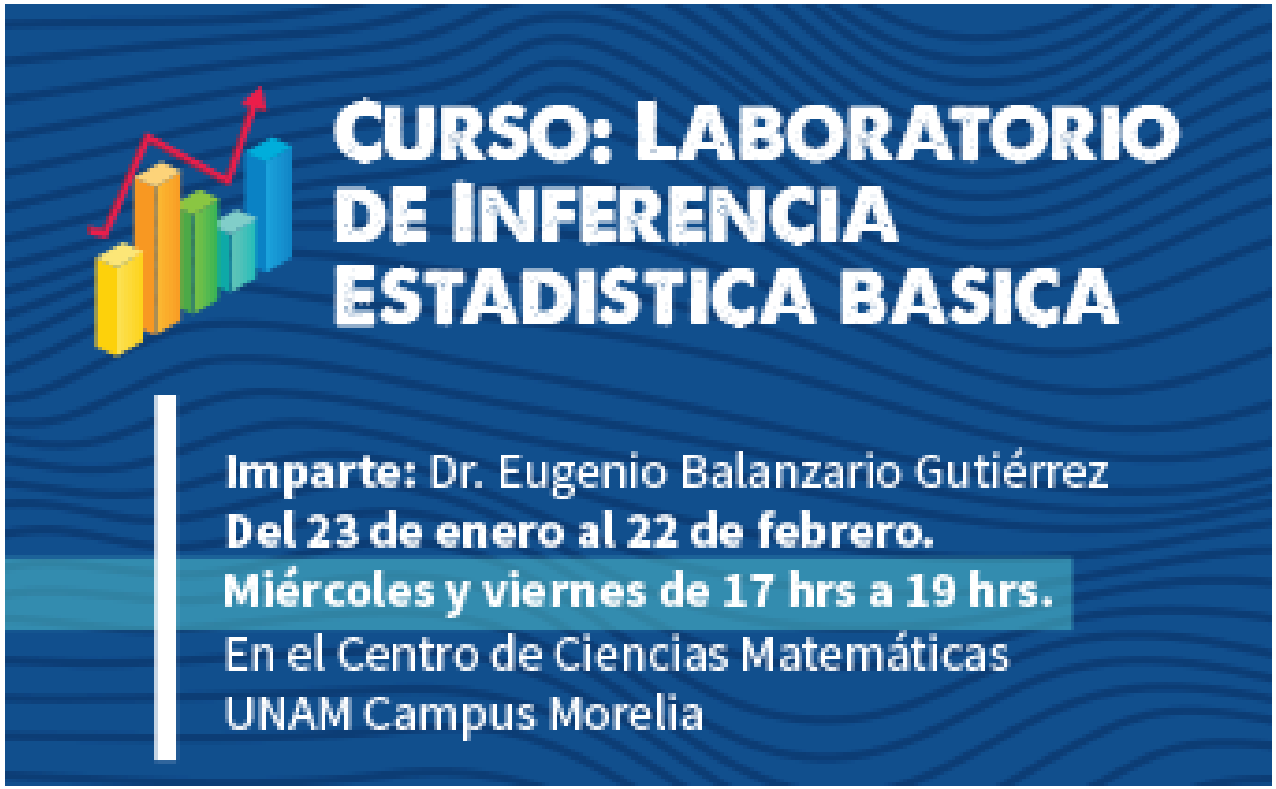
Laboratorio de inferencia estadística básica



Plan de trabajo

Este laboratorio se ofrece con el propósito de capacitar al oyente para realizar inferencias básicas sobre conjuntos de datos estadísticos con la ayuda de las tecnologías de computación disponibles en la actualidad. Para este fin, se expondrán los conceptos fundamentales sobre los siguientes tópicos.

- Introducción al lenguaje R (2 horas).
- Distribuciones de probabilidad que dependen de uno o más parámetros (2 horas).
- Métodos de estimación de parámetros, puntual y por intervalos (2 horas).
- El concepto de prueba de hipótesis estadística y ejemplos de las pruebas más comunes. En particular las pruebas de bondad de ajuste (6 horas).
- Regresión lineal simple (6 horas).
- Análisis de varianza (2 horas).

En total el laboratorio se desarrollará en alrededor de 20 horas de trabajo en grupo.

Lenguaje de programación R. Comandos básicos

El lenguaje R es sensible a las mayúsculas: no es lo mismo "variable1" que "Variable1".

Para interrumpir un cálculo extenso, oprimir Esc.

>q() # para salir de R

>x <- 5 # asigna el valor "5" a la variable x

>ls() # da la lista de las variables, u objetos, definidos por el usuario

>rm("x") # elimina la variable x

>rm(list=ls()) # elimina todos los objetos definidos por el usuario

>getwd() # muestra el directorio en uso

>setwd("/media/euba/ADATA UFD/Diplomado/Programas")
># cambia el directorio en uso

>dir() # muestra los archivos en el directorio en uso

Comandos básicos

>history(10) # nos da los últimos 10 comandos ejecutados

>savehistory("borrar.txt") # guarda la lista de comandos ejecutados en el archivo "borrar.txt"

>loadhistory("borrar.txt") # carga el archivo "borrar.txt"

R puede ejecutar un conjunto de instrucciones leyéndolas desde un archivo externo. Por ejemplo, considere el archivo Sumar.r que se muestra a continuación:

```
# suma
a <- 1
b <- 2
print(a+b)
```

Para ejecutar el archivo "Sumar.r" se usa el comando source("Sumar.r").

Ambiente de cálculo

```
>y <- c(1, 2, 3, 4, 5, 6, 7, 8, 9) # define el vector y

>y # despliega lo siguiente:
[1] 1 2 3 4 5 6 7 8 9

>m <- matrix(y, 3, 3) # define la matriz de 3x3
     [,1] [,2] [,3]
[,1]    1    2    3
[,2]    4    5    6
[,3]    7    8    9

>u <- matrix(1, 3, 1) # define la matriz u de 3x1
     [,1]
[1,]    1
[2,]    1
[3,]    1

>mean(y) # calcula la media de las entradas en el vector y
```

Ambiente de cálculo

En la siguiente tabla se reportan los comandos para calcular otros estadísticos básicos.

length(y)	número de entradas en el vector y
sum(y)	suma de las entradas de y
sort(y)	ordena de menor a mayor
min(y), max(y)	obtiene el mínimo y el máximo de y
mean(y)	media del vector y
median(y)	mediana del vector y
sd(y)	desviación estándar del vector y
var(y)	varianza del vector y
cor(x, y)	correlación entre x, y
cov(x, y)	covarianza entre x, y

Ambiente de cálculo

En la siguiente tabla se reportan algunas operaciones matemáticas.

comando	operación	comando	operación
+	suma	trunc(x)	elimina decimales
-	resta	round(x, digits=0)	redondea x
*	multiplicación	signif(x, digits=3)	redondea x
/	división	log(x)	logaritmo natural
x%%y	x módulo y	log(x, base=2)	log base 2
abs(x)	valor absoluto	log10(x)	log base 10 de x
sqrt(x)	raíz cuadrada	exp(x)	exponencial de x
ceiling(x)	función techo	%*%	multiplica matrices
floor(x)	punción piso	n:m	genera n,n+1,$\cdots$,m

Ejemplos.

```
>y <- c(1:5) # sirve para generar el vector
[1] 1 2 3 4 5
```

```
>seq(from=1, to=13, by=3) # da como resultado
[1] 1 4 7 10 13
```

Si x es un vector, entonces x[i] es la i-ésima entrada de x. Por ejemplo:

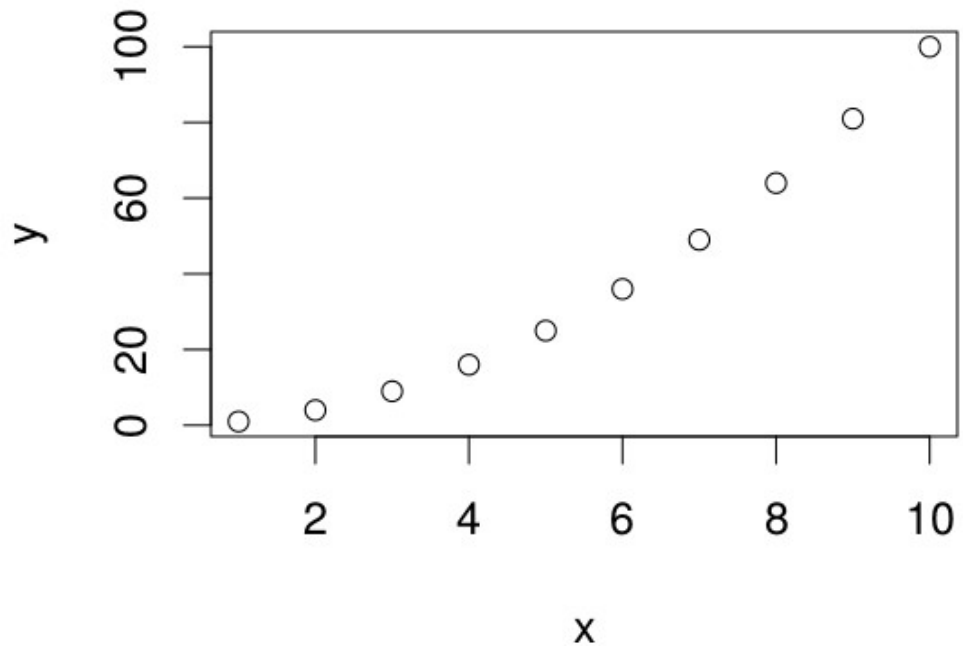
```
>x <- c(4, 6, 2, 4, 8, 9, 1, 5)
```

```
>x[3]
[1] 2
```

```
>x[1:3] # da como resultado
[1] 4 6 2
```

Gráficas en R

```
>dev.new(width=6, height=5)
># especifica las dimensiones de la gráfica
>par(cex=1.5) # especifica el tamaño de los caracteres de la gráfica
>x <- 1:10
>y <- x^2
>plot(x, y) # produce la gráfica:
```



La gráfica anterior se puede imprimir en un archivo pdf mediante las siguientes instrucciones:
>pdf("rplot.pdf", width=6, height=5)
>par(cex=1.5)
>plot(x, y)
>dev.off()

Las funciones para graficar en R tienen varios parámetros. El comando "par" sirve para especificar de manera global los parámetros de las gráficas que se realizarán en la sesión. Por ejemplo, >par(cex=1.5) sirve para definir el tamaño de los caracteres de las gráficas.

- "xlab" y "ylab" sirven para especificar las leyendas de los ejes horizontal y vertical.
  >plot(x, y, xlab="equis", ylab="cuadrado")
- "col" y "bg" cambian el color de la gráfica y el color del fondo de la gráfica.
  >par(bg="yellow")
  >plot(x, y, col="red")

- “pch” para seleccionar el símbolo para los puntos de la gráfica.
  >plot(x, y, pch=1:10)
- “lty” y “lwd” cambian el tipo de linea y el grueso de la linea respectivamente. “lines” sirve para añadir a la gráfica una curva a trazo continuo.
  >plot(x, y, lwd=3)
  >lines(x, x^2, lty=5)
- “cex” tamaño del texto y puntos de la gráfica.
  >plot(x, y, cex=2:3)
- “main” es el título de la gráfica.
  >plot(x, y, main=“hola”)
- “ps” sirve para especificar el tamaño del texto dentro de la gráfica.
  >par(ps=10, cex=1.5, cex.main=2)
  >plot(x, y, cex=2:3, main=“Cuadrado”)

- “fg” especifica el color del marco de la gráfica.
  >plot(x, y, fg=“blue”)
- “xlim” y “ylim” especifica el rango en el eje de la x y en el eje de las y, respectivamente.
  >plot(x, y, ylim=c(-10, 110), xlim=c(-1, 12))
- “text” sirve para añadir texto a la gráfica.
  >text(4, 20, “(4, 16)”)
- “mtext” añade texto en los márgenes.
  >mtext(“aquí”, side=4)
- “type” especifica el tipo de gráfica.
  >plot(x, y, type = “b”)

La gráfica puede ser de uno de los tipos que se especifican en la siguiente tabla.

“p”	puntos
“l” (ele)	lineas
“c”	puntos vacíos con lineas
“o”	puntos con lineas sobrepuestas
“s” o “S”	dos tipos de función escalón
“h”	tipo histograma
“n”	ni puntos ni lineas

Ejemplos adicionales.

```
>plot(x, y, pch=21, lwd=2, col="red", bg="blue")
>plot(x, y, pch=c("a", "b"), col=c("red", "blue"))
>par(cex.lab=2, cex.main=2)
># tamaño de las leyendas en los ejes y título
>plot(x, y, pch=c("a", "b"), col=c("red", "blue"),
+main="cuadrado")
```

Generación de números aleatorios

>set.seed(777) # se inicializa el generador de números aleatorios

Si el generador de números aleatorios se inicia con un mismo valor cada vez que se corra el comando, entonces la sucesión de números aleatorios permanecerá sin cambiar.

>runif(5) # da como resultado:
[1] 0.6878574 0.4921926 0.3451156 0.9950499 0.6952672

R maneja distintos tipos de variables aleatorias. En la siguiente tabla se reportan los nombres de algunas de las variables aleatorias de uso común.

Variable	Nombre en R	Parámetros
Binomial	binom	size, prob
Geométrica	geom	p
Poisson	pois	lambda
Uniforme en (a,b)	unif	min, max
Exponencial	exp	rate
Gama	gamma	shape, scale
Logística	logis	location, scale
Normal	norm	mean, sd
Ji cuadrada	chisq	df
T de Student	t	df
F	f	df1, df2

Los nombres de las variables aleatorias se usan en conjunto con una raíz que indica la función a ejecutar.

dnorm	función de densidad normal
pnorm	función de distribución normal
qnorm	cuantiles de la distribución normal
rnorm	números aleatorios normales

>rnorm(5, 1, 0.01) # da como resultado:
[1] 0.9979378 0.9962103 0.9969574 1.0005416 0.9811907

>rbinom(5, size=10, prob=0.5) # da como resultado:
[1] 5 5 9 6 7

La siguiente instrucción sirve para calcular la probabilidad de que una variable aleatoria $X \sim$ Binomial(10, 5) sea igual a 7.

