Tema 1. Estadística Descriptiva con R (2ª parte)

Estadística

Ángel Serrano Sánchez de León

Índice

- Medidas de tendencia central
- Medidas de dispersión
- Medidas de asimetría

Media aritmética

- Creamos un vector numérico de notas de clase:

```
> notas <- c(8,6.5,2,5,6.6,7.1,5.3,9.5,6.3,4.5,6.1)
```

- Número de elementos en el vector:

```
> length(notas)
[1] 11
```

- **Media aritmética** (suma total entre nº de elementos):

```
> mean(notas)
[1] 6.081818
```

- Otra manera:

```
> sum(notas)/length(notas)
[1] 6.081818
```

3

Media aritmética recortada

- Vamos a añadir valores extremos y veamos su efecto en la media aritmética:

```
> notas2 <- c(notas,20,-3) # Añadimos dos hipotéticas notas
  extremas 20 y -3
> notas2
 [1] 8.0 6.5 2.0 5.0 6.6 7.1 5.3 9.5 6.3 4.5 6.1 20.0
[13] -3.0
> mean(notas2) # Afectada por valores extremos
[1] 6.453846
```

- La **media recortada** calculada con el 80% central de los valores:

```
> mean(notas2,trim=0.1) # Se elimina el 10% de los valores
  superiores y el 10% de los valores inferiores y luego se
  calcula la media
[1] 6.081818
```

Medias cuadrática, geométrica, armónica

- **Media cuadrática** (raíz cuadrada de la media de los cuadrados):

```
> sqrt(mean(notas^2))
[1] 6.358316
```

- **Media geométrica**, recordando que la raíz *n*-ésima es igual que elevar al exponente 1/*n*.

```
> prod(notas)^(1/length(notas))
[1] 5.714413
```

- **Media armónica** (inversa de la media de las inversas):

```
> 1/mean(1/notas)
[1] 5.213182
```

5

Media ponderada

- Notas de un determinado alumno en una asignatura:

```
> notas.alumno <- c(8,6.5,7,9)
```

- Cada nota corresponde a una parte de la asignatura, con diferentes pesos:

```
> pesos <- c(0.3,0.4,0.2,0.1)
```

- Aquí los pesos están en tanto por 1, pero podrían estar en %.
- La nota final es la **media ponderada**:

```
> weighted.mean(notas.alumno,pesos)
[1] 7.3
```

- Otra manera de hacerlo:

```
> sum(notas.alumno * pesos) # * actúa elemento a elemento
[1] 7.3
```

Mediana

- **Mediana** (elemento central), sin agrupar los datos:

```
> median(notas)
[1] 6.3
```

- Efectivamente, al ordenar, la mediana es el elemento número 6=(11+1)/2:

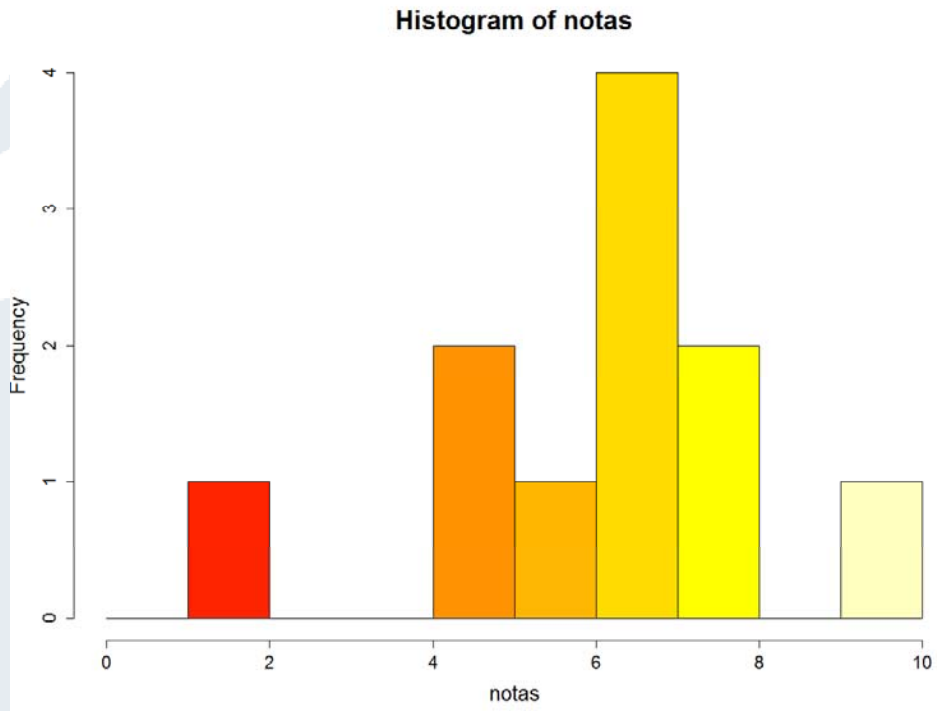
```
> sort(notas)
[1] 2.0 4.5 5.0 5.3 6.1 6.3 6.5 6.6 7.1 8.0 9.5
```

