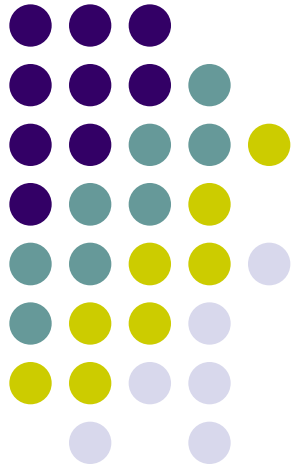
Metodología del analisis estadístico con R



Metodología del Análisis Estadístico con R

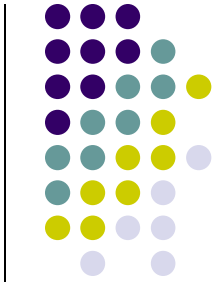
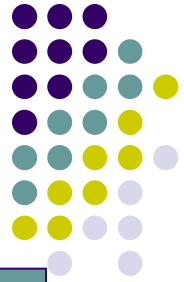


Tabla de contenidos

Objetivos

- Cargar hojas de datos existentes en libros de r (datasets).
- Estudiar funciones elementales para la generación de gráficos.
- Realizar resúmenes descriptivos de los datos.
- Resolver problemas de regresión simple y contrastes de hipótesis.
- Guardar resultados de un análisis estadístico (de tipo texto y gráfico).

Práctica de Estadística en R



Aplicación.

- Datos: chickwts, cars e iris (fuente: *package datasets* de R)
- Problema: *realizar una descripción estadística elemental de las variables de las hojas de datos*

Resolución.

Paso 1. Cargar las hojas de datos

Paso 2. Construir tablas de frecuencias

Paso 3. Representar gráficamente los datos

Paso 4. Realizar una descripción numérica

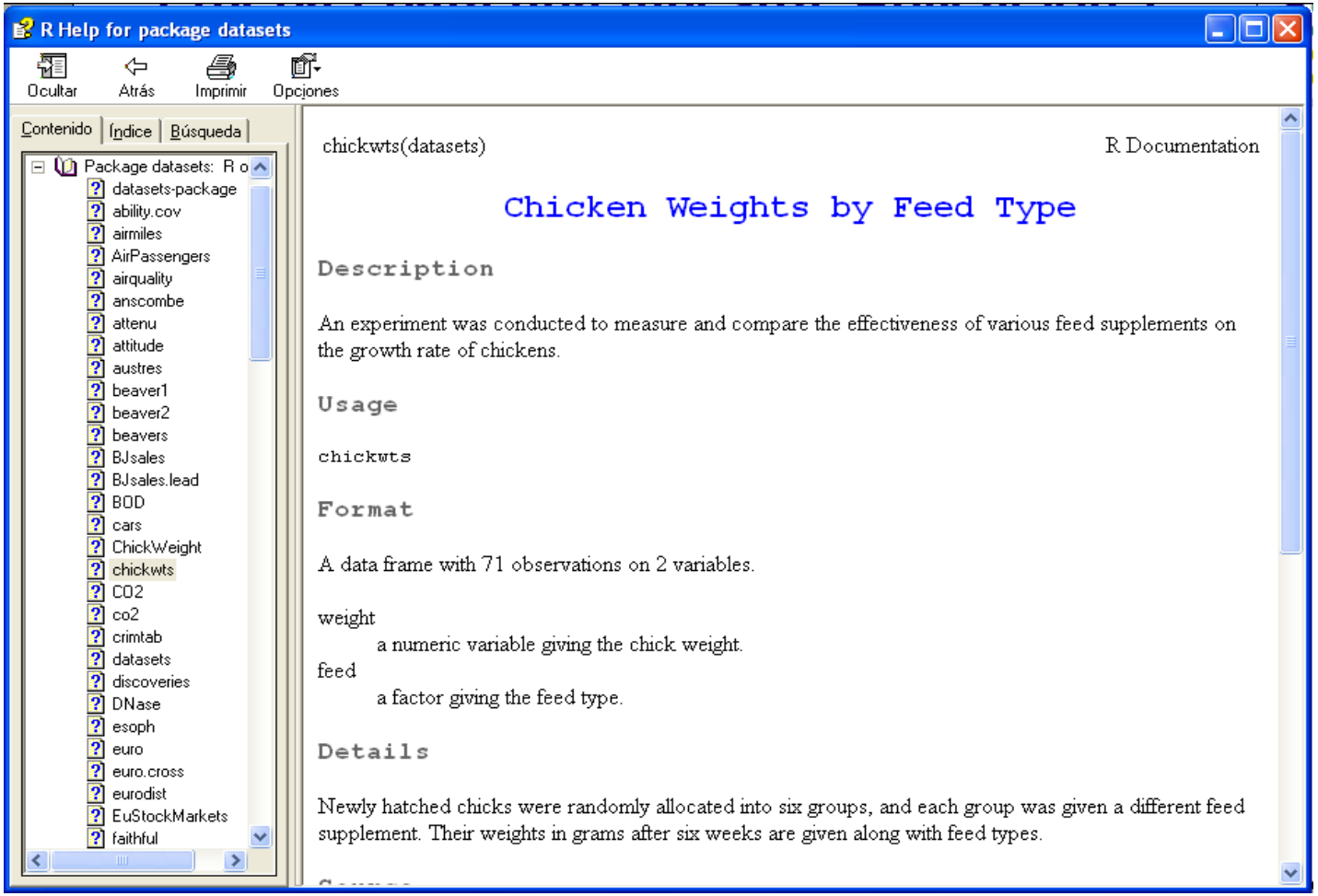
Paso 5. Resolver un problema de regresión lineal