>dbinom(7, size=10, prob=0.5)
[1] 0.1171875

Nota: R reconoce a “dbinom” como a una función de densidad, aún cuando en sentido estricto se trata de una función de distribución de masa.

Con la siguiente instrucción se calcula la probabilidad de que $X \sim$ Binomial(10, 5) sea menor o igual a 7.

>pbinom(7, size=10, prob=0.5) # da como resultado:
[1] 0.9453125

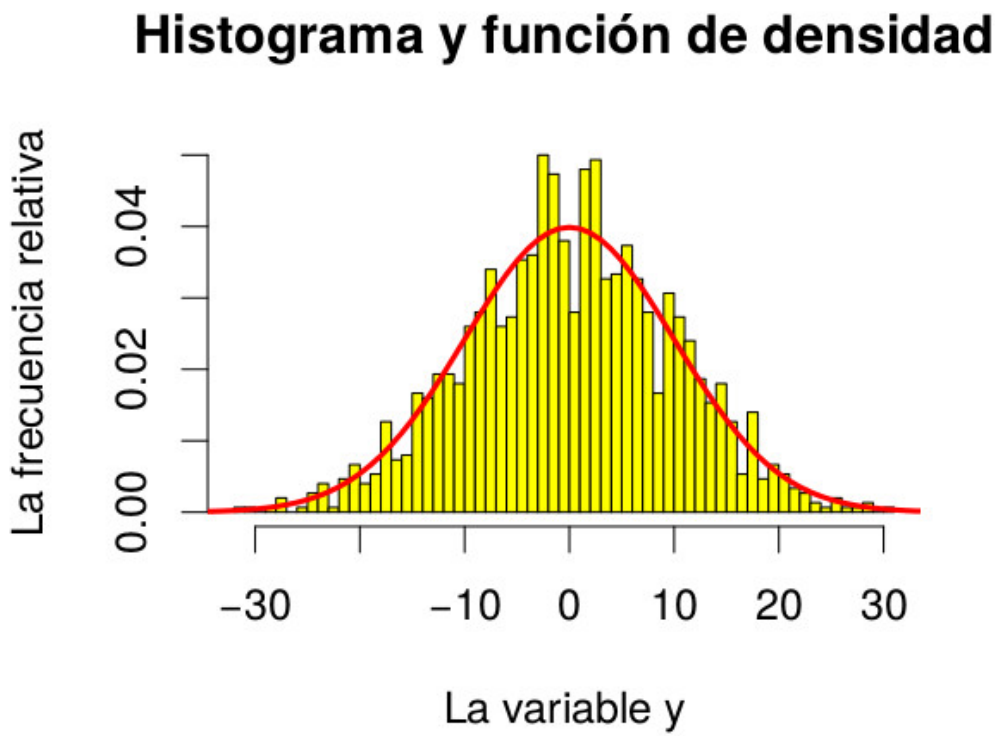
>qnorm(0.99, mean = 0, sd = 1) # para calcular los cuantiles, dando como resultado:
[1] 2.326348

Se puede obtener más información sobre las distribuciones de probabilidad en el contexto de R con las funciones de ayuda: ?Normal, ?Binomial, ?TDist, ?Chi-squared, etcétera.

Se sale de la página de ayuda con “q”.

Construcción de un histograma

El archivo Histograma.Normal.r contiene los comandos necesarios para generar la siguiente figura.



En particular, el comando hist(x) genera el histograma de los datos en el vector x.

Construcción de un histograma

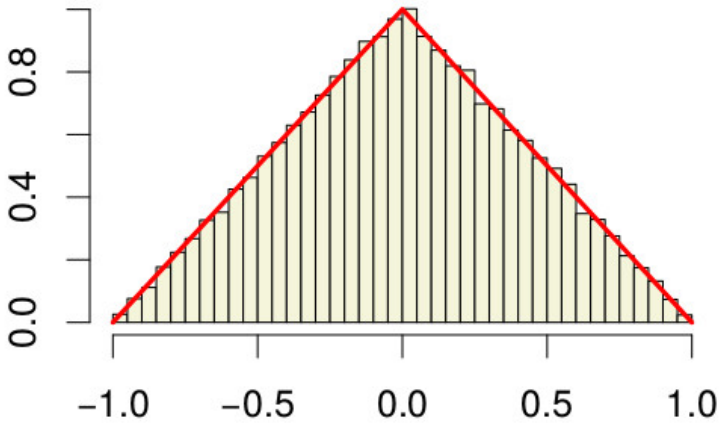
```
dev.new(width=6, height=5)
par(cex=1.5)
y <- rnorm(1500, 0, 10)
hist(y, breaks=50, col="yellow", freq=FALSE)
x <- seq(from=-40, to=40, by=1)
w <- dnorm(x, mean=0, sd=10)
lines(x, w, lwd=3, col=red")
```

Ley de los grandes números

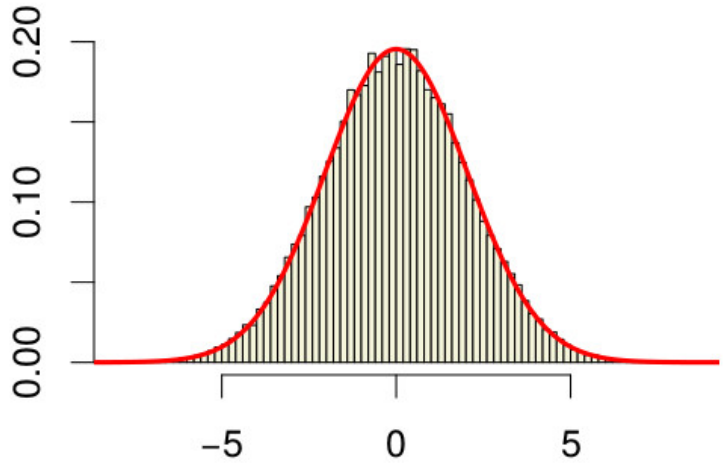
En el histograma anterior, el tamaño de la muestra fue de 1 500.

Por la ley de los grandes números, cuando el tamaño de la muestra se incrementa, entonces el histograma tiende a coincidir con más exactitud con la función de densidad de la que se tomó la muestra.

Teorema del límite central



Suma de 2 uniformes



Suma de 50 uniformes

Si $S_n = X_1 + \cdots + X_n$, $\mu = \mathrm{E}(X)$ y $\sigma^2 = \mathrm{var}(X)$, entonces

$$\lim_{n\to\infty} \mathrm{P}\left\{ \frac{S_n - n\mu}{\sigma\sqrt{n}} \leq x \right\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}y^2}\, dy.$$

Ver el archivo Teorema.Central.del.Límite.r .

Teorema del límite central

```
fun <- function(x) {
    if(x<=0) {aux <- 1+x}
    if(x>0) {aux <- 1-x}
    return(aux)
} # la función de densidad triangular

a <- 50000
b <- 2
x <- runif(a*b, -0.5, 0.5)
m <- matrix(x, a, b)
u <- matrix(1, b, 1)
y <- m %*% u # multiplicación de matrices
hist(y, breaks=50, col="beige", freq=FALSE)
z <- seq(from=-1, to=1, by=0.1)
w <- sapply(z, fun)
lines(z, w, lwd=3, col="red")
```

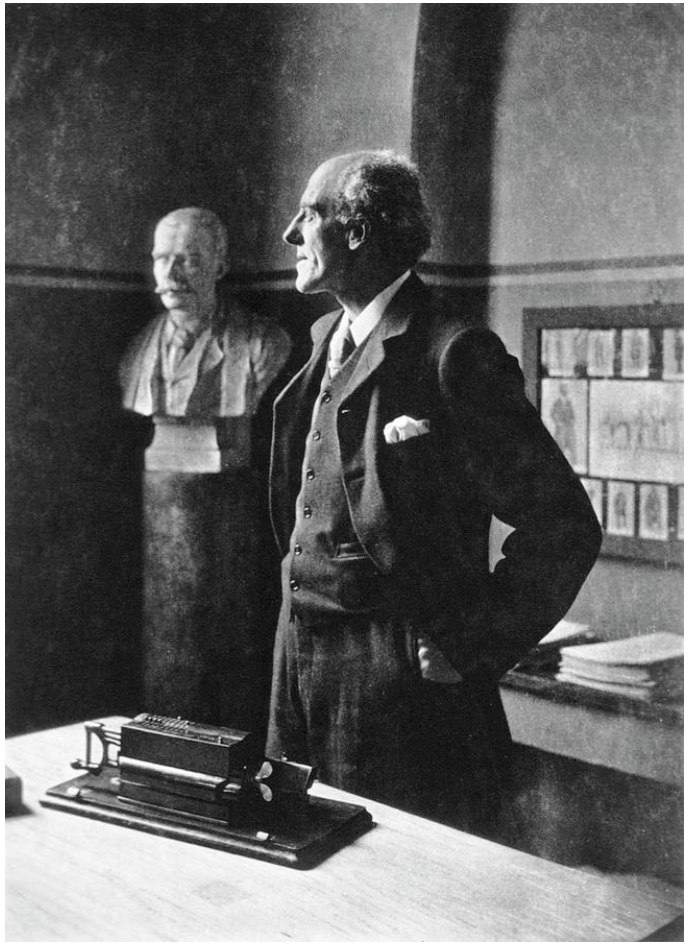
Teorema del límite central

"As a rule of thumb, the sample size must be at least 30 for the central limit theorem to hold true". Charles Wheelan

"I know of scarcely anything so apt to impress the imagination as the wonderful form of cosmic order expressed by the law of frequency of error. The law would have been personified by the Greeks if they had known of it". Francis Galton

Distribuciones de probabilidad que dependen de uno o más parámetros

El científico realiza mediciones y las usa para encontrar fórmulas matemáticas que describan la naturaleza. Karl Pearson (1857-1936) tuvo la idea de que cuando se realiza un número grande de mediciones, lo que se obtiene es una distribución de valores. Esta distribución de valores se describe mediante una fórmula matemática que depende de uno o más parámetros.

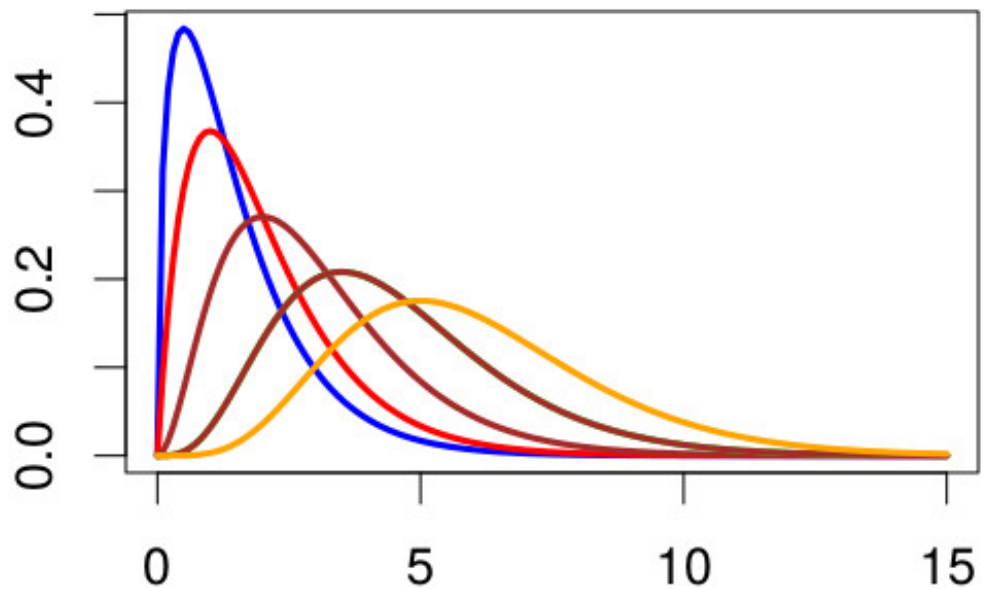


La distribución Gama y sus parámetros

La función de densidad de una variable Gamma está dada por

$$f(x) = \frac{1}{\theta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-x/\theta}$$

en donde α es el parámetro de forma y θ el parámetro de escala. El la figura, α toma los valores 1.5, 2, 3, 4.5, 6. El parámetro de escala θ se mantuvo constante e igual a 1. Ver archivo Gama.1.r .



Método del rechazo para generar muestras

El método del rechazo nos permite simular la realización de una variable aleatoria definida por una función de densidad arbitraria. Esta función puede depender de uno o más parámetros. Ver el archivo Método.del.Rechazo.r .

```
rejectionK <- function(fx, a, b, K) {
    # simula una variable con función de densidad fx
    # supone que fx es 0 fuera de [1, b] y acatada por K
    while (TRUE) {
        x <- runif(1, a, b)
        y <- runif(1, 0, K)
        if (y < fx(x)) return(x)
    }
}
```

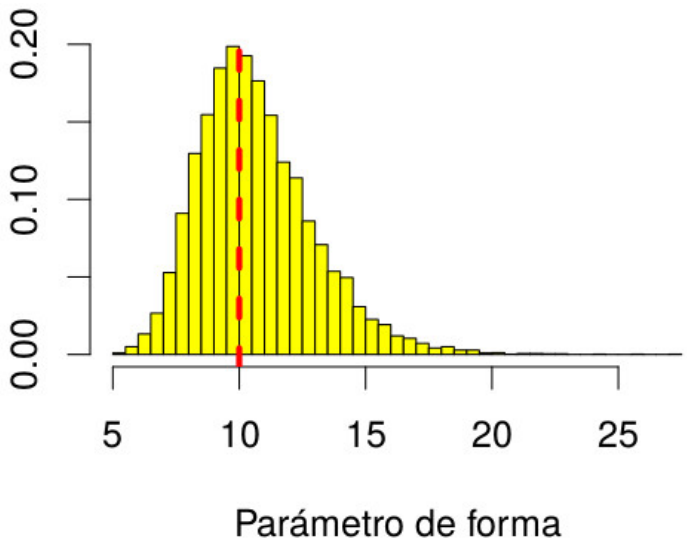
1. Genere X$\sim$Uni[a,b] y Y$\sim$Uni[0,K], variables independientes.
2. Si Y < fx(X) entonces se reporta X, en otro caso, ir al paso 1.

