Estadística Descriptiva en R: Funciones

José Enrique Martín García
Universidad Politécnica de Gimialcón

Summary

La función más recurrida para obtener estadísticos descriptivos de los datos es Summary

```
> p1   <- rnorm(1000,  3,  0.8)
> p2   <- rnorm(1000,  2,  0.5)
> p  <- c(p1,  p2)
> altura  <- c(rnorm(1000,  87,  7),  rnorm(1000,  97,  6))
> #  length(p); hist(p)
> grupo <- c(rep("M",  1000),  rep("H",  1000))
> df <- data.frame(p, altura, grupo)
> head(df)
         p   altura grupo
1 3.631115 99.81936     M
2 3.135630 89.78738     M
3 3.808861 97.43998     M
4 3.533223 92.98578     M
5 1.602240 87.92617     M
6 3.293906 91.16983     M
> str(df)
'data.frame':       2000 obs. of  3 variables:
 $ p     : num  3.63 3.14 3.81 3.53 1.6 ...
 $ altura: num  99.8 89.8 97.4 93 87.9 ...
 $ grupo : Factor w/ 2 levels "H","M": 2 2 2 2 2 2 2 2 2 2 ...
```

```
> summary(df)
       p                altura       grupo
 Min.   :0.08224   Min.   : 65.55   H:1000
 1st Qu.:1.90634   1st Qu.: 86.96   M:1000
 Median :2.38038   Median : 92.51
 Mean   :2.51208   Mean   : 92.30
 3rd Qu.:3.06516   3rd Qu.: 98.11
 Max.   :6.04453   Max.   :116.60
```

Summary

También podemos asignar dataframes a cada grupo para simplificar la sintaxis

```
> df.M <-  df[which(df$grupo == "M"),]
> summary(df.M)
       p                altura        grupo
 Min.   :0.08224   Min.   : 65.55   H:   0
 1st Qu.:2.48552   1st Qu.: 82.94   M:1000
 Median :3.01158   Median : 87.68
 Mean   :3.02616   Mean   : 87.56
 3rd Qu.:3.56867   3rd Qu.: 92.14
 Max.   :6.04453   Max.   :106.88
```

```
> df.H <-  df[which(df$grupo == "H"),]
> summary(df.H)
       p                altura        grupo
 Min.   :0.2053   Min.   : 77.77   H:1000
 1st Qu.:1.6572   1st Qu.: 92.81   M:   0
 Median :2.0126   Median : 97.00
 Mean   :1.9980   Mean   : 97.05
 3rd Qu.:2.3391   3rd Qu.:101.15
 Max.   :3.3470   Max.   :116.60
```

Función stat.desc()

La función stat.desc() del paquete pastecs, tiene varias opciones muy interesantes

```
> library("pastecs")
> stat.desc(df)
                        p       altura grupo
nbr.val      2.000000e+03 2.000000e+03    NA
nbr.null     0.000000e+00 0.000000e+00    NA
nbr.na       0.000000e+00 0.000000e+00    NA
min          8.224415e-02 6.554759e+01    NA
max          6.044528e+00 1.166041e+02    NA
range        5.962284e+00 5.105648e+01    NA
sum          5.024165e+03 1.846078e+05    NA
median       2.380383e+00 9.251240e+01    NA
mean         2.512083e+00 9.230389e+01    NA
SE.mean      1.909287e-02 1.805699e-01    NA
CI.mean.0.95 3.744401e-02 3.541249e-01    NA
var          7.290755e-01 6.521097e+01    NA
std.dev      8.538592e-01 8.075331e+00    NA
coef.var     3.399009e-01 8.748635e-02    NA
```

Función stat.desc()

Veamos algunos cambios en los argumentos p.e. norm=TRUE

```
> stat.desc(df[-3],  norm   =  TRUE)
                         p        altura
nbr.val      2.000000e+03  2.000000e+03
nbr.null     0.000000e+00  0.000000e+00
nbr.na       0.000000e+00  0.000000e+00
min          8.224415e-02  6.554759e+01
max          6.044528e+00  1.166041e+02
range        5.962284e+00  5.105648e+01
sum          5.024165e+03  1.846078e+05
median       2.380383e+00  9.251240e+01
mean         2.512083e+00  9.230389e+01
SE.mean      1.909287e-02  1.805699e-01
CI.mean.0.95 3.744401e-02  3.541249e-01
var          7.290755e-01  6.521097e+01
std.dev      8.538592e-01  8.075331e+00
coef.var     3.399009e-01  8.748635e-02
skewness     5.411580e-01 -2.106689e-01
skew.2SE     4.943776e+00 -1.924576e+00
kurtosis     1.160297e-01 -1.720119e-01
kurt.2SE     5.302614e-01 -7.861029e-01
normtest.W   9.797425e-01  9.962903e-01
normtest.p   3.059961e-16  8.206427e-05
```

