab = NULL, main = "Función 1",
     col="forestgreen", prob=TRUE)
hist(score$func2,xlab = NULL, ylab = NULL, main = "Función 2",
     col="forestgreen", prob=TRUE)
hist(score$func3, xlab = NULL, ylab = NULL, main = "Función 3",
     col="royalblue3", prob=TRUE)

boxplot(score$func1,horizontal=TRUE)
boxplot(score$func2,horizontal=TRUE)
boxplot(score$func3,horizontal=TRUE)
```



Conclusiones (6)

- Las medianas de 1 y 2 están más cerca de 1 que la de 3, que no está en el rango deseado.
- La dispersión en los datos correspondientes a la función 1 y a la 3 es similar. En el caso de 2, es bastante mayor. Esto en los histogramas se observa (aunque no tan inmediatamente) considerando que la mayor parte de los datos están concentrados en una región, mientras que para 2 no es así.
- En los histogramas vimos que los datos de la función 2 eran bimodales. En los boxplots no podemos observar esto.
- Los datos de la función 3 están distribuidos en forma asimétrica. Esto se observa a partir de la posición de la línea de la mediana en la caja, y la diferente longitud de los bigotes superior e inferior.
- Para la función 3 se observan datos atípicos, pero para 1 y 2, no. Estos eran los que estaban en la cola derecha del histograma de la función 3.

Métodos gráficos: qqplots

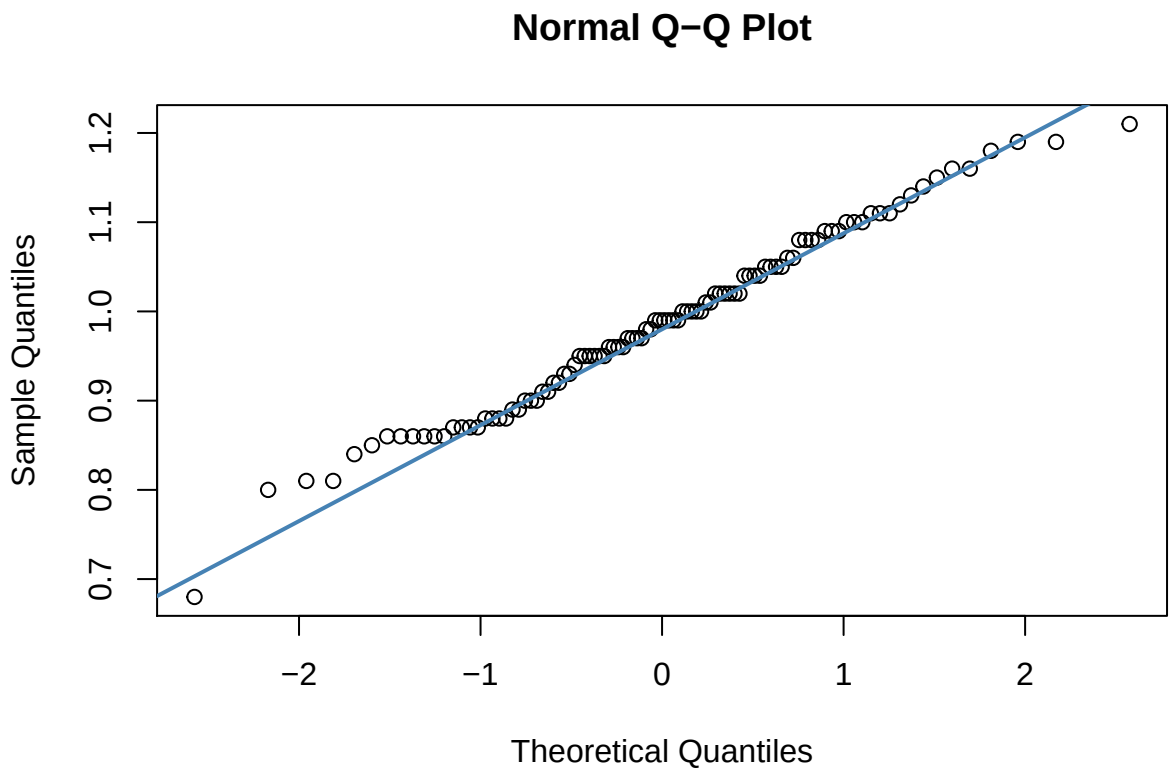
7) *Graficar los qqplots correspondientes. ¿En algún caso el ajuste normal parece razonable?*

Un qqplot (del inglés quantile-quantile, gráfico cuantil-cuantil) es un gráfico que sirve para comparar visualmente las diferencias en las distribuciones de dos conjuntos de datos (o, como haremos aquí, comparar un conjunto de datos contra una variable aleatoria de distribución conocida).