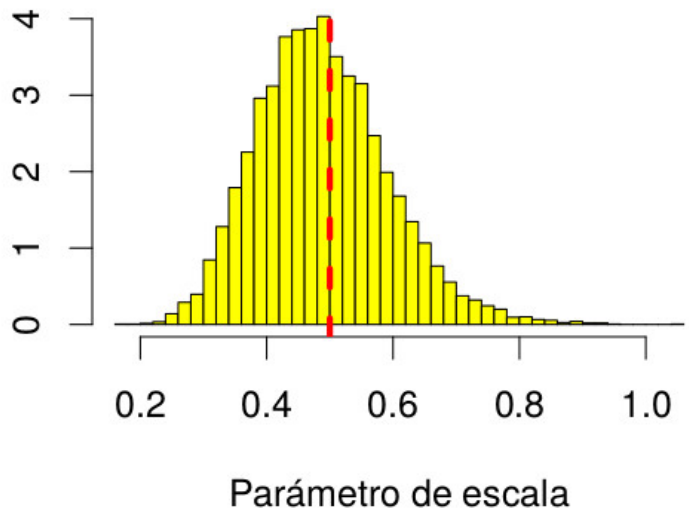
Método de momentos

Los estimadores por el método de los momentos están dados por

$$\hat{\alpha} = \frac{m_1^2}{m_2 - m_1^2} \quad \text{y} \quad \hat{\theta} = \frac{m_2 - m_1^2}{m_1} \quad \text{en donde} \quad m_k = \frac{1}{n}\sum_{i=1}^{n} X_i^k.$$

En la figura se muestra el histograma de 10 000 estimaciones realizadas a partir de muestra de tamaño 50 cada una. La linea punteada en rojo muestra el valor real del parámetro.





Ver archivo Gama.Momentos.r .

Método de momentos

```
muestra <- 200 # tamaño de la muestra
tot <- 10000 # número de muestras
a <- 10 # parámetro de forma
t <- 0.5 # parámetro de escala
t.hat <- c() # se inicializa esta variable
a.hat <- c() # se inicializa esta variable
for(i in 1:tot) {
    m <- rgamma(muestra, shape=a, scale=t)
    m1 <- mean(m)
    m2 <- sum(m^2)/muestra
    t.hat[i] <- (m2-m1^2)/m1
    a.hat[i] <- m1^2/(m2-m1^2)
}
```

>hist(a.hat) # para ver la salida del código anterior
>hist(t.hat) # para ver la salida del código anterior

Método de momentos

La siguiente tabla contiene algunos estimadores según el método de los momentos. Note que $m_1 = \bar{X}$.

Uniforme$[0, a]$	$\hat{a} = 2\bar{X}$
Bernoulli(p)	$\hat{p} = \bar{X}$
Binomial(n, p)	$\hat{n} = m_1^2/(m_1 + m_1^2 - m_2)$, $\hat{p} = m_1/\hat{n}$
Poisson(α)	$\hat{\lambda} = \bar{X}$
Normal(μ, σ^2)	$\hat{\mu} = \bar{X}$, $\hat{\sigma}^2 = m_2 - m_1^2$

La distribución logarítmica tiene distribución de masa y media dadas respectivamente por

$$f(k) = \frac{-p^k}{k \log(1-p)} \qquad \text{y} \qquad M_1(p) = \frac{-p}{(1-p)\log(1-p)}.$$

Desafortunadamente no es posible resolver la ecuación $M_1(p) = m_1$ para obtener $\hat{p}$ en función de m_1. Sin embargo, esta ecuación se puede resolver numéricamente. Para este fin, se define la siguiente función.

```
fun <- function(p) {
    a <- (1-p)*log(1-p)
    return(-p/a-m1)
}
```

Con el siguiente comando se obtiene el valor numérico para $\hat{p}$ una vez que m_1 es conocido.

```
>uniroot(fun, c(0.01, 0.9))$root
```

Ver el archivo Distribución.Logarítmica.r . En este archivo también se se programaron los comandos para realizar la simulación de la variable aleatoria.

Método de los momentos

```
fd <- function(n=k, probabilidad=p) { # la distribución de masa
    a <- -probabilidad^n
    b <- n*log(1-probabilidad)
return(a/b)
}

sim <- function(p) { # función que simula una variable logarítmica
    rnd <- runif(1)
    k <- 0
    sum <- 0
    while(rnd>sum) {
        k <- k+1
        sum <- sum+fd(k, p)
    }
    return(k)
}
```

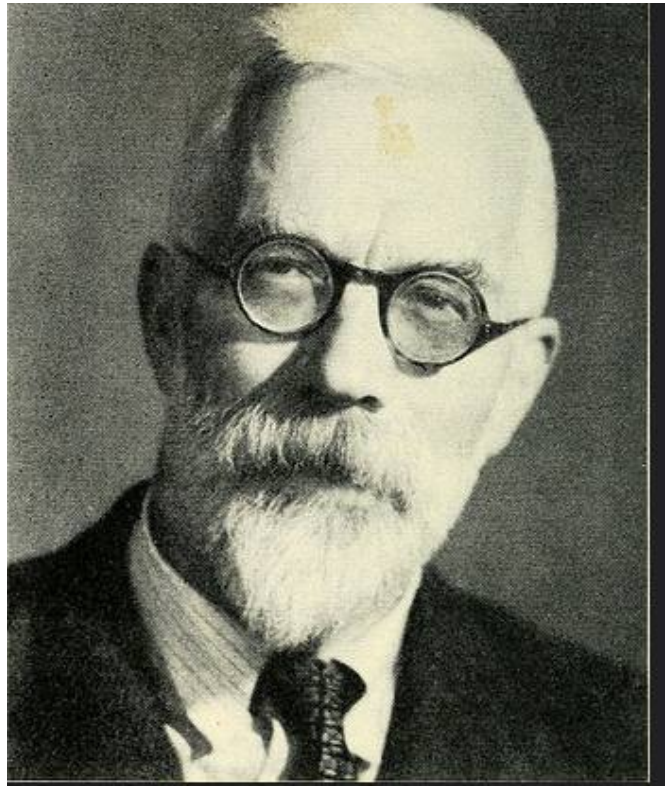
Método de los momentos

```
# se generan tot valores de la variable aleatoria
>tot <- 50000
>u <- matrix(0.35, tot, 1)
>z <- sapply(u, sim)

# se utilizan los valores generados para estimar el parámetro p
>print(uniroot(fun, c(0.01, 0.99), mean(z))$root)
```

Máxima verosimilitud

Ronald Fisher (1890-1962) se dio cuenta de que los métodos que Karl Pearson había estado usando para estimar los parámetros de una distribución, producían estadísticos que no necesariamente eran consistentes y que con frecuencia presentaban un sesgo. Para producir estadísticos consistentes y eficientes, Fisher propuso algo que él denominó «estimadores de máxima verosimilitud».



Máxima verosimilitud

La función de verosimilitud de una muestra $X_1, \cdots, X_n$, está dada por

$$L(\theta) = f(x_1|\theta) f(x_2|\theta) \cdots f(x_n|\theta).$$

El estimador máximo verosímil $\hat{\theta}$ del parámetro θ, es aquel valor para el cual

$$\ell(\theta) = \log L(\theta)$$

es máximo. Si el tamaño n de la muestra es grande, entonces $\hat{\theta}$ tiene una distribución aproximadamente normal con media θ y varianza

$$\text{var}(\hat{\theta}_n) = \frac{1}{n \, \text{E}\left[\frac{d}{d\theta} \log f(X|\theta)\right]^2}.$$

Note que $\text{var}(\hat{\theta}_n) \to 0$ cuando $n \to \infty$. Ver el archivo EMV.Exponencial.r .

Máxima verosimilitud

```
LL <- function(r) { # el log de la función de verosimilitud
    R <- dexp(x, r)
    -sum(log(R))
}
```

>rt <- 2.5; N <- 1250; x <- rexp(N, rate=rt)
>optim(rt, LL, method = "Brent", lower=0.001, upper=100)$par

```
tot <- 2000
a <- c()
for(i in 1:tot) {
    x <- rexp(N, rate=rt)
    a[i] <- optim(rt, LL, method = "Brent", lower=0.001,
                  upper=100)$par
    if(i%%100==0) {print(i)}
}
```

>hist(a)

Máxima verosimilitud: ajuste de una distribución

Con el comando fitdistr de la librería MASS es posible estimar los parámetros de una distribución mediante el método de máxima verosimilitud. He aquí algunos ejemplos.

```
>x <- rnorm(500, mean=0, sd=1)
>library(MASS)
>fitdistr(x, "normal")
```

```
>x <- rgamma(100, shape=10, scale=0.5)
>fitdistr(x, "gamma")
```

```
>x <- rgeom(100, p=0.1)
>fitdistr(x, "geometric")
```

Nota: En el último ejemplo, el comando fitdistr se ejecutó con el nombre "geometric" y no con el nombre "geom". Consulte la siguiente página.

https://stat.ethz.ch/R-manual/R-devel/library/MASS/html/fitdistr.html

Extracción de información en objetos

Considere nuevamente la tarea del ajuste de una distribución

```
>x <- rnorm(500, mean=0, sd=1)
>library(MASS)
>fit <- fitdistr(x, "normal")
```

El comando summary(fit) despliega el contenido en el objeto fit.

	Length	Class	Mode
estimate	2	-none-	numeric
sd	2	-none-	numeric
vcov	4	-none-	numeric
n	1	-none-	numeric
loglik	1	-none-	numeric

Mediante fit$estimate, o fit[[1]], se obtienen las estimaciones obtenidas.

Mediante fit$estimate[[1]], o fit[[1]][[1]], se obtiene el valor estimado para el primero de los parámetros.

Error cuadrático medio (ECM)

Un estimador $\hat{\theta}_1$ es mejor que $\hat{\theta}_2$ si $\text{ECM}(\hat{\theta}_1) < \text{ECM}(\hat{\theta}_2)$.

Si $\hat{\theta}$ es insesgado, entonces $\text{ECM}(\hat{\theta}) = \text{var}(\hat{\theta})$.

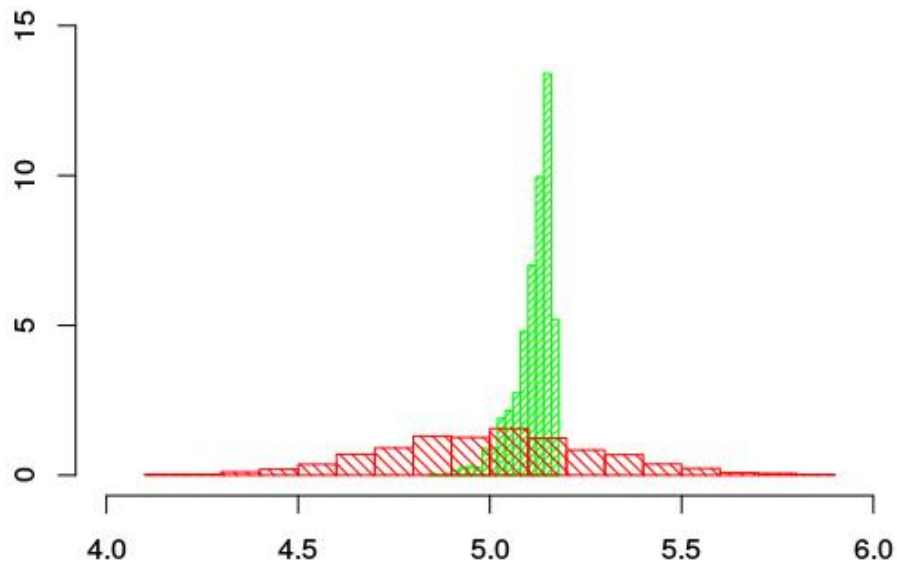
Ejemplo. La población: $X \sim \text{Uniforme}(0, \theta)$. Los estimadores:

$$\hat{\theta}_1 = 2\bar{X} \qquad \text{y} \qquad \hat{\theta}_2 = \frac{n+1}{n} \text{máx}(X_1, \cdots, X_n).$$

Las varianzas de los estimadores:

$$\text{var}(\hat{\theta}_2) = \frac{\theta^2}{n(n+2)},$$

$$\text{var}(\hat{\theta}_1) = \frac{\theta^2}{3n}.$$



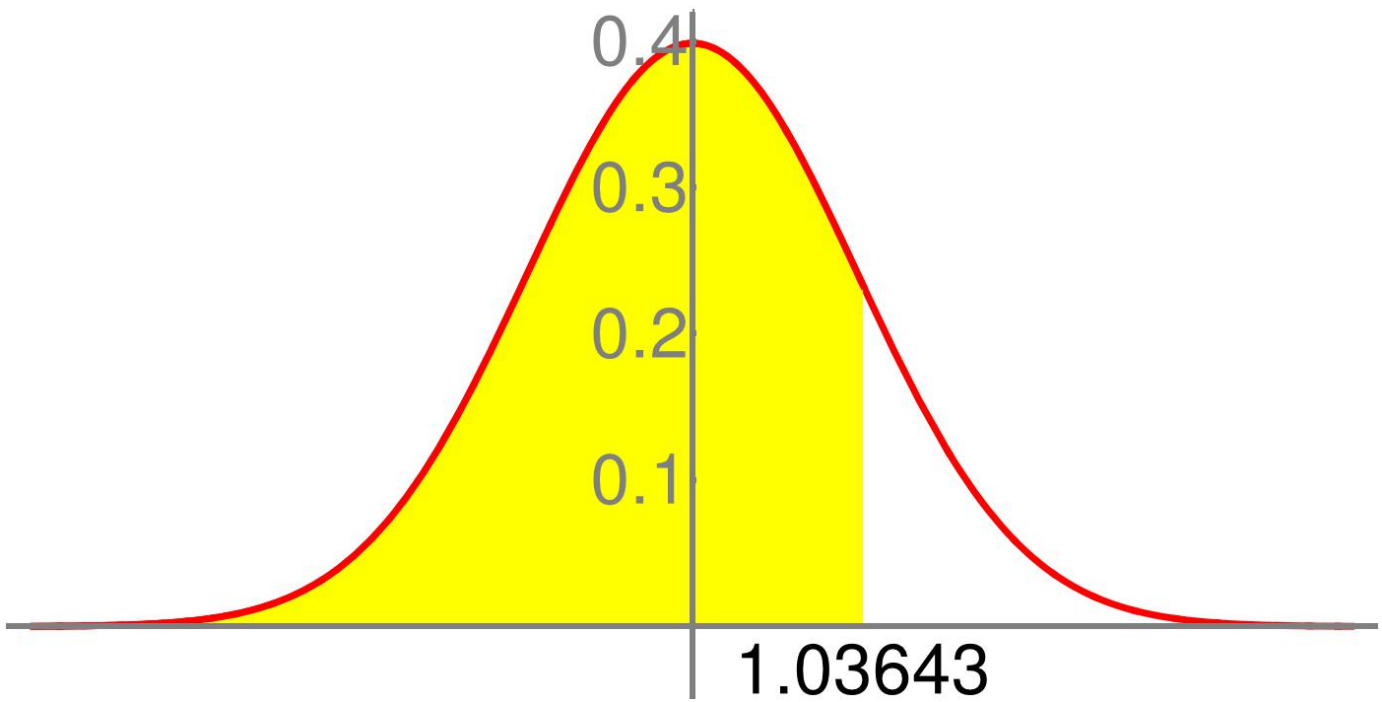
Ver el archivo Contraejemplo.r .