Paso 6. Resolver contrastes de hipótesis elementales

Práctica de Estadística en R

Los datos (el *package* datasets)

```
R Console
> help(datasets)    #Puedes explorar los datos disponibles
```



Práctica de Estadística en R
Paso 1. Cargar, visualizar y exportar los datos

Descripción de los datos ***chickwts(datasets)***

Medidas de dos variables, weight y feed, para una muestra de 71 gallinas

Formato: *data frame* con 71 observaciones y 2 variables

[,1]	weight	numérico	peso de los polluelos
[,2]	feed	categórica	tipo de alimentación

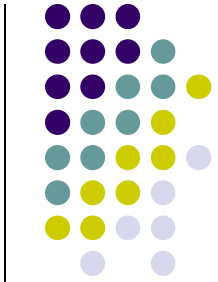
```
R Console
> data(chickwts)
> chickwts
```

Es posible exportar los datos desde R en formato ASCII (fichero de texto) utilizando la función write.table:

```
R Console
> chickwts
> write.table(chickwts,file="chickwts.txt",dec=",",row.names=F)
```

Práctica de Estadística en R

Paso 2. Construir tablas de frecuencias



Tablas estadísticas

Función: table()
Descripción: Crea una tabla de la variable indicada en sus argumentos. Calcula la frecuencia absoluta

Función: cumsum()
Descripción: Representa la frecuencia absoluta acumulada

Función: prop.table()
Descripción: Representa la frecuencia relativa

```
> table(chickwts$feed)
  casein horsebean   linseed  meatmeal   soybean sunflower
      12         10         12         11         14         12
> cumsum(table(chickwts$feed))
   casein horsebean   linseed  meatmeal   soybean sunflower
      12         22         34         45         59         71
> prop.table(tt)
    casein horsebean   linseed  meatmeal   soybean sunflower
 0.1690141 0.1408451 0.1690141 0.1549296 0.1971831 0.1690141
```

Representaciones gráficas en R

Para una demostración, escribir en la R-consola demo(“graphics”)

Gráficos unidimensionales

- **hist**(x, breaks = "Sturges", include.lowest = TRUE, right = TRUE,…)
- **boxplot**(x,…)
- **barplot**(height,…)

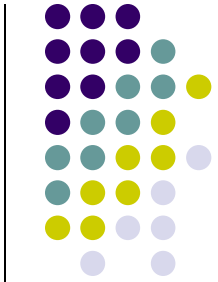
Gráficos bidimensionales

- **plot**(x, y, type=”p”, lty=1,…)
- **qqplot**(x, y, plot=TRUE)
- **qqnorm**(x, datax=FALSE, plot=TRUE)

Gráficos tridimensionales

- **contour**(x, y, z, …)
- **image**(x, y, z, …)
- **persp**(x, y, z, …)

Representaciones gráficas en R



Funciones gráficas

points(x,y,…) Añade puntos o líneas conectadas al gráfico actual
lines(x,y,…)

abline(a,b,…) Línea de pendiente a y que corta al origen en b
abline(h,…) Línea horizontal que corta al eje y en h=y
abline(v,…) Línea vertical que corta el eje x en v=x

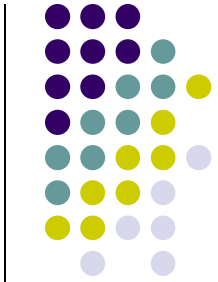
polygon(x,y,…) Dibuja un polígono

title(main,sub) Añade título y subtítulo al gráfico actual

axis(side,…) Añade ejes al gráfico actual (side=1,2,3,4)

text (x, y=NULL, labels=seq_along(x),…) Añade texto al gráfico actual

Representaciones gráficas en R



Control de los parámetros gráficos:

par(…) controla los parámetros gráficos de R:

- colores del gráfico
- fuente del texto
- símbolos del gráfico
- etiquetas de los ejes
- márgenes
- …

Nota. Algunos de estos parámetros también se pueden especificar como argumentos de funciones específicas de gráficos (plot, hist,etc.)