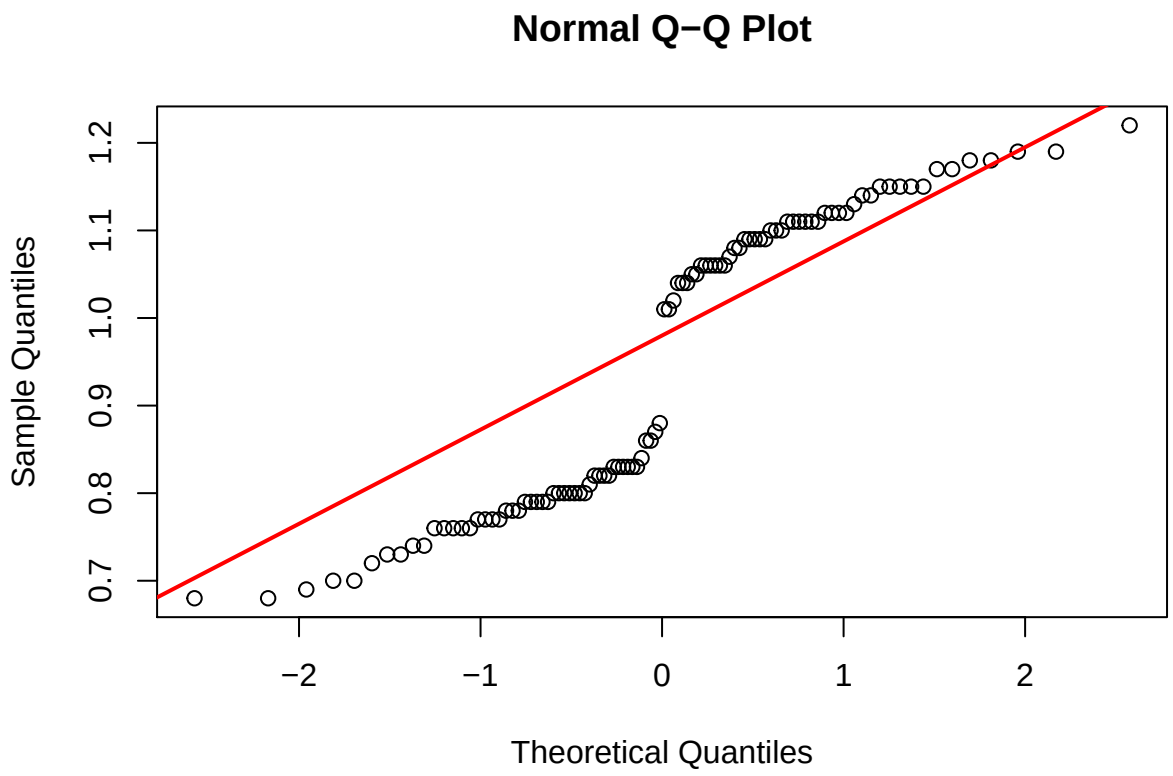
Los gráficos que haremos aquí comparan los cuantiles de los datos contra los cuantiles de una normal de igual media y desvío que los datos. El método de comparación es: primer cuantil de los datos contra primer cuantil correspondiente normal, segundo valor contra segundo cuantil, y así. Por ejemplo, cuando lleguemos a la mitad, habrá un punto que tendrá como coordenadas a la mediana de la normal (o sea, su media) y a la mediana de los datos.