Cuantiles

El p-ésimo cuantil de una variable aleatoria X es aquel valor $\phi_p = \phi_p(X)$ para el cual

$$p = \mathrm{P}\{X \leq \phi_p\}.$$

>qnorm(0.85, 0, 1) # para calcular $\phi_{0.75}$ de una Normal$(0, 1)$
[1] 1.036433



Intervalos de confianza

Un intervalo (L, U) del $(1-\alpha)\,\%$ de confianza para un parámetro θ es aquel para el cual

$$\mathrm{P}\{L \leq \theta \leq U\} = 1 - \alpha.$$

Por ejemplo, el estimador de máxima verosimilitud para θ que se obtiene a partir de una muestra $X_1, \cdots, X_n$ de la población $X \sim \text{Exponencial}(\theta)$ tiene una distribución aproximadamente normal

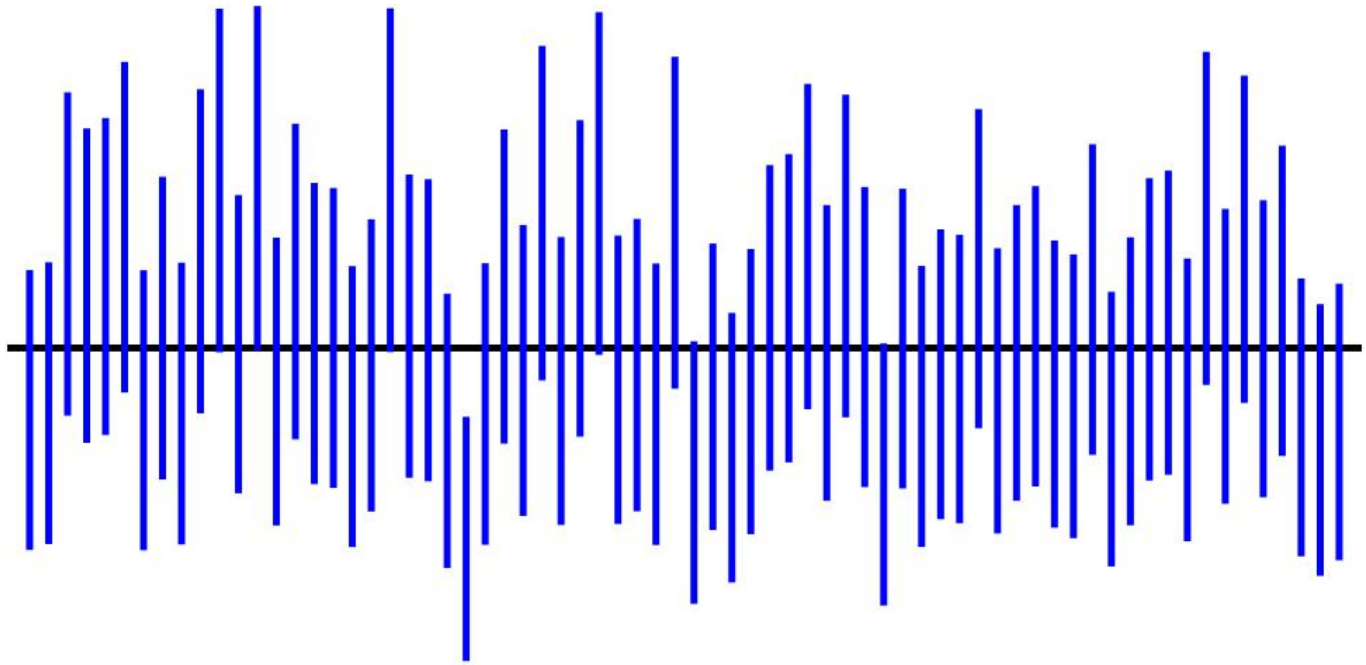
$$\hat{\theta} \sim \text{Normal}\left(\theta, \frac{1}{n\,\mathrm{E}\left[\frac{d}{d\theta}\log f(X|\theta)\right]^2}\right) = \text{Normal}\left(\theta, \frac{\theta^2}{n}\right).$$

Por lo tanto, un intervalo del $0.95\,\%$ de confianza está dado por

$$\left(\hat{\theta} + \frac{\hat{\theta}}{\sqrt{n}}\,\phi_{0.025},\ \hat{\theta} + \frac{\hat{\theta}}{\sqrt{n}}\,\phi_{0.975}\right) \text{ con } \phi_p = \phi_p(Z).$$

Intervalos de confianza

En la figura se muestran 70 intervalos de confianza de nivel 95 %.



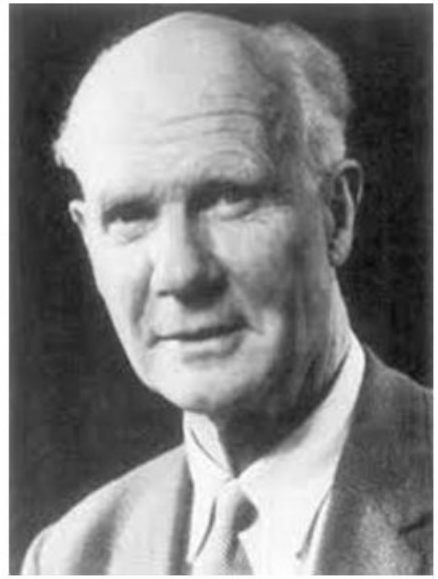
Los intervalos fueron calculados usando la fórmula de la transparencia anterior. Algunos de los intervalos contienen al valor del parámetro verdadero (marcado en linea negra). Un porcentaje de aproximadamente 5 % de los intervalos, no contienen al parámetro verdadero. Ver archivo IntervaloDeConfianza.nb .

Prueba de hipótesis

Three individuals are responsible for developing the statistical analog to Popper's falsificationism: Ronald Fisher, Jerzy Neyman, and Egon Pearson.

Ronald Fisher
(1890 – 1962)

Jersey Neyman
(1894 – 1981)



Egon Pearson
(1895 – 1980)

Tres personajes son responsables de haber desarrollado el análogo estadístico al falsacionismo de Popper.

Prueba de hipótesis

Una prueba de hipótesis es forma de verificar una afirmación sobre la distribución de una variable aleatoria X. Se trata de ver si la información acerca de X que está contenida en una muestra $X_1, \cdots, X_n$, tiende a confirmar la afirmación o más bien la contradice.

Consideremos un ejemplo. Sabemos que $X \sim \text{Normal}(\mu, 2)$, en donde $\mu \in \{0, 2\}$. ¿Cómo podemos utilizar la información en una muestra $X_1, \cdots, X_n$, para decidir si la afirmación

$$\text{H}_0 : \mu = 0$$

es cierta o no?

Note que si H_0 es cierta, entonces $\bar{X}$ debería estar muy cerca de 0. ¿Qué pasa si observamos que $\bar{X} = 2$? Por otro lado, ¿es el valor $\bar{X} = 0.75$ evidencia suficiente como para afirmar que H_0 es falsa?

Prueba de hipótesis

Es natural rechazar la hipótesis H_0 cuando $\bar{X}$ sea grande, digamos, mayor que q. ¿Qué tan grande debe ser q para que la probabilidad

$$\alpha = \mathrm{P}\{\text{error tipo I}\} = \mathrm{P}\{\text{rechazar } H_0 \,|\, H_0 \text{ es verdadera}\}$$

de equivocarnos cuando rechazamos H_0 sea tolerable?

Cuando el tamaño de la muestra es igual a n, entonces $\bar{X} \sim \text{Normal}\left(\mu, \frac{2}{\sqrt{n}}\right)$. Poniendo $\alpha = 0.05$ se obtiene que

$$\begin{aligned} 0.05 &= \mathrm{P}\{\bar{X} \geq q | \mu = 0\} = \mathrm{P}\left\{\frac{\bar{X}}{2/\sqrt{n}} \geq \frac{\sqrt{n}}{2} q \,\middle|\, \mu = 0\right\} \\ &= \mathrm{P}\left\{Z \geq \frac{\sqrt{n}}{2} q\right\}. \end{aligned}$$

Por lo tanto $q = \frac{2}{\sqrt{n}} \phi_{0.95}(Z) = \frac{3.28971}{\sqrt{n}}$.

El siguiente código implementa la prueba de hipótesis que se dedujo en la transparencia anterior

```
mu <- 0 # el valor de la media de la población
n <- 5 # el tamaño de la muestra
q <- 3.289707/sqrt(n) # ver la lámina anterior
tot <- 10 # número de veces que se aplica la prueba
sum <- 0
for(i in 1:tot) {
    x <- rnorm(n, mean=mu, sd=2)
    x.bar <- mean(x)
    if(x.bar < q) {sum <- sum+1}
}
print(sum/tot) # porcentaje de error tipo I
```

Córrase este código con distintos valores para la variable tot. Ver el archivo Prueba.de.Hipótesis.r .

Prueba de hipótesis: tipos de error

Error tipo I: rechazar H_0 cuando H_0 correcta.

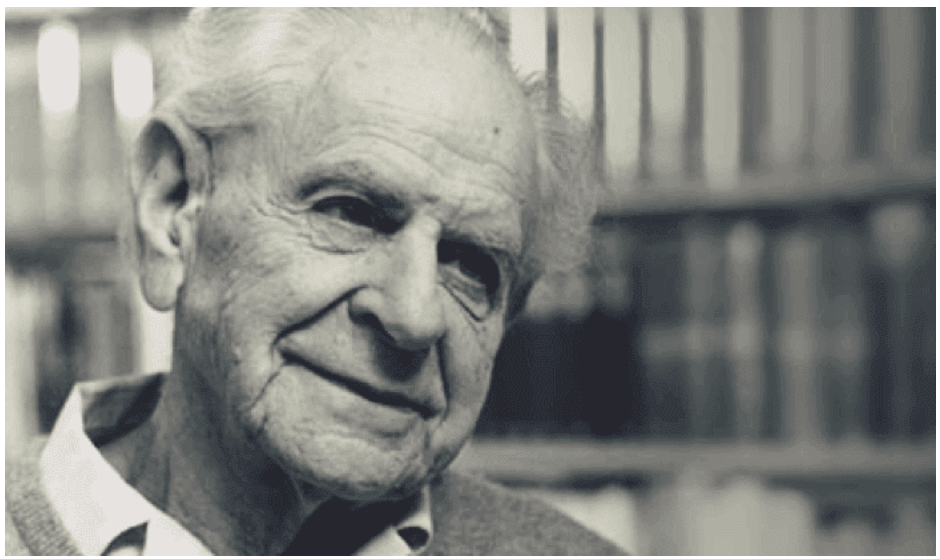
Error tipo II: no rechazar H_0 cuando H_0 incorrecta.

Type I error
(false positive)

Type II error
(false negative)

El falsacionismo de Karl Popper

El falsacionismo, o principio de falsabilidad, es una corriente epistemológica fundada por Karl Popper (1902-1994). Para Popper, contrastar una teoría significa intentar refutarla mediante un contraejemplo. Si no es posible refutarla, dicha teoría queda corroborada, pudiendo ser aceptada provisionalmente, pero no verificada; es decir, ninguna teoría es absolutamente verdadera, sino a lo sumo se mantiene como «no refutada». El falsacionismo es uno de los pilares del método científico.



Prueba de hipótesis para la diferencia de medias

Suponga que tenemos dos poblaciones

$$X \sim \text{Normal}(\mu_x, \sigma_x^2) \qquad \text{y} \qquad Y \sim \text{Normal}(\mu_y, \sigma_y^2).$$

Estamos interesados en probar la hipótesis

$$\text{H}_0 : \mu_x - \mu_y = \Delta_0 \quad \text{conta} \quad \text{H}_1 : \mu_x - \mu_y \neq \Delta_0$$

Sean $X_1, \ldots, X_{n_x}$ y $Y_1, \ldots, Y_{n_y}$ muestras de las poblaciones X y Y. Es estadístico de prueba es

$$T_0 = \frac{\bar{X} - \bar{Y} - \Delta_0}{s_p \sqrt{\frac{1}{n_x} + \frac{1}{n_y}}} \quad \text{en donde} \quad s_p^2 = \frac{(n_x - 1)s_x^2 + (n_y - 1)s_y^2}{n_x - n_y - 2}$$

y $s_x^2 = \dfrac{1}{n_x - 1} \displaystyle\sum_{i=1}^{n_x} (X_i - \bar{X})^2$ el estimador de la varianza σ_x^2.