- Veamos en qué posición está la mediana en el vector sin ordenar:

```
> which(notas==median(notas))
[1] 9
```

Histograma: datos agrupados

```
> h <- hist(notas,breaks=0:10,col=heat.colors(10),
  cex.lab=1.6,cex.axis=1.4,cex.main=2)
```



8

Histograma

```
> str(h)
List of 6
 $ breaks : int [1:11] 0 1 2 3 4 5 6 7 8 9 ...
 $ counts : int [1:10] 0 1 0 0 2 1 4 2 0 1
 $ density : num [1:10] 0 0.0909 0 0 0.1818 ...
 $ mids : num [1:10] 0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5
 $ xname : chr "notas"
 $ equidist: logi TRUE
 - attr(*, "class")= chr "histogram"
```

Mediana: datos agrupados (variable continua)

- Supongamos que en vez de tener las notas de cada alumno, las tuviéramos **agrupadas** según los intervalos del histograma anterior.
- Los valores de las marcas de clase son:

```
> h$mids
[1] 0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5
```

- **OJO:** Técnicamente las marcas de clase serían: 0.45, 1.45, 2.45, etc., si consideramos 1 decimal en la nota. Por defecto, R considera los infinitos decimales.
- Las frecuencias absolutas de cada intervalo son:

```
> h$counts
[1] 0 1 0 0 2 1 4 2 0 1
```

- Las frecuencias absolutas acumuladas son:

```
> cumsum(h$counts)
[1] 0 1 1 1 3 4 8 10 10 11
```

- El valor $N/2$ vale 5.5, que no coincide con ninguna frecuencia absoluta acumulada. En este caso la mediana se calcularía por **interpolación**.

Mediana: datos agrupados (variable continua)

- Recordemos: Si $N/2$ no coincide con ninguna frecuencia absoluta acumulada, y este valor se encuentra entre las frecuencias N_{i-1} y N_i, correspondientes a los intervalos (a_{i-1}, a_i) y (a_i, a_{i+1}), entonces la mediana se calcula por **interpolación** como:

$$M_e = a_i + \frac{N/2 - N_{i-1}}{n_i}(a_{i+1} - a_i)$$

- Donde:
  - $N = 11$ (nº total de datos)
  - Intervalo anterior: $a_{i-1} = 5$, $a_i = 6$, $N_{i-1} = 4$
  - Intervalo que contiene la mediana: $a_i = 6$, $a_{i+1} = 7$, $n_i = 4$, $N_i = 8$

Mediana: datos agrupados (variable continua)

```
> # El valor N/2 se supera con la frecuencia absoluta acumulada número
  7, luego i=7
> N <- length(notas) # Nº total de datos agrupados: 11
> ai <- h$mids[7]-(h$mids[7]-h$mids[6])/2 # Frontera inferior del
  intervalo que contiene la mediana: 6
> aimas1 <- h$mids[7]+(h$mids[7]-h$mids[6])/2 # Frontera superior del
  intervalo que contiene a la mediana: 7
> Nimenos1 <- cumsum(h$counts)[6] # Frecuencia absoluta acumulada del
  intervalo anterior: 4
> ni <- h$counts[7] # Frecuencia absoluta del intervalo que contiene
  la mediana: 4
> mediana <- ai + (N/2-Nimenos1)*(aimas1-ai)/ni
> mediana
[1] 6.375
```

- ¿Por qué no coincide con el valor de median(notas), que antes vimos que era 6.3?