Si estos valores son iguales, se hallarán sobre la recta $y = x$. Si hay diferencias en la distribución de los datos, la posición de los puntos (x, y) se alejará de la recta. Si un conjunto de datos tienen distribución normal, se mantendrán sobre esa recta. Si eso no sucede, podremos rechazar la hipótesis de que esos datos siguen una distribución normal.

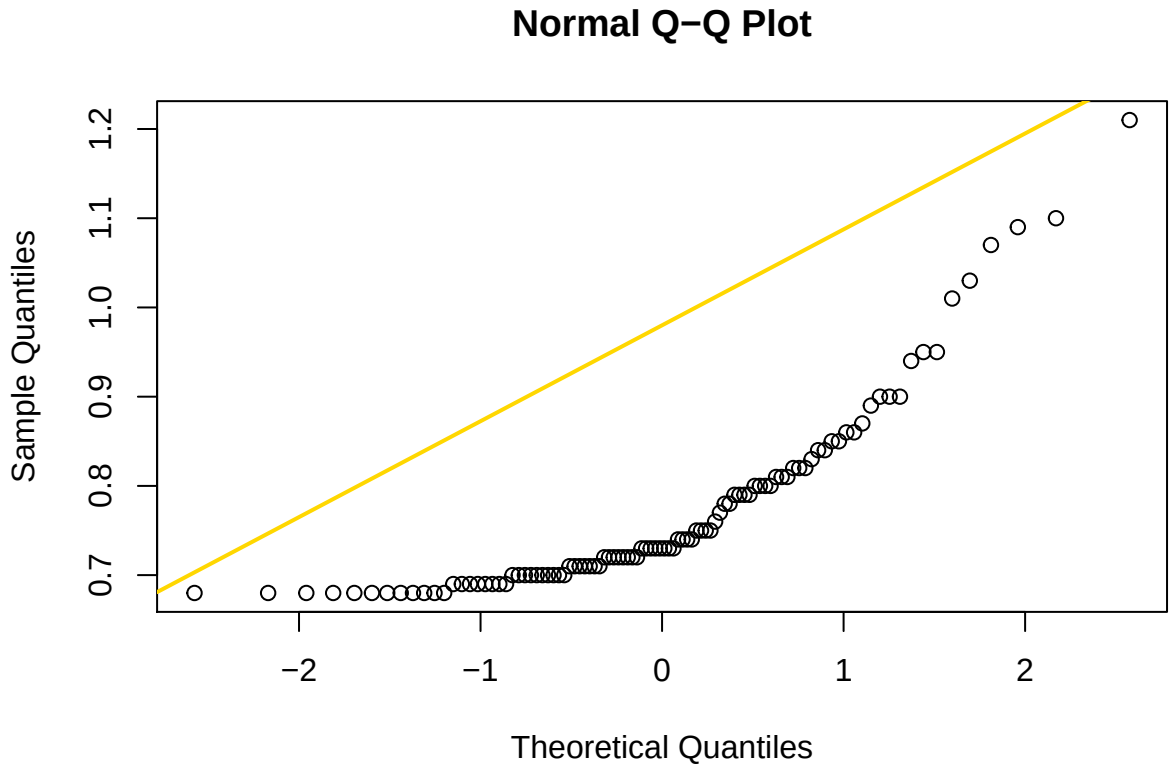