Prueba de hipótesis para la diferencia de medias

Es claro que valores de $|T_0|$ pequeños son congruentes con la hipótesis H_0. ¿Qué tan grande debe ser $|T_0|$ para poder rechazar H_0? Puesto que la distribución del estadístico de prueba es conocida (distribución T de Student) entonces es posible calcular el valor q tal que H_0 se rechaza siempre que $|T_0| > q$ con una probabilidad de error tipo I prefijada.

El comando

>t.test(x, y, mu=0.0)

realiza la prueba de la hipótesis H_0 en donde mu=Δ_0. Como resultado de ejecutar este comando se obtendrá el así llamado p-valor.

William Sealy Gosset

William Sealy Gosset (1876-1937) fue un estadístico, mejor conocido por su sobrenombre literario Student. Estudió química y matemática en el New College de Oxford. Tras graduarse en 1899, se incorporó a las destilerías Guinness en Dublín. Guinness era un negocio agroquímico y ahí Gosset pudo aplicar sus conocimientos estadísticos tanto a la destilería como a la granja para seleccionar las mejores variedades de cebada.

Gosset introdujo la distribución T de Student para realizar la prueba de diferencia de medias.

$$T(k) = \frac{Z}{\sqrt{\mathrm{Ji}^2(k)/k}}$$

El p-valor de una prueba de hipótesis

El p-valor es la probabilidad de que se observe un valor del estadístico de prueba como el que se ha observado al realizar el experimento, bajo el supuesto de que H_0 es cierta.

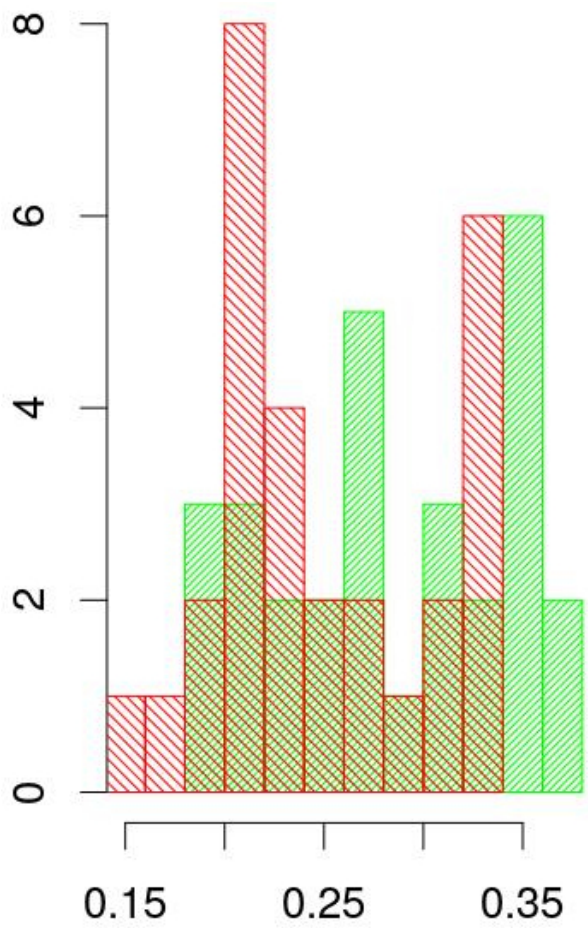
Por lo tanto, p-valores pequeños deben interpretarse como evidencia en contra de H_0.

Como ayuda para asimilar el concepto de p-valor, se puede correr el código en el archivo Prueba.T.de.Student.r con distintos para las medias de X y Y.

```
nx <- 100 # tamaño de la muestra de X
ny <- 50 # tamaño de la muestra de Y
desviación <- 1 # desviación estándar común de las dos poblaciones
x <- rnorm(nx, 0.0, sigma)
y <- rnorm(ny, 0.5, sigma)
t.test(x, y, mu=0) # prueba de la hipótesis H0 : μx = μy
```

Aplicación al consumo de energía eléctrica

En la figura se muestran dos histogramas sobrepuestos. El histograma en color rojo representa el consumo personal diario de energía eléctrica durante los meses de verano, y en verde, en consumo en los meses de invierno. Se aplicó la prueba T de Student para contrastar la hipótesis nula según cual no existe diferencia en los consumos, y se observó un p-valor igual a 0.02701, con lo cual podemos rechazar esta hipótesis. Ver el archivo Consumo.Electricidad.r .



Prueba para la igualdad de dos varianzas

Sean σ_1^2 y σ_2^2 las varianzas de dos poblaciones normales e independientes. Para probar la hipótesis

$$H_0 : \sigma_1^2 = \sigma_2^2$$

se usa el comando var.test cuyos dos primeros argumentos son dos vectores numéricos que contienen los datos de cada muestra.

```
>x <- rnorm(50, mean=0, sd=1)
>y <- rnorm(50, mean=0, sd=2)
>var.test(x, y)
F test to compare two variances
data: x and y
F = 0.28962, num df = 49, denom df = 49, p-value = 2.808e-05
alternative hypothesis: true ratio of variances is not equal to 1
```

Ver la página siguiente para más detalles de este comando.
www.rdocumentation.org/packages/stats/versions/3.5.1/topics/var.test

Prueba de significancia para la correlación

El comando cor.test(x, y) nos permite probar la hipótesis nula de que los vectores x, y no están correlacionados.

```
>recta <- function(x) 1+5*x # definimos la relación entre x, y
>set.seed(415)
>x <- c(1:10)
>y <- recta(x)+rnorm(10, 0, 5)
>cor.test(x, y, method="pearson") # otro método: "spearman"
```

Pearson's product-moment correlation

data: x and y
t = 6.0949, df = 8, p-value = 0.0002911
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
0.6469479 0.9780966
sample estimates:
cor 0.907086

La prueba de bondad de ajuste de Kolmogorov-Smirnov

Dada una muestra $X_1, \cdots, X_n$, tomada de una población X, nos interesa probar la siguiente hipótesis.

$$H_0 : X \text{ tiene una distribución } F(x)$$

El comando ks.test nos permite realizar la prueba de bondad de ajuste de Kolmogorov y Smirnov. He aquí algunos ejemplos.

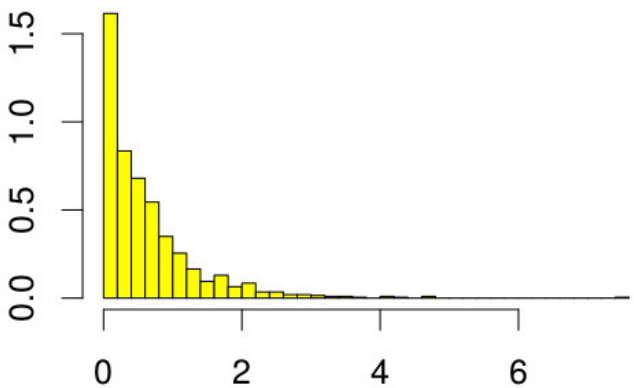
```
>y <- rnorm(20, mean=0, sd=5)
>ks.test(y, "pnorm", mean=0, sd=3)
```

```
>y <- rchisq(50, df=5)
>ks.test(y, "pchisq", df=2)
```

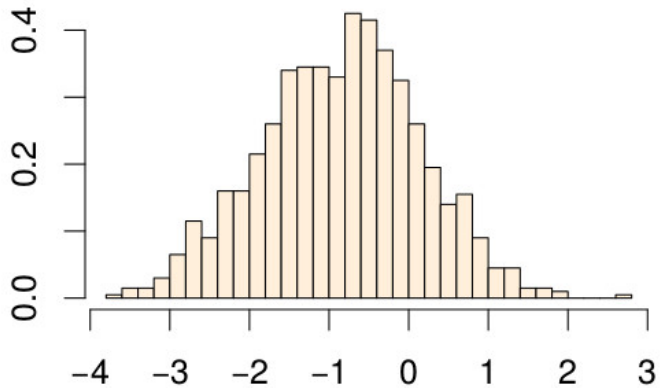
Si el p-valor obtenido es pequeño, entonces no es creíble que la muestra haya sido generada por la distribución $F(x)$.

La transformación Box-Cox

Cuando un conjunto de datos no se distribuye normalmente, entonces es posible transformarlos de manera que los datos transformados sí se distribuyan normalmente. En el lado derecho tenemos una muestra de una población gamma. En el lado derecho tenemos la misma muestra después de que se aplicó la transformación Box-Cox. Ver el archivo Box.Cox.r .



Los datos originales



Los datos transformados

Aquí se puede usar la prueba de Kolmogorov-Smirnov para evaluar el ajuste de los datos transformados a la distribución normal.

La prueba Ji cuadrada de bondad de ajuste

Suponga que una variable X tiene una función de distribución de masa de probabilidad (dmp) dada por la siguiente tabla.

j	1	2	3	4	5
$\mathrm{P}\{X = j\}$	p_1	p_2	p_3	p_4	p_5

Dada una muestra de la cual sospechamos que fue tomada de la distribución dmp, con el comando

>chisq.test(muestra, p=dmp)

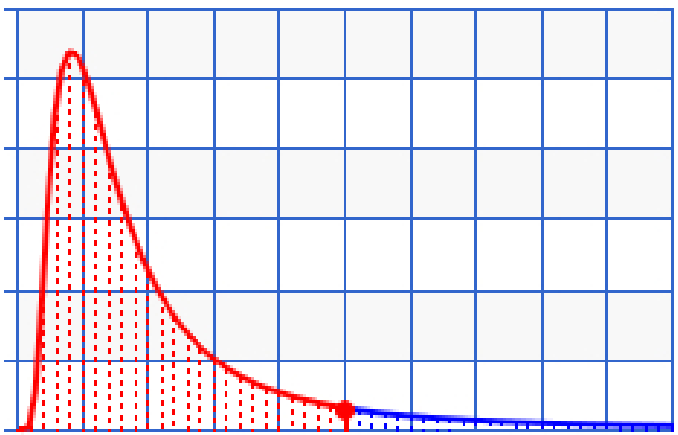
se obtiene el p valor para la prueba de la hipótesis nula H_0 que afirma que la muestra fue tomada de la población X.

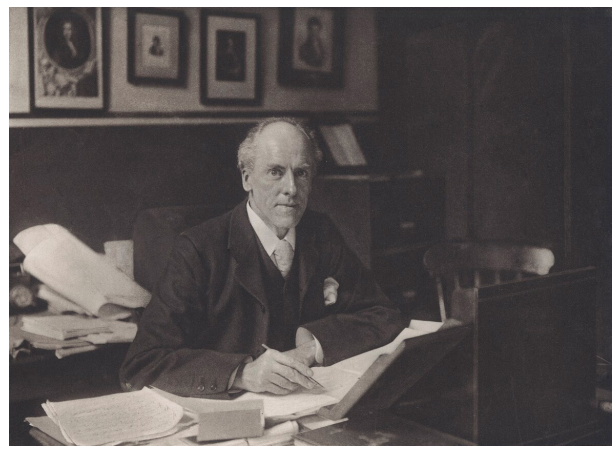
Ver el archivo Ji.Cuadrada.r .

La prueba χ^2 de bondad de ajuste

En el siglo XIX, se aplicaron métodos estadísticos a datos biológicos. Los investigadores suponían que los datos seguían una distribución normal. Pero, en 1900, Karl Pearson criticó este supuesto y observó que los histogramas obtenidos presentaban una asimetría (skewness).

En una serie de artículos entre 1893 y 1916, Pearson introdujo una familia de distribuciones, de la cual, la distribución normal era un caso particular. Pearson introdujo la prueba Ji cuadrada para mostrar que las muestras obtenidas no no podían provenir de una distribución normal.





Regresión lineal simple

Considere una variable Y de la forma

$$Y = \beta_0 + \beta_1 x + \epsilon$$

en donde β_0 y β_1 son dos parámetros del modelo, x es una variable no aleatoria y

$$\epsilon \sim \text{Normal}(0, \sigma^2).$$

Aquí, σ^2 es el tercer parámetro del modelo. Suponga que tenemos n pares de datos

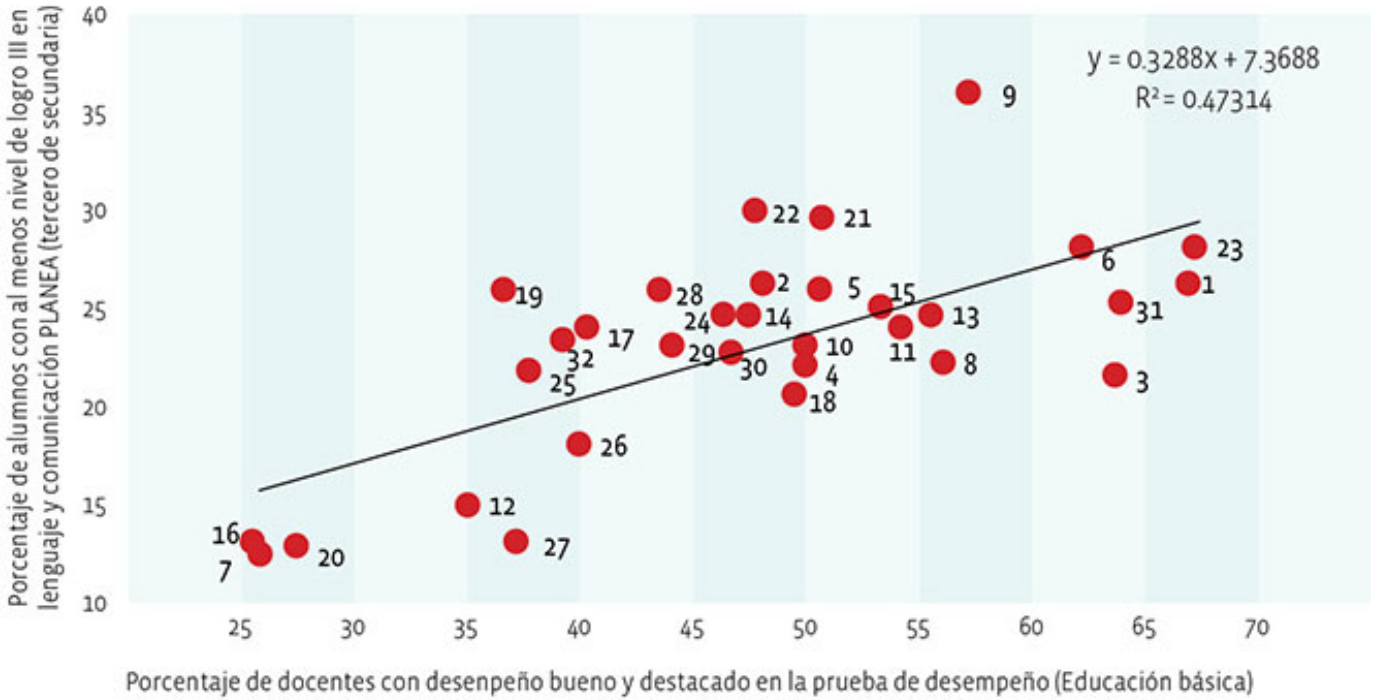
$$(x_1, Y_1), (x_2, Y_2), \cdots, (x_n, Y_n).$$

A partir de esta muestra, se obtienen estimaciones para los tres parámetros del modelo, $\hat{\beta}_0$, $\hat{\beta}_1$ y $\hat{\sigma}^2$.