Moda de variables categóricas

- Creamos un vector de caracteres con el nivel de estudios de una serie de personas:

```
> nivel.estudios<-factor(c("Primaria","ESO",
  "Universidad","ESO","Bachillerato","Bachillerato",
  "Bachillerato", "Primaria","FP","Universidad",
  "ESO","Sin estudios","FP","Universidad","Universidad",
  "FP", "Bachillerato","Universidad"),
  levels=c("Sin estudios","Primaria","ESO",
  "Bachillerato","FP", "Universidad"),ordered=TRUE)
> nivel.estudios
[1] Primaria ESO Universidad ESO
[5] Bachillerato Bachillerato Bachillerato Primaria
[9] FP Universidad ESO Sin estudios
[13] FP Universidad Universidad FP
[17] Bachillerato Universidad
6 Levels: Sin estudios < Primaria < ... < Universidad
```

13

Moda de variables categóricas

- La función `table` devuelve la distribución de frecuencias absolutas:

```
> table(nivel.estudios)
nivel.estudios
Sin estudios Primaria ESO Bachillerato FP
           1        2   3            4  3
Universidad
          5
```

- La moda es la categoría con la frecuencia absoluta máxima:

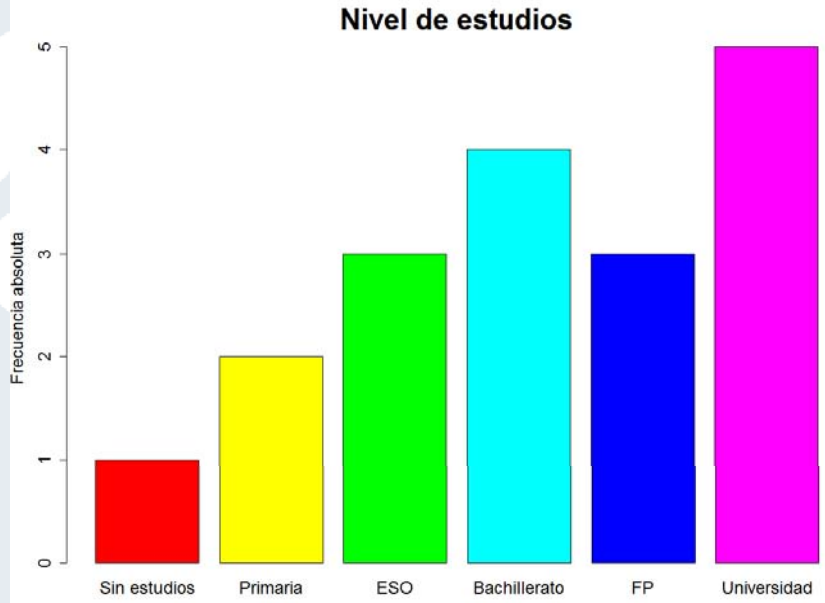
```
> names(which.max(table(nivel.estudios)))
[1] "Universidad"
```

- Valor de la frecuencia modal:

```
> max(table(nivel.estudios))
[1] 5
```

Moda de variables categóricas

```
> barplot(table(nivel.estudios),ylab="Frecuencia
absoluta",main="Nivel de estudios",
col=rainbow(length(table(nivel.estudios))),
cex.axis=1.5,cex=1.5,cex.lab=1.5,cex.main=2.5)
```



Moda: datos agrupados (variable continua)

- La variable `notas` es continua y vemos que no se repite ningún valor (no tiene sentido el concepto de moda).
- Sin embargo, supongamos de nuevo que trabajamos con los datos agrupados por intervalos según el histograma visto anteriormente.
- El intervalo modal es aquel con la frecuencia absoluta más alta.

```
> h$counts
[1] 0 1 0 0 2 1 4 2 0 1
```

- Séptimo intervalo, correspondiente a [6,7).

Moda: datos agrupados (variable continua)

- Recordemos: la mayor frecuencia absoluta n_j corresponde al intervalo (a_j, a_{j+1}), llamado **intervalo modal**. La frecuencia de dicho intervalo supera al intervalo premodal en δ_1, y supera al intervalo postmodal en δ_2. Entonces la moda se calcula por **interpolación** como:

$$M_o = a_j + \frac{\delta_1}{\delta_1 + \delta_2}\left(a_{j+1} - a_j\right)$$

- donde:
  - Intervalo premodal: $a_{j-1} = 5$, $a_j = 6$, $n_{j-1} = 1$, $\delta_1 = 4 - 1 = 3$
  - Intervalo modal: $a_j = 6$, $a_{j+1} = 7$, $n_j = 4$
  - Intervalo postmodal: $a_{j+1} = 7$, $a_{j+2} = 8$, $n_{j+1} = 2$, $\delta_2 = 4 - 2 = 2$