```
# Con qq-norm pedimos que nos haga el gráfico
# cuantil - cuantil, y con qqline la línea para comparar con la identidad y = x.
# R después estandariza los qqplots para que podemos comparar entre diferentes datos

with(data=score, qqnorm(func1))
qqline(score$func1, col = "steelblue", lwd = 2)
```



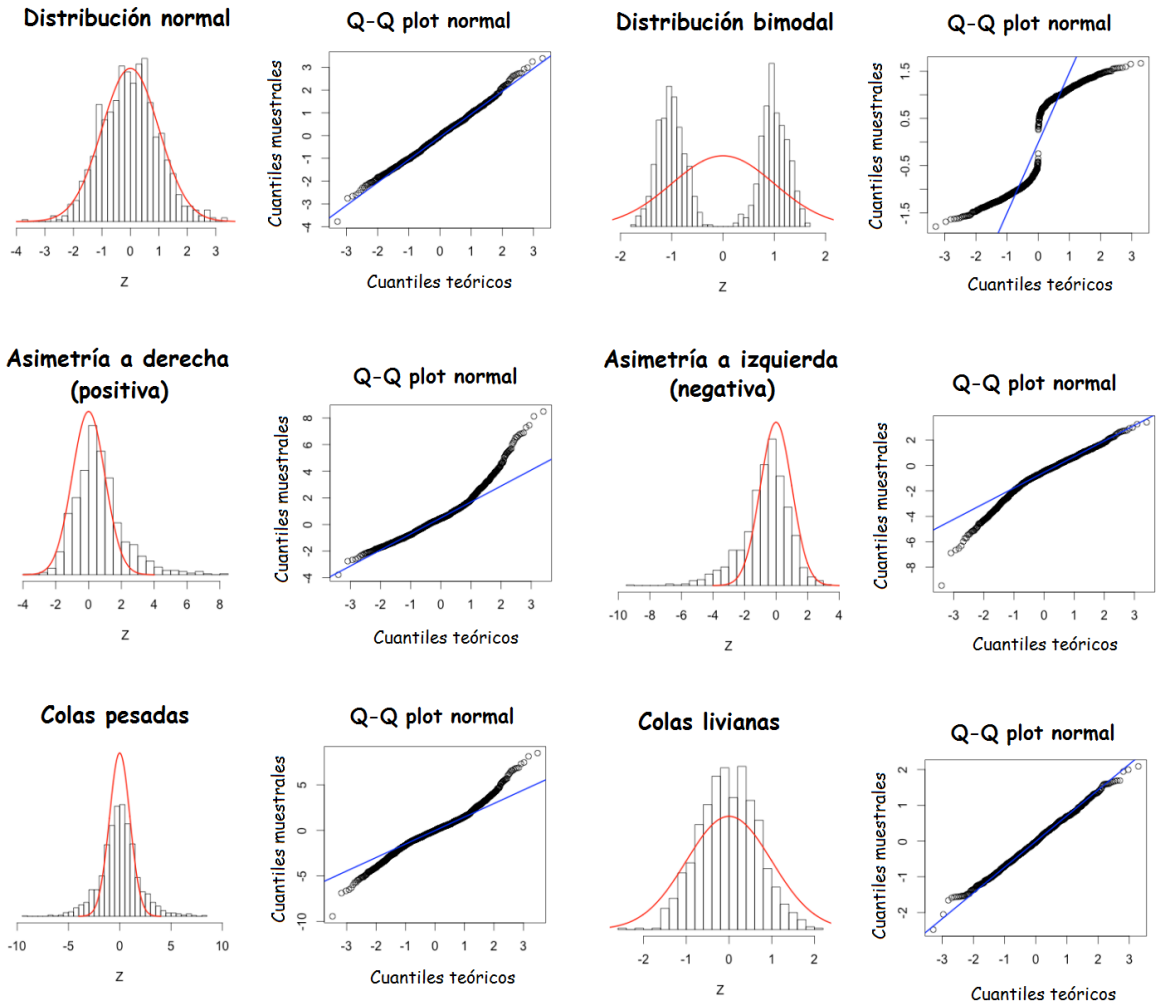

```
with(data=score, qqnorm(func2))