Ejemplo de aplicación del modelo

GRÁFICA 1

Desempeño docente y calificaciones obtenidas por los alumnos



1. Aguascalientes
2. Baja California
3. Baja California Sur
4. Campeche
5. Coahuila
6. Colima
7. Chiapas
8. Chihuahua
9. Distrito Federal
10. Durango
11. Guanajuato
12. Guerrero
13. Hidalgo
14. Jalisco
15. México
16. Michoacán
17. Morelos
18. Nayarit
19. Nuevo León
20. Oaxaca
21. Puebla
22. Querétaro
23. Quintana Roo
24. San Luis Potosí
25. Sinaloa
26. Sonora
27. Tabasco
28. Tamaulipas
29. Tlaxcala
30. Veracruz
31. Yucatán
32. Zacatecas

Propiedades de los estimadores del modelo

Notación: $e_i = y_i - \hat{y}_i$ es el i-ésimo residual, siendo $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$

$$\text{SCE} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \quad \text{es la suma de los cuadrados del error}$$

1. $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$ y $\hat{\beta}_1 = \dfrac{S_{xy}}{S_{xx}}$ en donde $S_{xy} = \displaystyle\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$
2. $\text{E}(\hat{\beta}_0) = \beta_0$ y $\text{E}(\hat{\beta}_1) = \beta_1$
3. $\text{var}(\hat{\beta}_1) = \dfrac{\sigma^2}{S_{xx}}$ y $\text{var}(\hat{\beta}_0) = \left(\dfrac{1}{n} + \dfrac{\bar{x}^2}{S_{xx}}\right)\sigma^2$
4. $\hat{\beta}_1$, $\bar{Y}$ y SCE son independientes
5. $\displaystyle\sum_{i=1}^{n} \left(\frac{e_i}{\sigma}\right)^2 \sim \text{Ji}^2(n-2)$ y $\hat{\sigma}^2 = \dfrac{\text{SCE}}{n-2} =: \text{CME}$ es insesgado

Las distribuciones de la teoría estadística

$$Z = \frac{1}{\sqrt{\text{var}(x)}} \sum_{i=1}^{n} (X_i - \mu) \quad \text{(teorema del límite central)}$$

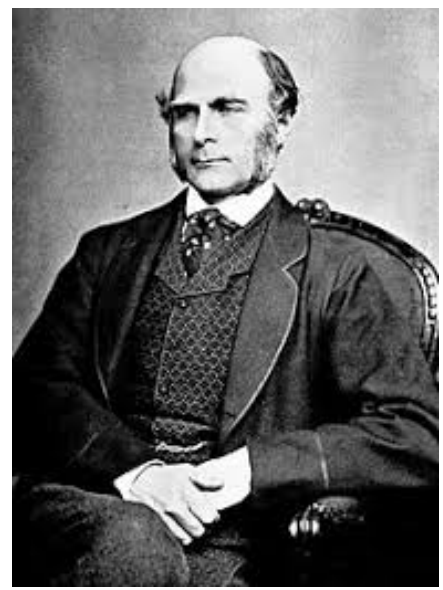
$$\text{Ji}^2(k) = \sum_{i=1}^{k} Z_i^2 \quad \text{(suma de cuadrados de normales estándar)}$$

$$T(k) = \frac{Z}{\sqrt{\text{Ji}^2(k)/k}} \quad \text{(estandarización usando } \hat{\sigma} \text{ en lugar de } \sigma\text{)}$$

$$F(k, r) = \frac{\text{Ji}^2(k)/k}{\text{Ji}^2(r)/r} \quad \text{(cociente de Ji cuadradas)}$$

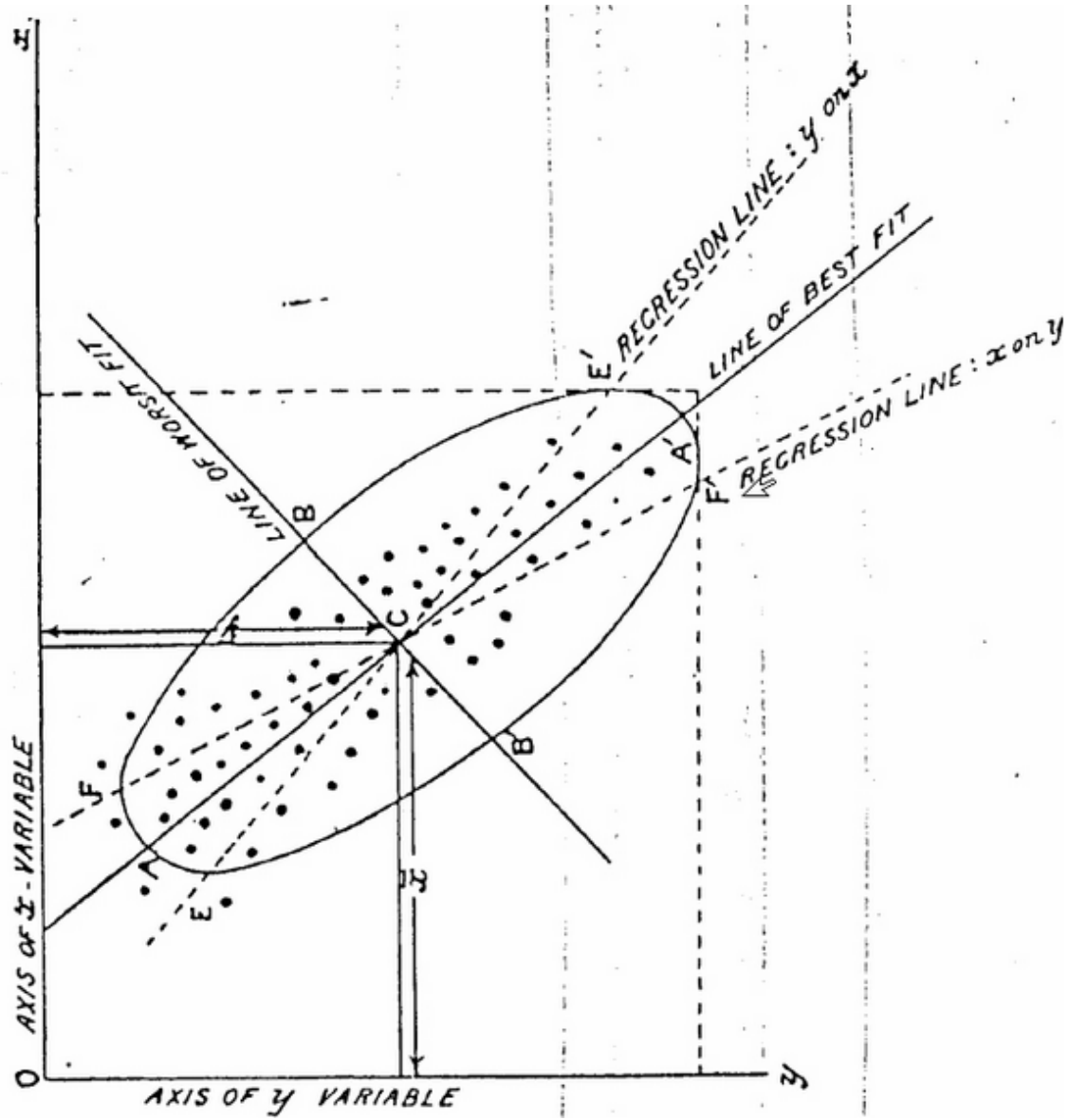
Francis Galton

Francis Galton (1822-1911) fue un polímata, antropólogo, geógrafo, explorador, inventor, meteorólogo, estadístico, psicólogo y eugenista británico. Creó el concepto estadístico de correlación y regresión hacia la media, altamente promovido. Él fue el primero en aplicar métodos estadísticos para el estudio de las diferencias humanas y la herencia de la inteligencia, introdujo el uso de cuestionarios y encuestas para recoger datos sobre las comunidades humanas.



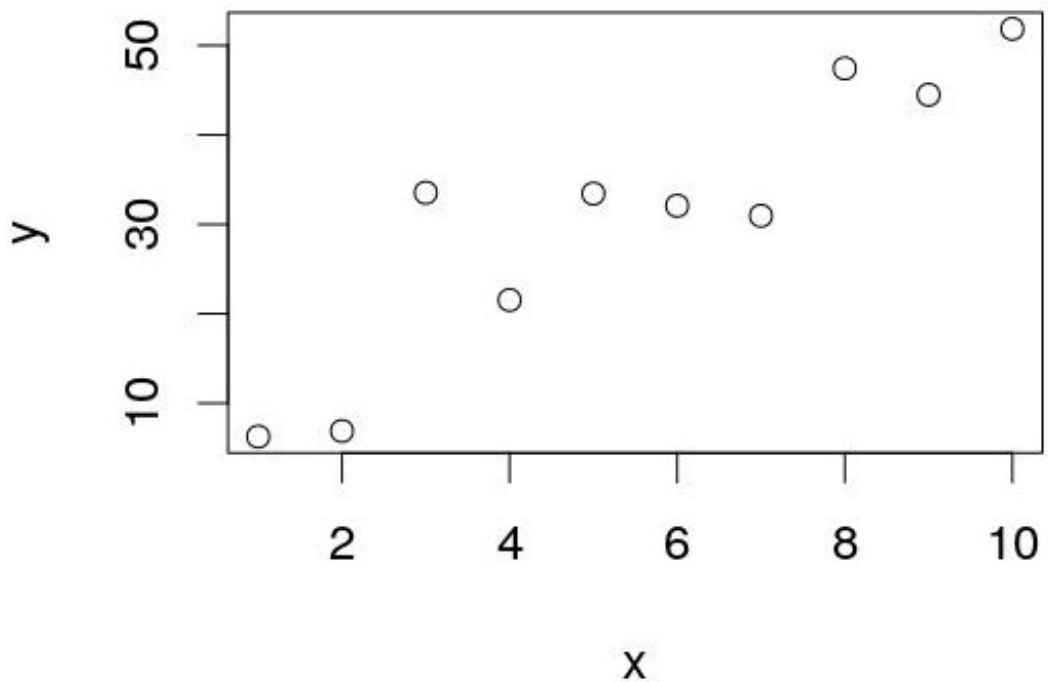
Regresión hacia la media

En estadística, la regresión hacia la media es el fenómeno en el que si una variable es extrema en su primera medición, tenderá a estar más cerca de la media en su segunda medición y, paradójicamente, si es extrema en su segunda medición, tenderá a haber estado más cerca de la media en su primera.



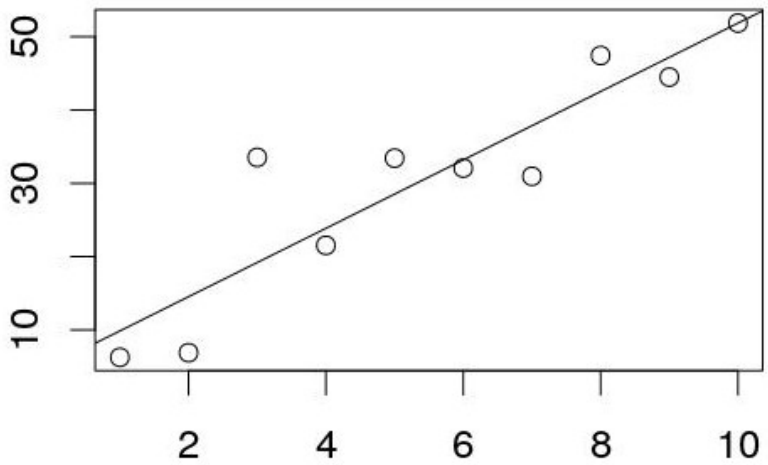
Regresión lineal simple: diagrama de dispersión

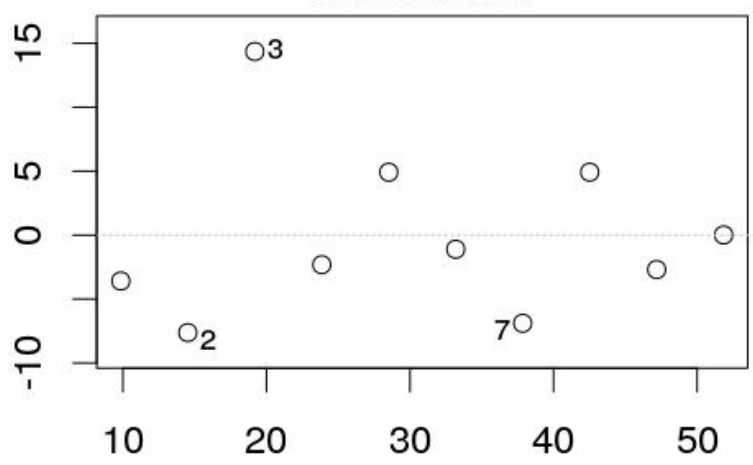
```
>recta <- function(x) 1+5*x # define una función lineal "recta(x)"
>set.seed(415)
>x <- c(1:10)
>y <- recta(x)+rnorm(10, 0, 5)
>dev.new(width=6, height=5)
>par(cex=1.5)
>plot(x, y) # para obtener el diagrama de dispersión
```



Regresión lineal simple: recta estimada

>cor(x, y) # calcula la correlación entre x, y
>A <- lm(y ∼ x) # asigna a A el modelo estimado y=b+m∗x
>print(A) # despliega los coeficientes calculados para el modelo
>abline(A) # añade la recta de regresión al diagrama de dispersión
>plot(A, which=1, add.smooth=FALSE) # grafica los residuales





Regresión lineal simple: los estadísticos del modelo

```
>summary(A) # para obtener el resumen del modelo

Residuals:
    Min      1Q  Median      3Q     Max
-7.623  -3.364  -1.708   3.693  14.355

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.1919     4.7498   1.093 0.306184
x             4.6657     0.7655   6.0950 0.000291    ***
- - -
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.953 on 8 degrees of freedom
Multiple R-squared: 0.8228, Adjusted R-squared: 0.8007
F-statistic: 37.15 on 1 and 8 DF,  p-value: 0.0002911
```

Coeficiente de determinación R^2

El coeficiente de determinación de la regresión está dado por

$$R^2 = \frac{\text{suma de cuadrados de la regresión}}{\text{suma de cuadrados del error}} = \frac{S_{xy}^2}{S_{xx}S_{yy}}.$$

El estadístico R^2 nos dice cuál es el porcentaje de variación de los datos que se explica por el modelo.