- **mfcol, mfrow** Matriz de gráficos
- **mar, mai, mgp** Márgenes
- **main = ’título’**
- **sub = ’título de abajo’**
- **cex.axis, cex.lab, cex.main T**amaño fuente
- **lty** Tipo de Línea
- **pch** Caracter de dibujo
- **xlab = ’etiqueta del eje x’**
- **ylab = ’etiqueta del eje y’**
- **xlim = c(*xminimo; xmaximo*)**
- **ylim = c(*yminimo; ymaximo*)**

Práctica de Estadística en R

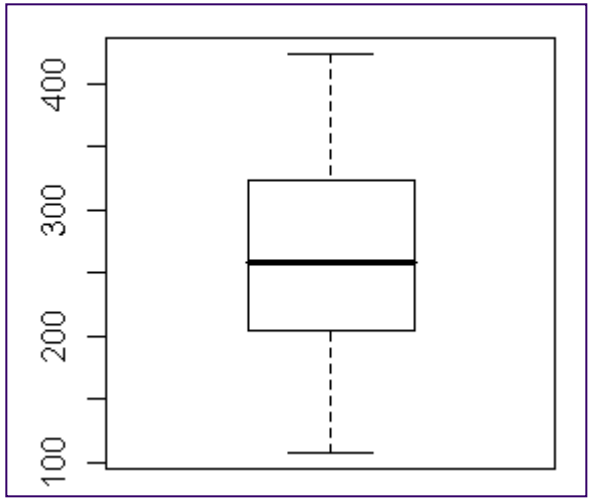
Paso 3. Representar gráficamente los datos

Boxplot o gráfico de cajas

Es un gráfico exploratorio que permite visualizar de una forma clara la distribución de los datos y sus principales características (dispersión, asimetría, medidas de posición central,...).

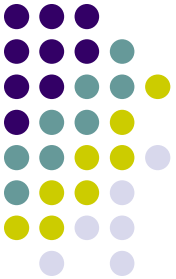
Permite comparar diversos conjuntos de datos simultáneamente.
La función boxplot permite realizar dicha representación en R. Puedes escribir help(boxplot) para obtener una descripción de su sintaxis.

```
> boxplot(chickwts$weight)
```



Práctica de Estadística en R

Paso 3. Representar gráficamente los datos



Histograma

El histograma de un conjunto de datos es un gráfico de barras que representan las frecuencias con que aparecen las mediciones agrupadas en intervalos.

Es útil para apreciar la forma de la distribución de los datos, si se escoge adecuadamente el número de clases y su amplitud. Se puede utilizar para comparar dos o más muestras o poblaciones.

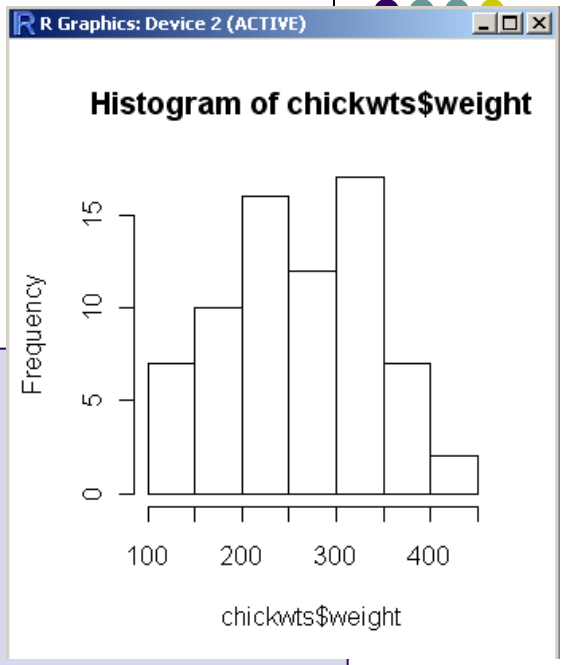
La función hist permite realizar dicha representación en R. Puedes escribir help(hist) para obtener una descripción de su sintaxis. Algunos de sus argumentos:

- **x**: vector para el que se construye el histograma.
- **breaks**: puntos de corte (un vector cuyos elementos indican los puntos de corte que definen las clases o intervalos, o un método para calcularlos).
- **freq**: TRUE (representación de frecuencias absolutas) o FALSE (relativas)
- **col**: Define el color de las barras. Por defecto, “NULL” produce barras sin fondo.
- **plot**: Argumento lógico. “TRUE” produce el gráfico del histograma; “FALSE” produce una tabla de frecuencias

Práctica de Estadística en R

Paso 3. Representar gráficamente los datos

Histograma




```
> hist(chickwts$weight)
> hist(chickwts$weight,plot=F)
$breaks
[1] 100 150 200 250 300 350 400 450
$counts
[1]  7 10 16 12 17  7  2
$intensities
[1] 0.0019718306 0.0028169014 0.0045070423 0.0033802817 0.0047887324
[6] 0.0019718310 0.0005633803
$density
[1] 0.0019718306 0.0028169014 0.0045070423 0.0033802817 0.0047887324
[6] 0.0019718310 0.0005633803
$mids
[1] 125 175 225 275 325 375 425
$xname
[1] "chickwts$weight"
$equidist
[1] TRUE
attr(,"class")
[1] "histogram"
```

Práctica de Estadística en R

Paso 3. Representar gráficamente los datos

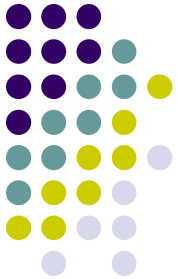


Gráfico de sectores (Pie-Charts)

Este gráfico es una gran herramienta para datos porcentuales tomadas sobre individuos o elementos. La función pie permite su construcción en R (probar help(pie)). Algunos de sus argumentos:

- **x**: vector de cantidades positivas proporcionales al tamaño de cada sector.
- **labels**: vector de etiquetas de los sectores
- **radius**: modifica el tamaño del diagrama.
- **col**: vector de colores, para rellenar los sectores del gráfico.
- **main**: para dar título al gráfico.