Función stat.desc()

Veamos algunos cambios en los argumentos p.e. basic =FALSE y norm=TRUE

```
> stat.desc(df.M[-3],  basic =  FALSE,  norm   =  TRUE)
                      p      altura
median       3.01157743 87.68085348
mean         3.02615949 87.55634911
SE.mean      0.02606603  0.22067863
CI.mean.0.95 0.05115045  0.43304682
var          0.67943789 48.69905598
std.dev      0.82428023  6.97847089
coef.var     0.27238493  0.07970263
skewness     0.05906331 -0.12030911
skew.2SE     0.38182307 -0.77775520
kurtosis     0.03840118 -0.11264143
kurt.2SE     0.12424810 -0.36445455
normtest.W   0.99874020  0.99774431
normtest.p   0.71550603  0.19133830
```

Función tapply

Con la función tapply nos podemos construir fácilmente nuestras tablas de descriptivos de una forma muy elegante

```
> tapply(df$p,  df$g,  mean)
       H        M
1.998006 3.026159
```

```
> m <- tapply(df$p,  df$g, mean)
> s <- tapply(df$p, df$g, sd)
> m2 <- tapply(df$p, df$g, median)
> n <- tapply(df$p,df$g,length) cbind(media = m, sd = s, mediana = m2,n)
> n <- tapply(df$p,df$g,length)
> cbind(media  =  m, sd  =  s, mediana =  m2,   n)
     media        sd  mediana    n
H 1.998006 0.5003633 2.012560 1000
M 3.026159 0.8242802 3.011577 1000
```

Tablas de frecuencias y probabilidades

En estadística, se le llama distribución de frecuencias a la agrupación de datos en categorías mutuamente excluyentes que indican el número de observaciones en cada categoría

```
> pais <- c( "ES", "ES", "ES", "US", "US","UK" )
> sexo <- c( "F", "F", "M", "F", "M","M" )
> t <- table( pais, sexo )  # tabla de frecuencias absolutas
> t
    sexo
pais F M
  ES 2 1
  UK 0 1
  US 1 1
> # frec relativas
> prop.table( t )  # porcentajes totales
    sexo
pais         F         M
  ES 0.3333333 0.1666667
  UK 0.0000000 0.1666667
  US 0.1666667 0.1666667
> prop.table( t ) * 100
    sexo
pais        F        M
  ES 33.33333 16.66667
  UK  0.00000 16.66667
  US 16.66667 16.66667
```

Tablas de frecuencias y probabilidades

```
> # porcentajes por filas
> prop.table( t, 1 )
    sexo
pais         F         M
  ES 0.6666667 0.3333333
  UK 0.0000000 1.0000000
  US 0.5000000 0.5000000
> # porcentajes por columnas
> prop.table( t, 2 )
    sexo
pais         F         M
  ES 0.6666667 0.3333333
  UK 0.0000000 0.3333333
  US 0.3333333 0.3333333
```

Comandos básicos

Nombre Comando	Explicación
install.packages()	Visualiza los paquetes de datos disponibles en internet.
install.packages(*name*)	Descarga el paquete indicado.
library()	Visualiza los paquetes disponibles.
library(*name*)	Carga el paquete indicado.
data()	Visualizar los datos disponibles.
data(*name*)	Carga en memoria el dato indicado.

Nombre Comando	Explicación
class(*data*)	Muestra el tipo de objeto.
dim(*data*)	Muestra las dimensiones del objeto.
ncol(*data*), nrow(*data*)	Muestra el número de columnas/filas del objeto.
names(*data*)	Muestra los nombres de las columnas.
objects(), ls()	Visualiza las variables cargadas en memoria.
rm(*data1*, *data2*)	Elimina las variables indicadas.
help(*data*), ?*data*	Muestra la ayuda asociada con el comando o variable.
Ctrl+L	Borra la pantalla.

Comandos básicos

Nombre Comando	Explicación
summary(*data*)	Resumen estadístico
min(*data*)	Mínimo
max(*data*)	Máximo
range(*data*)	Rango
mean(*data*)	Media aritmética
median(*data*)	Mediana
length(*data*)	Tamaño
sd(*data*)	Desviación típica
var(*data*), cov(*data*)	Varianza
cor(*data*)	Correlación
quantile(*data*, 0.25)	Cuantil Q1
quantile(*data*, 0.75)	Cuantil Q3
IQR(*data*)	Rango intercuartílico
sort(*data*)	Ordenar
table(*data*)	Tabla de frecuencias absolutas

Comandos básicos

Nombre Comando	Explicación
stem(*data*)	Diagrama de tallos y hojas
hist(*data*)	Histograma
boxplot(*data*)	Gráfico boxplot
plot(*data1*, *data2*)	Gráfico de puntos
pairs(*data*)	Gráfico de dispersión cruzado