R^2 no debe usarse para evaluar el ajuste del modelo a los datos.

Use the Correlation Coefficient to Summarize Regression Performance?

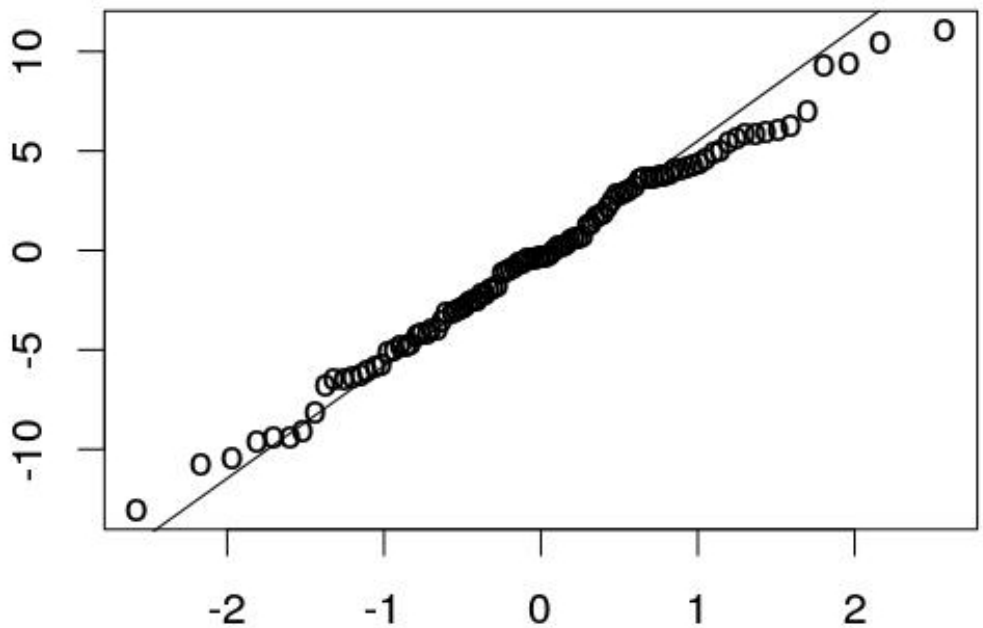
KEYWORDS:
Teaching;
Goodness of fit.

Prakash Gorroochurn
Columbia University, New York, USA.
e-mail: pg2113@columbia.edu

Summary
The correlation coefficient is commonly used to indicate the quality of fit in regression. This practice is questionable.

Gráfica de los residuales en papel normal

```
>set.seed(405)
>x <- seq(from=0.1, to=10, by=0.1)
>Y <- recta(x)+rnorm(length(x), 0, 5)
>A <- lm(Y ~ x)
>modelo <- function(x) -0.3041+5.2817*x
>Y1 <- modelo(x)
>qqnorm(Y-Y1, pch="o")
>qqline(Y-Y1)
```



Lectura de datos desde un archivo externo

>datos <- read.table("dat.txt") # lee el archivo "dat.txt"
>dput(datos, file="borrar.txt") # guarda el objeto "datos"
>datos1 <- load("borrar.txt") # carga el archivo "borrar.txt"

R tiene en su memoria permanente distintos conjuntos de datos.

>data() # muestra la colección de datos
>data(AirPassengers) # carga los datos del archivo "AirPassengers"
>AirPassengers # despliega el archivo "AirPassengers"

>save(AirPassengers, file="AirPassengers.RData") # archiva "AirPassengers" en el directorio en uso

>rm("AirPassengers") # elimina el "AirPassengers" del ambiente R

>load("AirPassengers.RData") # lectura de "AirPassengers" del directorio en uso

Estructura de datos: dataframe

R usa cuatro tipos básicos de estructuras de datos:

1. Un vector puede tener entradas numéricas, de texto o de valores lógicos.
2. Un arreglo (array) es un vector con especificaciones de dimensión. El arreglo más común es la matriz.
3. Un factor es un objeto usado para definir variables categóricas.
4. Un marco de datos (data frame) es una lista de vectores de la misma dimensión. Cada vector en un marco de datos corresponde a una variable de un experimento.

Estructura de datos: dataframe

Con la siguiente sucesión de comandos vamos a definir un marco de datos.

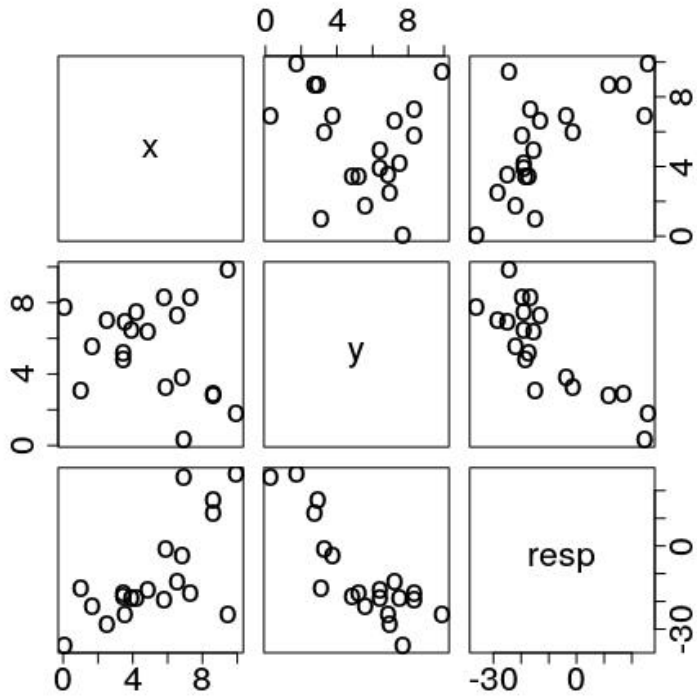
```
>vec2 <- c(1, 2, 3.4, 5.6, 7)
>vec3 <- c("tipo A", "tipo B", "tipo C", "tipo D", "tipo E")
>vec4 <- c(TRUE, TRUE, FALSE, TRUE, FALSE)
>f1 <- factor(vec3) # usa el vector vec3 para hacer el factor f1
>datos <- data.frame(vec2, f1, vec4) # da como resultado:
```

	vec2	f1	vec4
1	1.0	tipo A	TRUE
2	2.0	tipo B	TRUE
3	3.4	tipo C	FALSE
4	5.6	tipo D	TRUE
5	7.0	tipo E	FALSE

Con el comando fix(datos) se pueden hacer modificaciones a este marco de datos.

Regresión lineal múltiple

```
>set.seed(777)
>f <- function(x, y) 3*x-5*y+1
>x <- runif(20, 0, 10)
>y <- runif(20, 0, 10)
>resp <- f(x, y) + rnorm(20, 0, 3)
>datos <- data.frame(x, y, resp)
>plot(datos, cex=1.5, cex.axis=2)
# genera la gráfica:
>lm(datos$resp~datos$x+datos$y)
# da como resultado:
```



Coefficients:

(Intercept)	datos$x	datos$y
-0.658	3.256	-4.958

Ver el archivo Regresión.Múltiple.r .

Varios comandos para la regresión lineal

Suponga que se tienen datos de la forma

$$y_i = \beta_0 + \beta_1 u_i + \beta_2 v_i + \beta_3 w_i + \epsilon_i.$$

Con el comando >fit <- lm(y $\sim$ u + v + w) se obtiene el ajuste del modelo. La tabla siguiente contiene comandos para obtener distintas estadísticos de la regresión.

anova(fit)	Tabla ANOVA
coefficients(fit)	Coeficientes del modelo
confint(fit)	Intervalos de confianza para los coeficientes
fitted(fit)	Valores estimados y_i
residuals(fit)	Los residuales
summary(fit)	Los principales estadísticos de la regresión

Ver el archivo Regresión.Comandos.Varios.r .

Más comandos para la regresión

Con el comando >lm(y $\sim$ x + 0) se ajusta el modelo sin término constante $y_i = \beta x_i + \epsilon_i$.

Con el comando >lm(y $\sim$ u$*$v) se ajusta el modelo

$$y_i = \beta_0 + \beta_1 u_i + \beta_2 v_i + \beta_3 u_i v_i + \epsilon_i.$$

Ver el archivo Regresión.Términos.Cruzados.r .

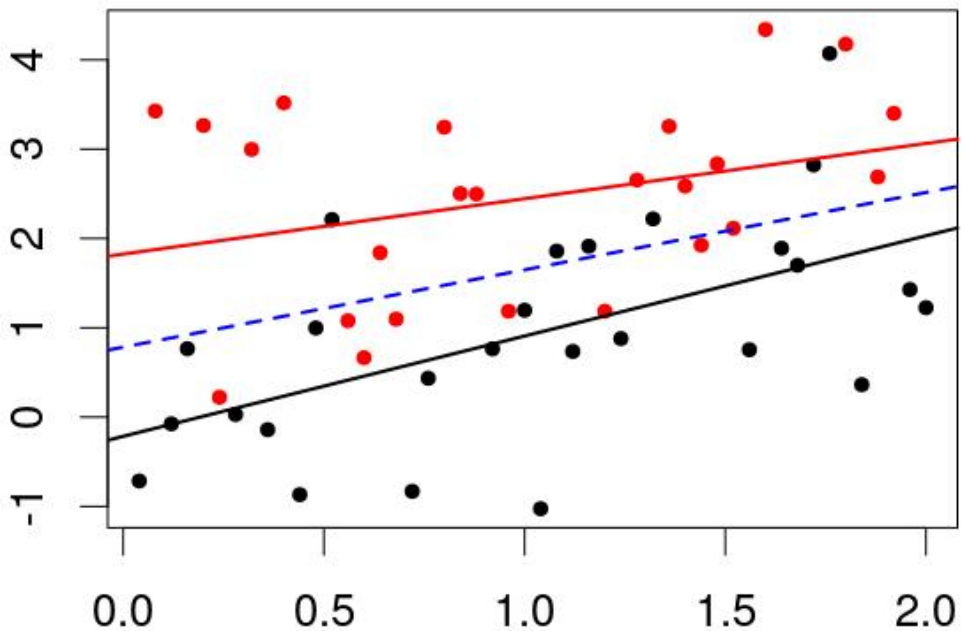
Con los dos comandos que siguen, es posible seleccionar el mejor conjunto de variables explicativas.

>full.model <- lm(y $\sim$ u + v + w + z)
>reduced.model <- step(full.model, direction="backward")

Ver el archivo Regresión.Modelo.Reducido.r .

Análisis de regresión: variables indicadoras

Considere el modelo $y = \beta_0 + \beta_1 u + \beta_2 v + \epsilon$, en donde la variable v solamente puede tomar los valores 0 o 1. En la gráfica los valores de y que se obtienen cuando $v = 0$, aparecen en negro y los mismos valores cuando $v = 1$, aparecen en rojo. En interesante la prueba de hipótesis según la cual los dos interceptos (cuando $v = 0$ y $v = 1$) son iguales.



Ver el archivo Variables.Indicadoras.r .

Análisis de regresión: variables indicadoras

Con el modelo de la lámina anterior, $Y = \beta_0 + \beta_1 u + \beta_2 v + \epsilon$, consideramos ahora el ajuste del modelo ampliado

$$Y = \beta_0 + \beta_1 u + \beta_2 v + \beta_{12} uv + \epsilon.$$

Cuando $v = 0$, se obtiene que

$$\mathrm{E}(Y) = \beta_0 + \beta_1 u.$$

Cuando $v = 1$, se obtiene que

$$\begin{aligned} \mathrm{E}(Y) &= \beta_0 + \beta_1 u + \beta_2 + \beta_{12} u \\ &= (\beta_0 + \beta_2) + (\beta_1 + \beta_{12}) u. \end{aligned}$$

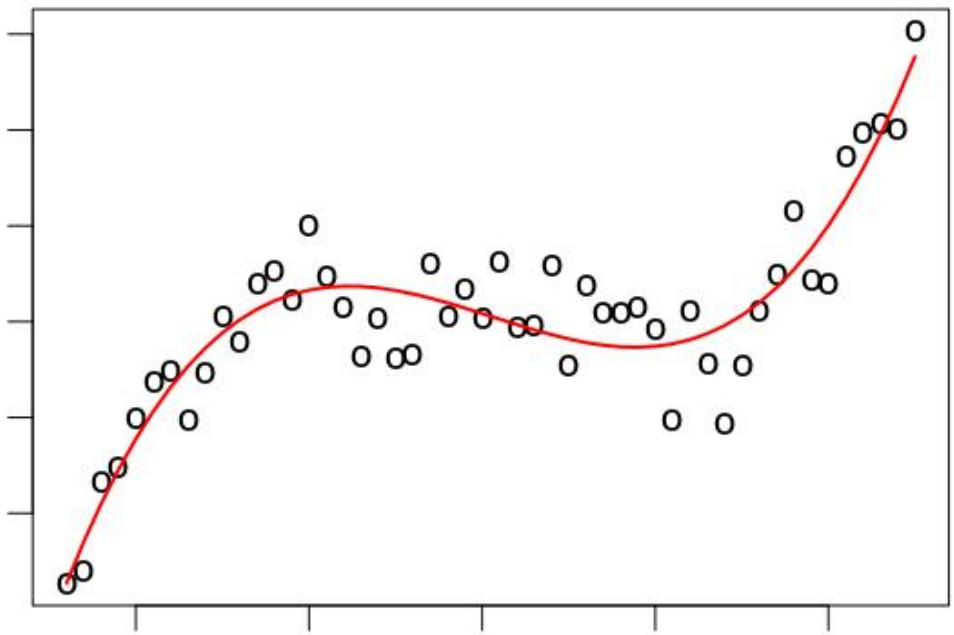
Son interesante las hipótesis nulas $\mathrm{H}_0 : \beta_2 = 0$ y $\mathrm{H}_0 : \beta_{12} = 0$.