17

Moda: datos agrupados (variable continua)

```
> j <- which.max(h$counts) # Vale 7
> amplitudIntervalo <- h$mids[j]-h$mids[j-1] # Vale 1
> aj <- h$mids[j]-amplitudIntervalo/2 # Vale 6
> ajmas1 <- aj + amplitudIntervalo # Vale 7
> delta1 <- h$counts[j]-h$counts[j-1] # 4 – 1 = 3
> delta2 <- h$counts[j]-h$counts[j+1] # 4 – 2 = 2
> moda <- aj + delta1*amplitudIntervalo/(delta1+delta2)
> moda
[1] 6.6
```

Mínimo, máximo y recorrido

- Máximo:

```
> max(notas)
[1] 9.5
```

- Mínimo:

```
> min(notas)
[1] 2
```

- Recorrido (o rango):

```
> range(notas)
[1] 2.0 9.5
> range(notas)[2]-range(notas)[1]  # Idem max(notas)-
  min(notas)
[1] 7.5
```

Mínimo, máximo y recorrido

- También para variable categórica ordinal:

```
> max(nivel.estudios)
[1] Universidad
6 Levels: Sin estudios < Primaria < ESO < Bachillerato
  < ... < Universidad
> min(nivel.estudios)
[1] Sin estudios
6 Levels: Sin estudios < Primaria < ESO < Bachillerato
  < ... < Universidad
> range(nivel.estudios)
[1] Sin estudios Universidad
6 Levels: Sin estudios < Primaria < ESO < Bachillerato
  < ... < Universidad
```

Cuartiles

- Primer cuartil (o percentil del 25%):

```
> quantile(notas,probs=0.25)
25%
5.15
```

- Tercer decil (o percentil del 30%):

```
> quantile(notas,probs=0.3)
30%
5.3
```

- Percentil del 79%:

```
> quantile(notas,probs=0.79)
79%
7.05
```

- Resumen automático de los cuartiles:

```
> summary(notas)
 Min. 1st Qu. Median  Mean 3rd Qu.  Max.
2.000   5.150  6.300 6.082  6.850  9.500
```

21

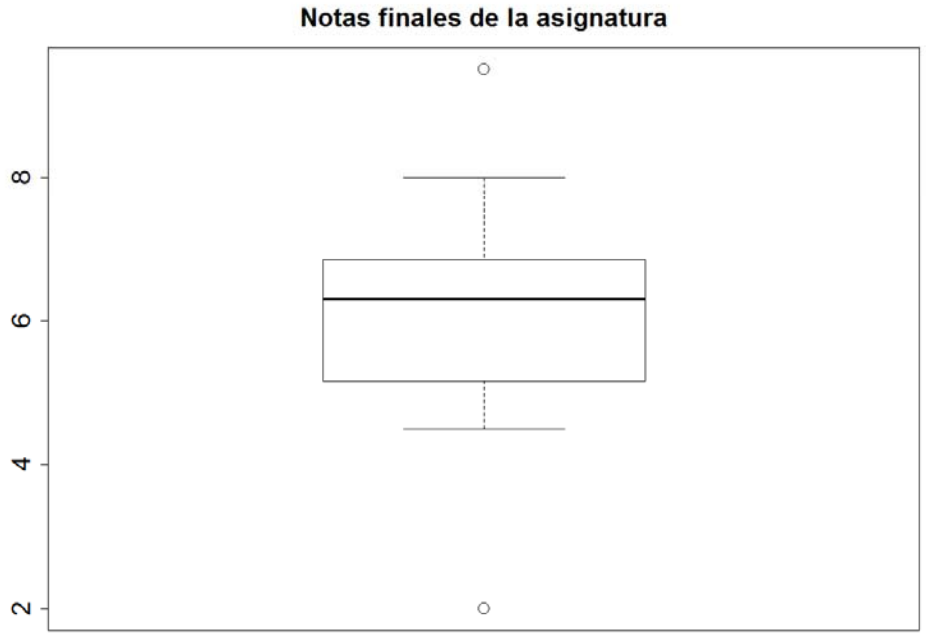
Recorrido intercuartílico

- Por defecto, la función `quantile` devuelve un vector con los nombres de cada cuartil devuelto.
- Quitando esos nombres, podemos calcular el recorrido intercuartílico:

```
> ri.vector <- quantile(notas,probs=c(0.25,0.75),names=FALSE)
> ri.vector # Vector con el primer y el tercer cuartil
[1] 5.15 6.85
> ri <- ri.vector[2]-ri.vector[1] # Recorrido intercuartílico
> ri
[1] 1.7
> rsi <- ri/2 # Recorrido semiintercuartílico
> rsi
[1] 0.85
```