qqline(score$func1, col = "red", lwd = 2)
```




```
with(data=score, qqnorm(func3))
qqline(score$func1, col = "gold", lwd = 2)
```



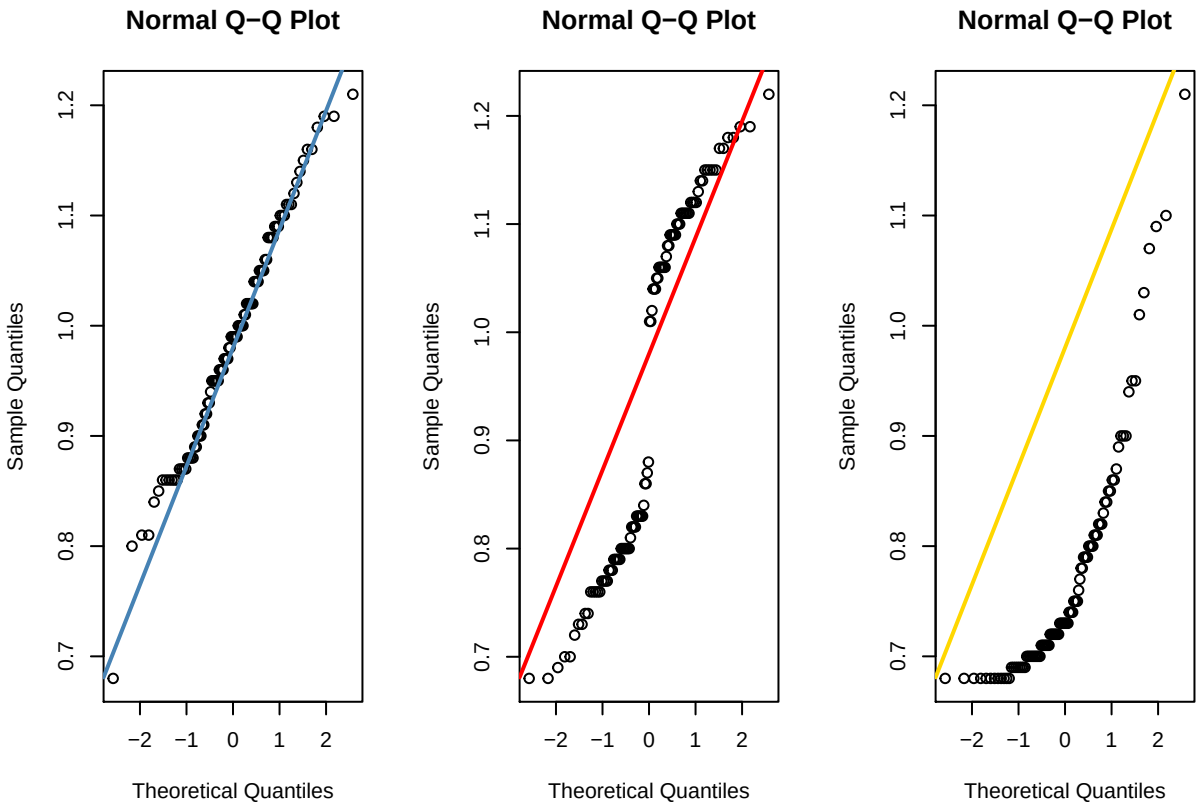
Dependiendo de cómo se desvíen los valores de los cuantiles de la recta $y = x$, podemos deducir algunas características de la distribución estudiada.



Fuente: https://seankross.com/2016/02/29/A-Q-Q-Plot-Dissection-Kit.html.