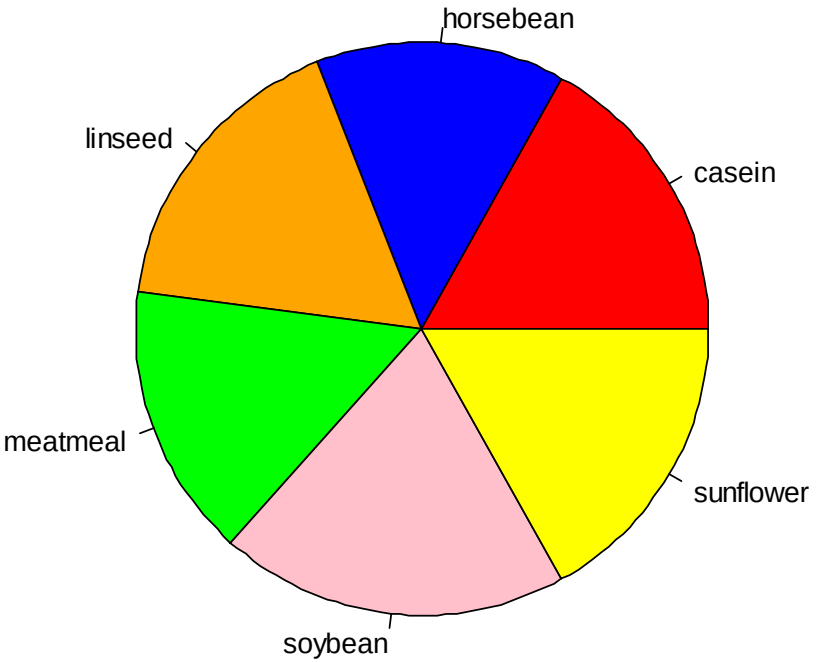
Práctica de Estadística en R

Paso 3. Representar gráficamente los datos

Gráfico de sectores

```
> table(chickwts$feed)

   casein horsebean    linseed  meatmeal    soybean sunflower
       12        10         12        11         14        12
> color<-c("red","blue","orange", "green","pink","yellow")
> pie(table(chickwts$feed),col=color)
```



Práctica de Estadística en R

Paso 3. Representar gráficamente los datos

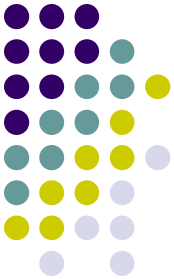


Gráfico de barras

La función barplot permite realizar gráficos de barras (help(barplot)). Algunos de sus argumentos:

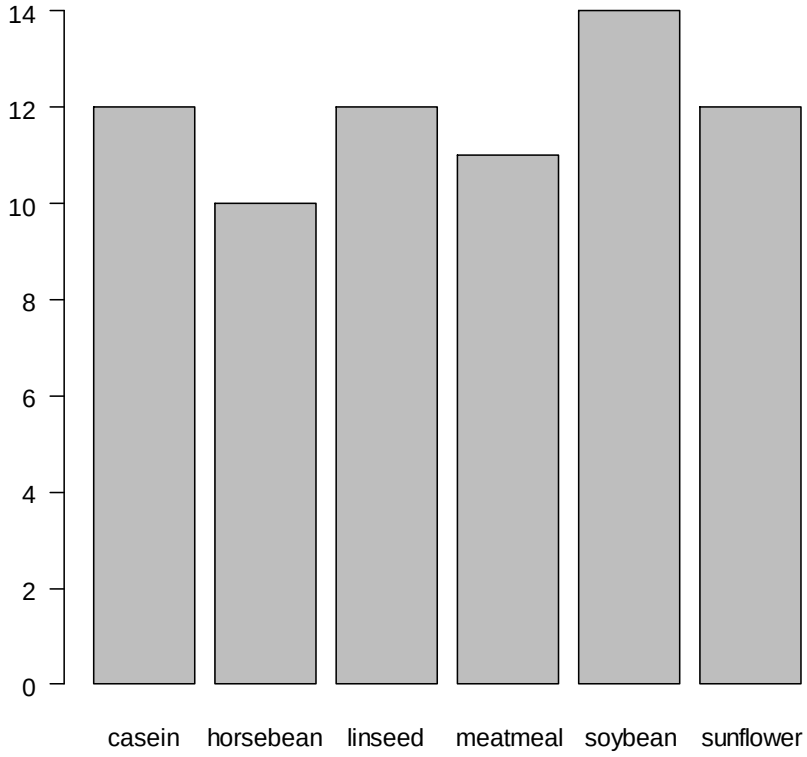
- **height**: Vector de frecuencias para cada valor
- **width**: Especifica mediante un vector el ancho de las barras.
- **space**: Fija el espacio entre barras.
- **names.arg**: Un vector de nombres para colocarlos debajo de cada barra o grupo de barras.
- **legend.text**: Un vector de texto para construir una leyenda para el gráfico..
- **beside**: Un valor lógico. “FALSE” indica barras apiladas y “TRUE” barras yuxtapuestas.
- **horiz**: Un valor lógico. “FALSE” indica barras verticales.
- **col**: Especifica un vector de colores para las barras.
- **xlim**: Delimita el rango de valores en el eje x.
- **ylim**: Delimita el rango de valores en el eje y.
- **axes**: Argumento lógico. Si es “TRUE”, dibuja el correspondiente eje.