Diagrama de caja (boxplot)

```
> boxplot(notas,main="Notas finales de la
  asignatura",cex.axis=2,cex.main=2,cex=2)
```



Desviación absoluta

- Desviación absoluta respecto de la media:

```
> mean(abs(notas-mean(notas)))
[1] 1.368595
```

- Desviación absoluta respecto de la mediana:

```
> mad(notas,constant=1)
[1] 1
> median(abs(notas-median(notas)))
[1] 1
```

- Por conveniencia, `mad` devuelve un valor 1,4826 veces mayor al resultado real (para corregir este factor, se asigna `constant = 1`).

Desviación típica

- Desviación típica sin sesgo (la que tiene $N - 1$ en el denominador):

```
> sd(notas)
[1] 1.945157
```

- Desviación típica sesgada (la que tiene N en el denominador):

```
> sqrt(mean((notas-mean(notas))^2))
[1] 1.854635
> sd(notas)*sqrt((N-1)/N)
[1] 1.854635
> sqrt(mean(notas^2)-(mean(notas))^2)
[1] 1.854635
```

Varianza

- Varianza sin sesgo (la que tiene $N - 1$ en el denominador):

```
> var(notas)
[1] 3.783636
```

- Varianza sesgada (la que tiene N en el denominador):

```
> mean((notas-mean(notas))^2)
[1] 3.439669
> var(notas)*(N-1)/N
[1] 3.439669
> mean(notas^2)-(mean(notas))^2
[1] 3.439669
```

Asimetría (o sesgo)

- Coeficiente de asimetría de Fisher:

```
> require(moments) # Necesitamos el paquete
  moments
> skewness(notas) # Sesgo
[1] -0.3624794
```

- La asimetría negativa (cola izquierda) también se puede ver en el diagrama de caja porque la mediana está más próxima al 3er cuartil que al 1º.
  - Los datos menores que la mediana están mucho más dispersos que los datos mayores que la mediana.

27

Curtosis

- También en el paquete moments:

```
> kurtosis(notas)
[1] 3.356527
```

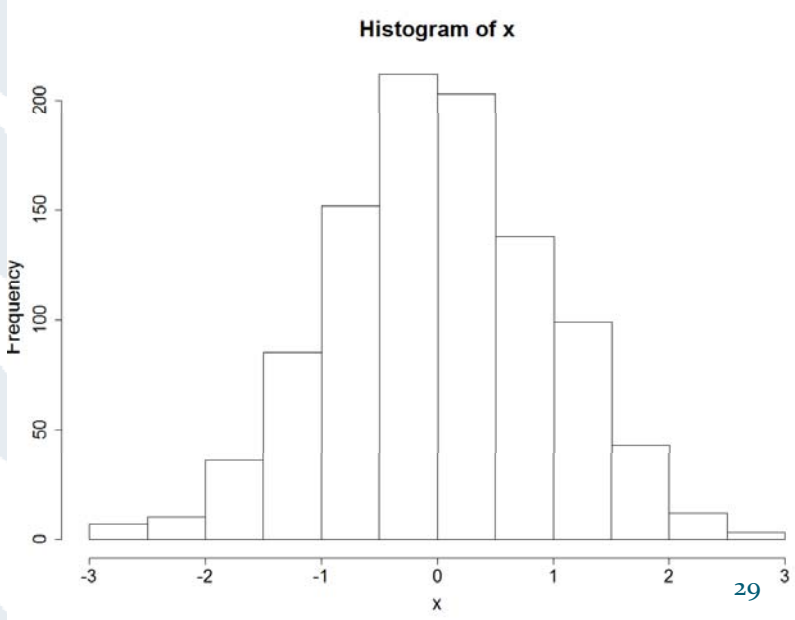
- Al ser mayor que 3, es una distribución de valores leptocúrtica (más picuda que una curva normal).

Números aleatorios

- Generación de un vector de 1000 números aleatorios según una **distribución normal** (media 0 y desviación típica 1):

```
> x <- rnorm(1000)
> skewness(x)
[1] -0.05653617
> kurtosis(x)
[1] 2.920826
```

La distribución normal teórica no tiene asimetría (=0) y es mesocúrtica (curtosis = 3).