Una distribución bimodal se caracteriza por un salto en el medio de los datos. La asimetría de los datos se relaciona con la convexidad (asimetría a derecha) o concavidad (asimetría a izquierda) de la curva de datos respecto de la línea identidad (aunque pueden estar por arriba o debajo de ella). El peso relativo de las colas respecto a la distribución normal se manifiesta en la posición relativa de los puntos extremos y la línea identidad. Si primero los datos están por debajo y después por arriba, las colas son más pesadas que las de la normal (hay más valores en los extremos). Si al principio los datos están por arriba y después por debajo, las colas son más livianas (los valores están más concentrados en el centro que en las puntas). Si los datos permanecen todo el tiempo sobre la recta identidad, tendremos una distribución parecida a una normal.

```
# Para representarlos en un sólo gráfico, hacemos como con los histogramas
par(mfrow=c(1,3))

with(data=score, qqnorm(func1))
qqline(score$func1, col = "steelblue", lwd = 2)
with(data=score, qqnorm(func2))
qqline(score$func1, col = "red", lwd = 2)
with(data=score, qqnorm(func3))
qqline(score$func1, col = "gold", lwd = 2)
```



Conclusiones (7)

En este caso, podemos observar bien el comportamiento bimodal de los datos de la función 2 (reconocemos el salto que pegan en el medio).

Los datos de la función 1 ajustan bastante bien a una distribución normal, aunque para las colas el ajuste es un poco peor. El comportamiento de las colas, indicaría que son un poco más livianas que las de la normal, y que los datos estarían más concentrados en el medio. Hay algunos valores al principio que se alejan un poco de la línea identidad; eso lo podíamos ver también en el histograma, donde había una barra alta al principio.

Para la función 3, se observa que tiene una asimetría a derecha, dado que la curva que hacen los datos es convexa. La distribución de los datos es más díficil de observar, pero se ve que se concentran mucho en el centro, y sobre todo en la cola de la derecha, hay muchos valores dispersos (estos eran los outliers).

Algunos recursos para analizar qqplots (en inglés)

La página de la figura 3 analiza qqplots en bastante profundidad: https://seankross.com/2016/02/29/A-Q-Q-Plot-Dissection-Kit.html

En esta página dan algunos puntos claves a la hora de analizar qqplots: https://stats.stackexchange.com/questions/101274/how-to-interpret-a-qq-plot/101290

En esta otra discuten cuál de los tres gráficos que hicimos es el mejor para analizar normalidad de un conjunto de datos. La primera respuesta cuenta un poco de qué se ve mejor con cada gráfico. La segunda respuesta da ejemplos donde varios histogramas distintos generan los mismos boxplots, y otro donde cambiando el tamaño de los intervalos del histograma cambia la distribución aparente de los datos: https://math.stackexchange.com/questions/2432513/histogram-box-plot-and-probability-plot-which-is-better-for-assessing-normali

Esta página permite ir cambiando parámetros de una distribución y ver el impacto en el qqplot normal correspondiente: https://xiongge.shinyapps.io/QQplots/

Conclusiones finales

8) *En base a todo el análisis anterior, ¿cuál sería la función de scoring que más se ajusta a los requerimientos pedidos?*

La función 1 es la que más se ajusta a lo pedido:

- su media (y también la mediana y la media α-podada) es la más cercana a 1, aunque la función 2 también cumpliría esto.
- los datos de 1 están concentrados en el rango $1 \pm 0,1$, como se observa a partir de las medidas de dispersión y del histograma y el boxplot. La función 3 tiene una dispersión similar, pero no es en el rango correcto.
- cumple los requerimientos de ajustar aproximadamente a una normal. La función 2 no los cumple por ser una distribución bimodal, y la función 3 tiene una asimetría a derecha, con una cola derecha muy larga con outliers, y una cola a izquierda muy corta. Con eso puedo descartar 2 y 3. Pero además, a partir del qqplot de 1, se observa que la distribución es similar a una normal.
- sólo comprobamos que visualmente 1 tiene distribución similar a una normal. Para terminar de verificar (en realidad, fallar en descartar) el supuesto de normalidad de la función 1, podríamos hacer un test de hipótesis (guía 9).