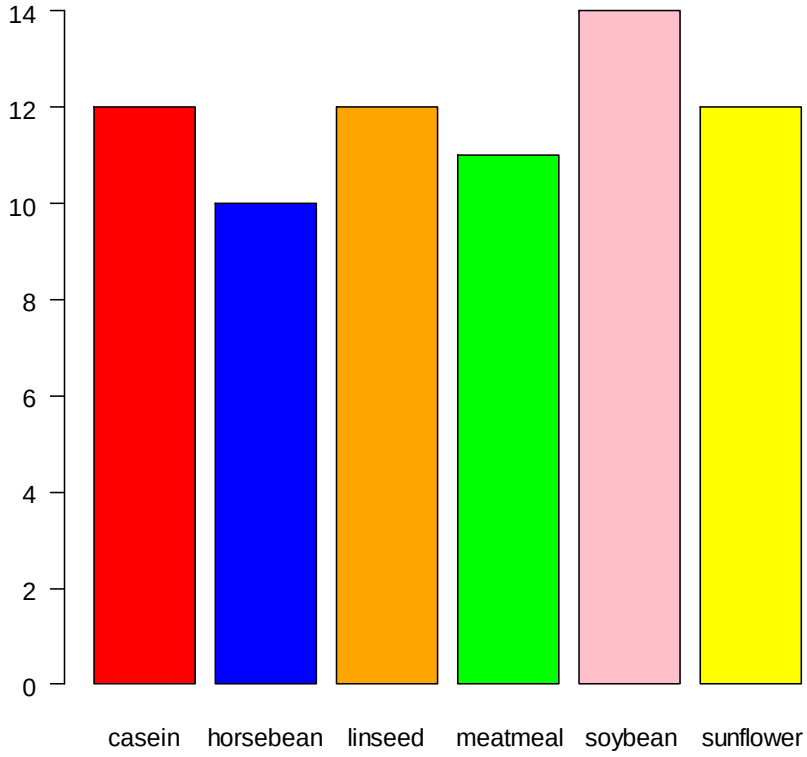
Práctica de Estadística en R

Paso 3. Representar gráficamente los datos

Gráfico de barras

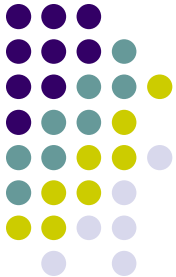
```
> barplot(table(chickwts$feed))
> barplot(table(chickwts$feed),col=color)
```





Práctica de Estadística en R

Paso 3. Representar gráficamente los datos



Gráficos de dispersión

Es posible obtener gráficos de dispersión (nubes de puntos) de hasta dos variables (bivariante) usando la función plot (help(plot)). Algunos de sus argumentos:

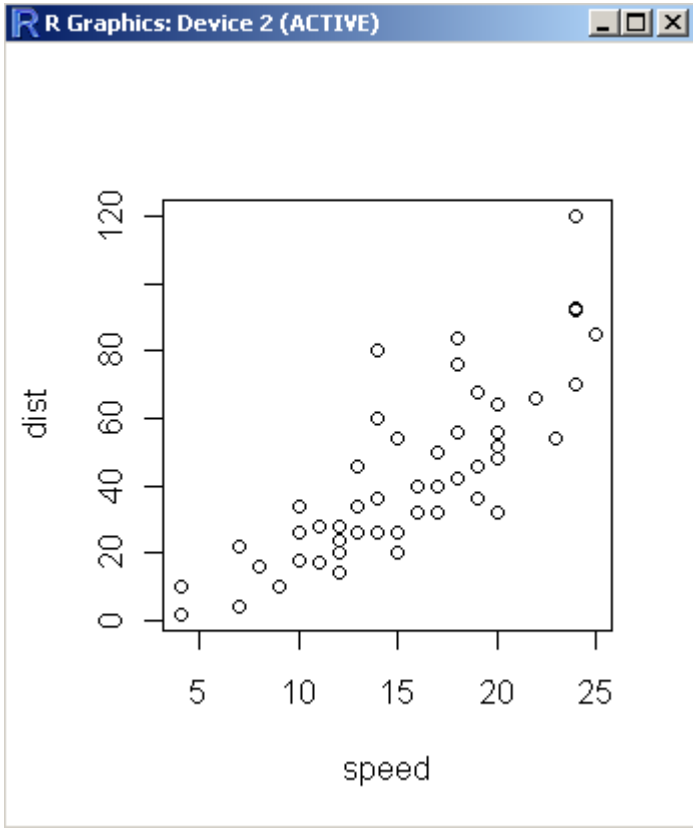
- **x**: vector que contiene las coordenadas (eje abscisas) o una hoja de datos
- **y**: ordenadas de los puntos (no es necesario si x es una hoja de datos)
- **type**: Especifica el tipo de gráfico, así :
  - "**p**" para puntos, lo cual es por defecto
  - "**l**" para trazar líneas entre los puntos
  - "**b**" para puntos y líneas
  - "**o**" para puntos y líneas superpuestos
  - "**h**" para un histograma pero en vez de rectángulos traza líneas verticales
  - ….