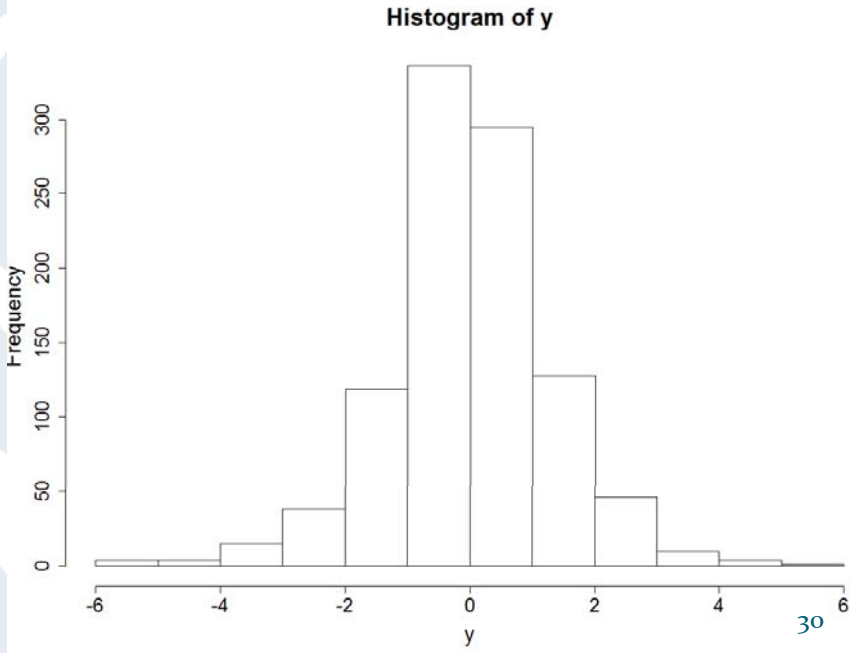
29

Números aleatorios

- Generación de un vector de 1000 números aleatorios según una **distribución t de Student** (con 5 grados de libertad):

```
> y <- rt(1000,5)
> skewness(y)
[1] -0.2320058
> kurtosis(y)
[1] 4.856477
```

La distribución t de Student teórica no tiene asimetría (= 0) y es leptocúrtica (curtosis > 3).



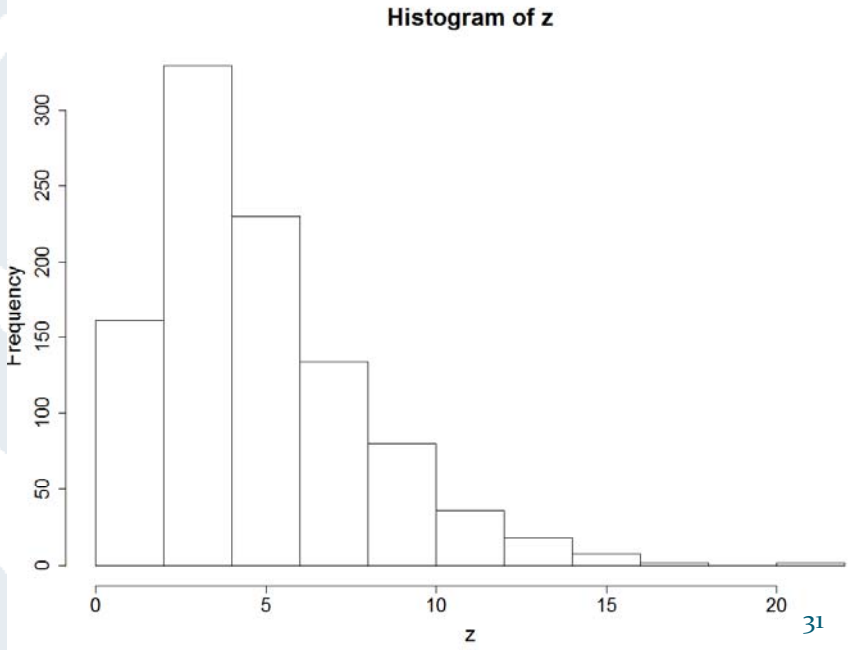
30

Números aleatorios

- Generación de un vector de 1000 números aleatorios según una **distribución χ^2 ("ji cuadrado")** (con 5 grados de libertad):

```
> z <- rchisq(1000,5)
> skewness(z)
[1] 1.266903
> kurtosis(z)
[1] 5.092302
```

La distribución χ^2 teórica tiene asimetría derecha (> 0) y es leptocúrtica (curtosis > 3).



31