Práctica de Estadística en R

Paso 3. Representar gráficamente los datos

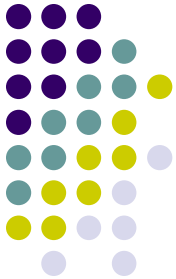
Gráficos de dispersión

```
> data(cars)
> plot(cars)
```



Práctica de Estadística en R

Paso 3. Representar gráficamente los datos



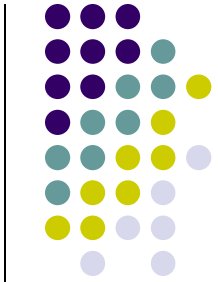
Matrices de dispersión

Las matrices de dispersión representan las relaciones entre pares de variables. Consisten en una matriz donde cada entrada presenta un gráfico de dispersión.
Es posible obtener esta representación gráfica usando la función pairs. Algunos de sus argumentos:

- **x**: Coordenadas de puntos.
- **labels**: Vector para identificar los nombres de las columnas.
- **panel**: Especifica una función para determinar los contenidos de los paneles o gráficos componentes. Por defecto es **points**
- **pch** y **col**, especifican símbolos y colores para los gráficos de dispersión.

Práctica de Estadística en R

Paso 3. Representar gráficamente los datos



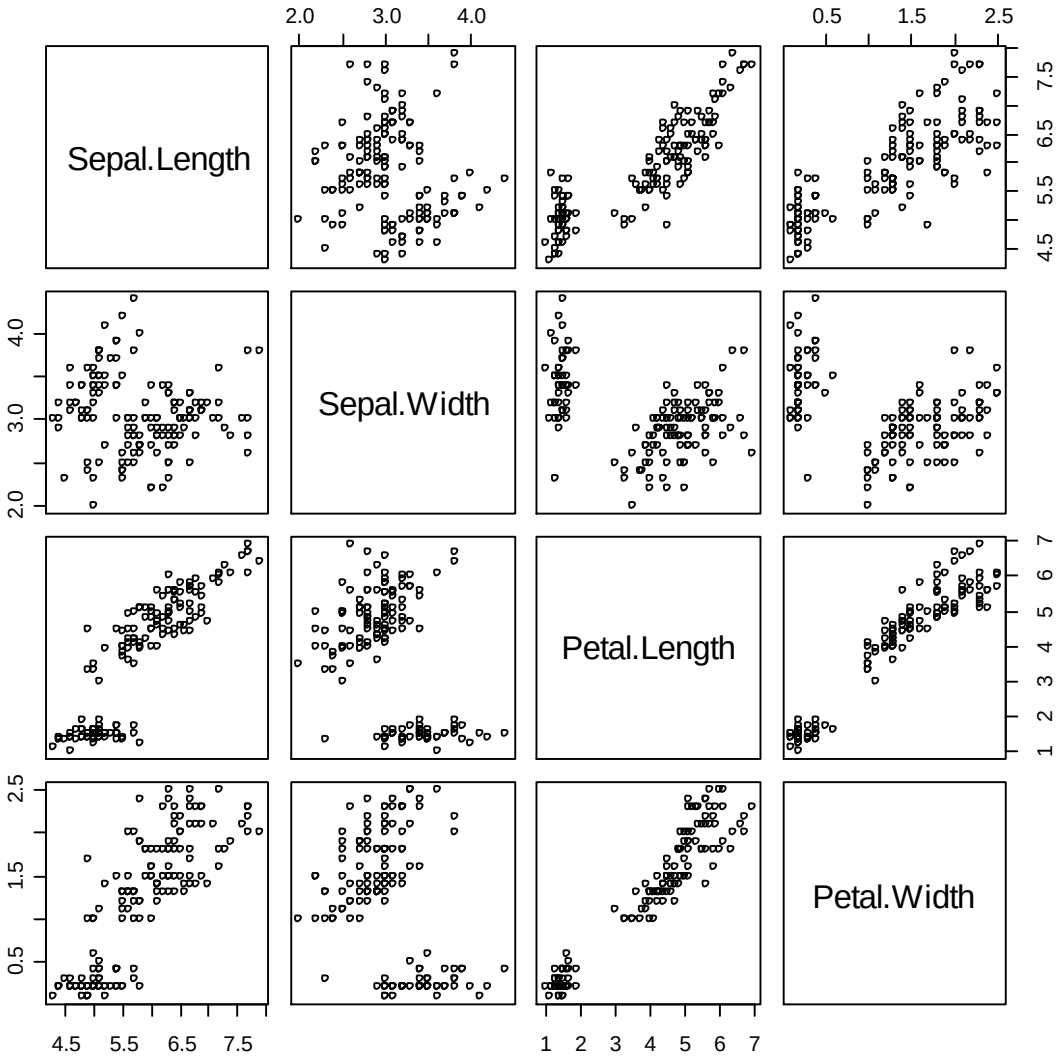
Matrices de dispersión

```
> data(iris)
> names(iris)
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
> datosIris<-data.frame(iris$Sepal.Length, iris$Sepal.Width,
+ iris$Petal.Length, iris$Petal.Width)
> varIris<-names(iris)
> varIris
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
> varIris<-varIris[1:4]
> varIris
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"
> pairs(datosIris, label=varIris)
```

Práctica de Estadística en R

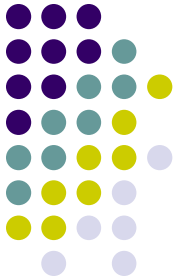
Paso 3. Representar gráficamente los datos

Matrices de dispersión



Práctica de Estadística en R

Paso 4. Descripción numérica



Algunas funciones disponibles:

Nota. Escribir help(nombrefunción) para una descripción de su sintaxis.

length(x), tamaño muestral

mean(x), media

median(x), mediana

sd(x), (cuasi)desviación típica

IQR(x), rango intercuartil

range(x), rango o recorrido (mínimo,máximo)

mad(x), mediana de las dif absolutas respecto la mediana

quantile(x, .25), percentil 25 (primer cuartil)

quantile(x, .50), percentil 50 (mediana)

quantile(x, .75), percentil 75 (tercer cuartil)

mean(x, trim=10/100), media recortada al 10%

mean(x, trim=5/100), media recortada al 5%

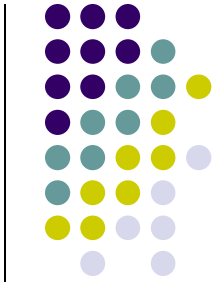
cov(x, y), covarianza

cor(x, y), coeficiente de correlación de Pearson

summary(x), resumen descriptivo elemental

Práctica de Estadística en R

Paso 4. Descripción numérica



```
> summary(chickwts)
     weight             feed
 Min.   :108.0   casein   :12
 1st Qu.:204.5   horsebean:10
 Median :258.0   linseed  :12
 Mean   :261.3   meatmeal :11
 3rd Qu.:323.5   soybean  :14
 Max.   :423.0   sunflower:12
> summary(cars)
     speed           dist
 Min.   : 4.0   Min.   :  2.00
 1st Qu.:12.0   1st Qu.: 26.00
 Median :15.0   Median : 36.00
 Mean   :15.4   Mean   : 42.98
 3rd Qu.:19.0   3rd Qu.: 56.00
 Max.   :25.0   Max.   :120.00
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
> |
```

Práctica de Estadística en R

Paso 5. Regresión lineal simple

Para ajustar un modelo lineal del tipo **y= a + b x** podemos utilizar la función **lm**. El resultado del ajuste es un objeto (de la clase lm). Si asignamos dicho resultado a "modelo", podemos visualizar su contenido escribiendo:

```
> modelo<-lm(cars$dist~cars$speed)
> modelo

Call:
lm(formula = cars$dist ~ cars$speed)

Coefficients:
(Intercept)   cars$speed
    -17.579        3.932
```

El objeto "modelo" contiene además de otra información la pendiente b=3.932 y la ordenada en el origen a=-17.579. Estos valores se pueden solicitar directamente escribiendo cualquiera de las dos sentencias siguientes:

```
coeficientes <- coef(modelo)
coeficientes <- modelo$coef
```

Con cualquiera de ellas crearíamos un vector con los coeficientes a y b.

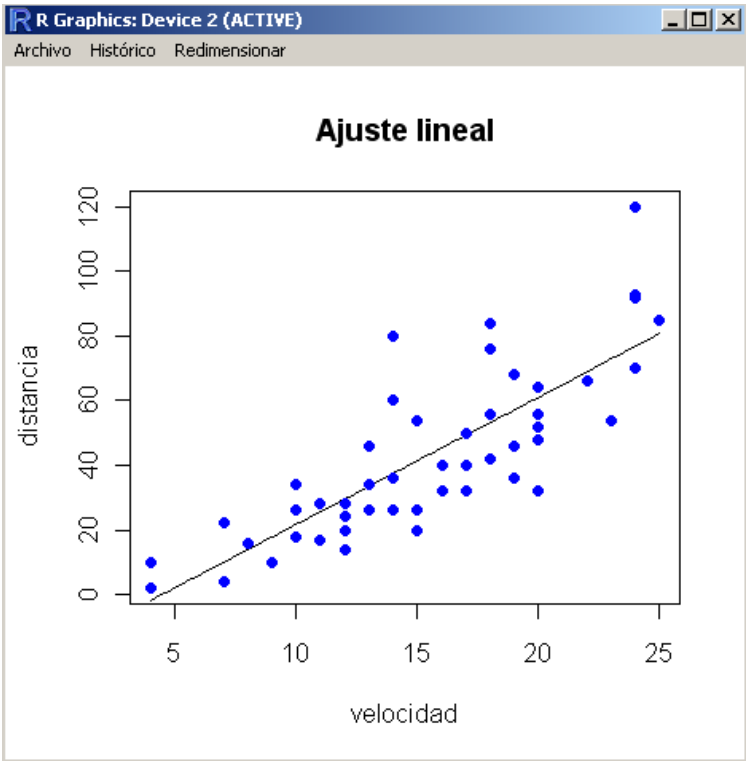
Escribir `help(lm)` para ver detalles sobre la función

Práctica de Estadística en R

Paso 5. Regresión lineal simple

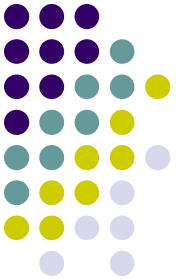
Para representar gráficamente el modelo ajustado podemos escribir:

```
> curve(modelo$coef[1]+modelo$coef[2]*x,ylab='distancia',
    xlab='velocidad',main='Ajuste lineal',xlim=range(cars$speed),
    ylim=range(cars$dist))
> points(cars$speed,cars$dist,pch=21,col='blue',bg='blue')
```



Práctica de Estadística en R

Paso 6. Problemas de contraste de hipótesis



Para resolver contrastes de hipótesis acerca de la media de una población normal (contraste de la t de Student) podemos utilizar la función **t.test**. Escribir help(t.test) para ver una descripción completa de su sintaxis.

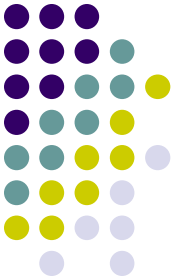
Para el caso de la variable **dist** de la hoja de datos **cars** se puede contrastar la hipótesis nula de que dicha distancia en media sea inferior a 60 escribiendo:

```
> t.test(cars$dist, alternative = "greater", mu = 60)
```

```
        One Sample t-test

data:  cars$dist
t = -4.6703, df = 49, p-value = 1
alternative hypothesis: true mean is greater than 60
95 percent confidence interval:
 36.87008      Inf
sample estimates:
mean of x
    42.98
```

Descriptiva y gráficos. Ejercicio propuesto



Supuesto práctico 1. El fichero de datos "poblacion.txt" recoge información básica demográfica y de consumo de una muestra ficticia de 6400 individuos. Se pide:

1. Leer el fichero de datos y almacenar su contenido en un data.frame con nombre "datos1".
2. Definir etiquetas para los 5 niveles de la variable nivel educativo ("educ") como sigue: 1, "Bachillerato incompleto"; 2 , "Bachillerato"; 3, "Universitarios parciales"; 4, "Universitarios"; 5, "Post-universitarios".
3. Obtener una tabla de frecuencias de la variable "educ". La tabla será un objeto de tipo data.frame con 5 columnas: modalidades de la variable, frecuencia absoluta, frecuencia relativa, absoluta acumulada y relativa acumulada.
4. Representar gráficamente la variable "educ" mediante un diagrama de barras y un diagrama de sectores. Ambos gráficos deben aparecen en la misma ventana gráfica dividida en dos filas y una columna (usar para ello la función `par()` especificando por ejemplo `mfrow(c(2,1))`.
5. Representar gráficamente la variable "edad" mediante un histograma y un gráfico de cajas. El histograma deberá realizarse definiendo como puntos de corte los percentiles 0%, 10%, 20%,…,90% y 100% de la variable. Y de nuevo ambos gráficos deben aparecer en una misma ventana gráfica.

Nota: Trabajar sobre los gráficos usando opciones que permitan mejorar su aspecto (etiquetas de los ejes, títulos, colores, etc.

Regresión. Ejercicio propuesto

Supuesto práctico 2. Como resultado de una nueva política empresarial, se ha aumentado progresivamente la inversión en formación de los empleados en una multinacional de software. Se sospecha que este incremento en inversión ha tenido gran importancia en los beneficios de la empresa. En la tabla adjunta se recogen datos correspondientes a los gastos en inversión (en millones de euros) y a los beneficios brutos de la empresa (en millones de euros) en los últimos diez meses. Se pide:

1. Crear un objeto de tipo data.frame con nombre “datos2” con los datos de la tabla.
2. Representar un diagrama diagrama de dispersión de la variable inversión (eje horizontal) frente a la variable beneficios (eje vertical).
3. Ajustar un modelo lineal a los datos que permita predecir la variable beneficios en función de la variable inversión.
4. Representar gráficamente el modelo ajustado (superponiendo la recta de regresión al diagrama de dispersión obtenido en el apartado 2).
5. Usando el modelo ajustado predecir los beneficios esperados para un mes donde la inversión en formación es de 1.5 millones de euros.

Inversión	Beneficios
,2	25,3
,1	26,7
,7	31,4
,8	33,5
1,1	39,7
1,3	40,6
2,4	45,5
2,9	56,8
3,5	75,4
3,9	97,2