Regresión polinomial

Cuando queremos ajustar a un conjunto de datos, un modelo de la forma

$$y = \beta_0 + \beta_1 u + \beta_2 u^2 + \cdots + \beta_p u^p + \epsilon,$$

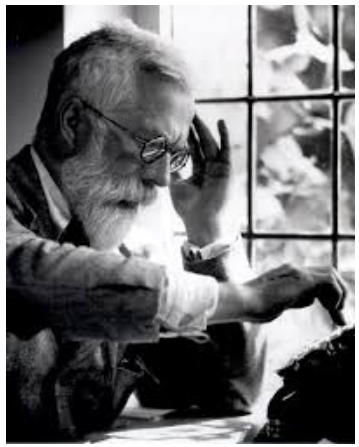
se usa el comando >lm(y $\sim$ poly(u, p, raw=TRUE)), en donde p es el grado del polinomio que queremos ajustar.



Ver el archivo Regresión.Polinomial.r .

Análisis de varianza

En 1919, la Estación Rothamsted contrató al joven estadístico Ronald Aylmer Fisher para que aprovechara los datos ahí acumulados. El análisis de Fisher sugería que la relación entre la lluvia y el crecimiento de las plantas eras más significativa que la relación entre el fertilizante y el crecimiento de las plantas. A los científicos de la estación Rothamsted, no les interesaba la lluvia como factor determinante de la cosechas, sino el el fertilizante. No se sabía separar los efectos de la lluvia de los efectos del fertilizante. Fisher comprendió que los efectos se podían separar si los experimentos se diseñaban de manera apropiada.



Análisis de varianza

El análisis de varianza (ANOVA) de un sólo factor se utiliza para comparar las medias $\mathtt{I}$ poblaciones. Es de interés la hipótesis

$$\mathrm{H}_0 : \mu_1 = \mu_2 = \cdots = \mu_{\mathtt{I}}.$$

En el ANOVA se tienen datos de la forma

$$X_{ij} = \mu_i + \epsilon_{ij}$$

en donde ϵ_{ij} son las desviaciones (o errores) aleatorios que la j-ésima observación respecto de la i-ésima media poblacional μ_i. Se supone que los errores ϵ_{ij} son independientes con media cero y desviación estándar constante.

Si H_0 se cumple, entonces el estadístico de prueba $F_{\text{cal}} = \text{CMTr}/\text{CME}$ tiene un valor alrededor de 1. Valores grandes de F_{cal} sugieren que H_0 es falsa. ¿Qué tan grande debe ser F_{cal} para poder rechazar a H_0?

Análisis de varianza

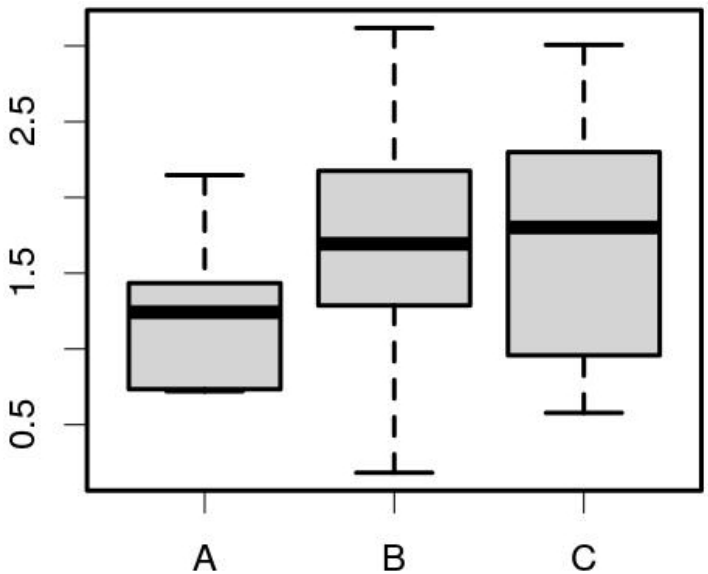
```
>set.seed(777)
>a <- 1.0 + rnorm(10, 0, 0.7)
>b <- 1.5 + rnorm(10, 0, 0.7)
>c <- 2.0 + rnorm(10, 0, 0.7)
>obs <- c(a, b, c)
>A <- rep("A", 10)
>B <- rep("B", 10)
>C <- rep("C", 10)
>fac <- factor(c(A, B, C))
>datos <- data.frame(obs, fac)
>mod <- oneway.test(obs ~ fac, data=datos, var.equal=TRUE)
>print(mod) # se obtiene como resultado lo siguiente:
```

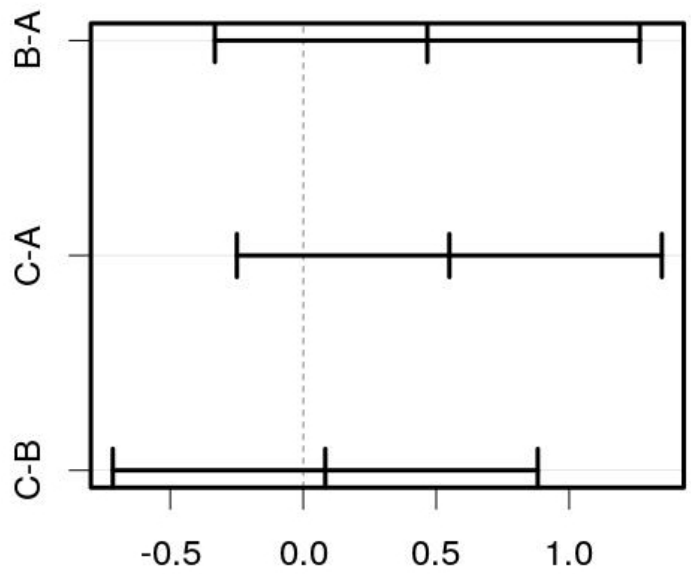
One-way analysis of means

data: obs and fac
F = 1.6901, num df = 2, denom df = 27, p-value = 0.2034

Análisis de varianza

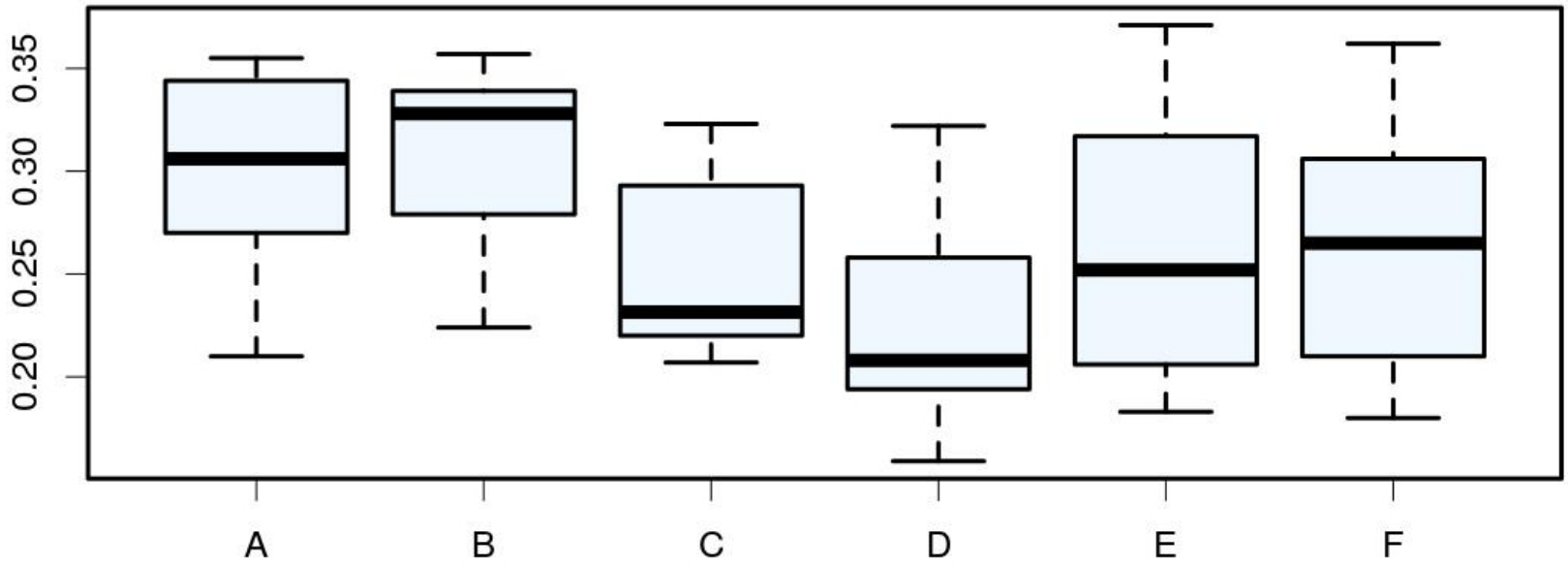
En la figura de la izquierda se reportan los diagramas de caja de las tres poblaciones generadas por el código de la página anterior. Dentro de cada caja yacen las observaciones entre los cuartiles Q_1 y Q_3 de cada población. Los "bigotes" se extienden hasta los valores máximo y mínimo de la serie o hasta $1.5 \times (Q_3 - Q_1)$.





En la figura de la derecha se reportan los intervalos "estudentizados" mediante el procedimiento de Tukey.

ANOVA: consumo bimestral de energía eléctrica

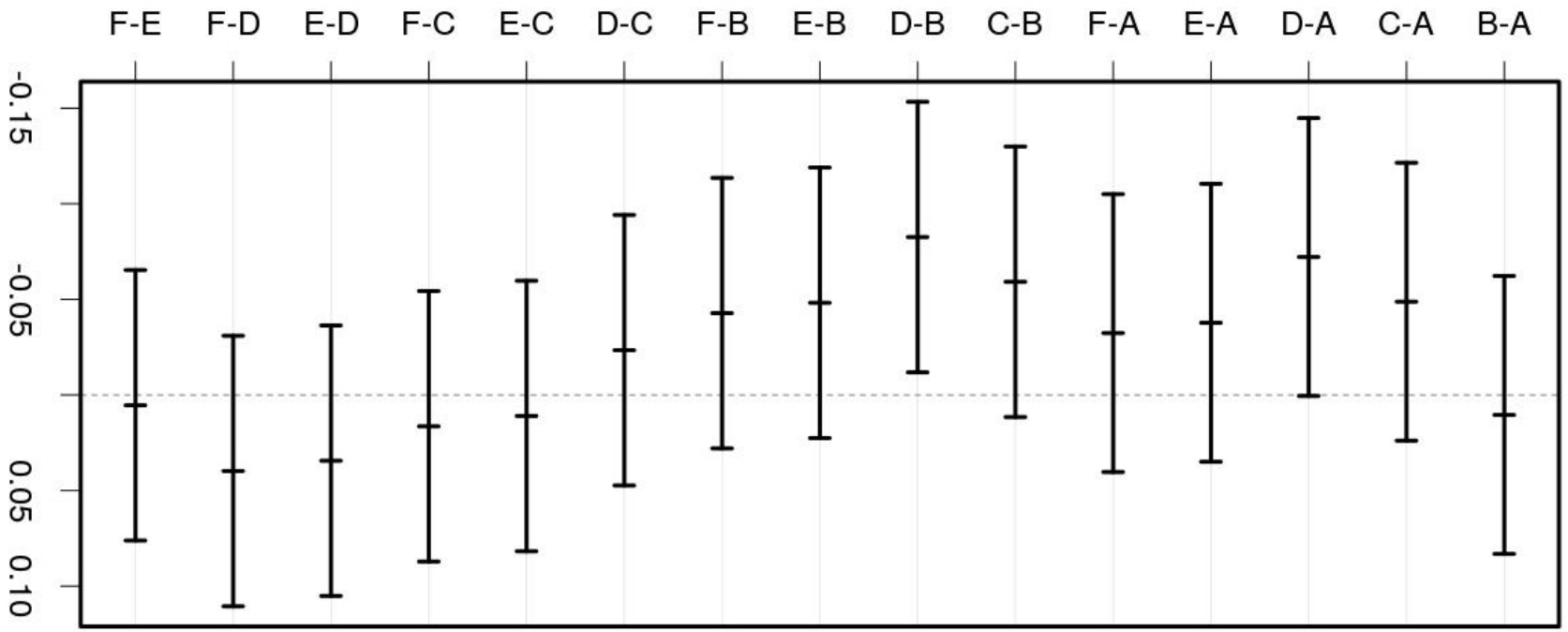


En estos diagramas de caja se muestra el consumo de personal de energía eléctrica por bimestre.

A	diciembre-enero	B	febrero-marzo	C	abril-mayo
D	junio-julio	E	agosto-setbre.	F	octubre-novbre.

ANOVA: consumo bimestral de energía eléctrica

Para la hipótesis nula según la cual no hay diferencias en el consumo, se tiene un p-valor de 0.0126. Por lo tanto se rechaza esta hipótesis. En la figura, los intervalos estudentizados de Tukey.



Se observa la mayor diferencia entre los bimestres junio-julio contra febrero-marzo. Nota: Estas conclusiones se obtuvieron con un tamaño de muestra pequeño.

El ANOVA de un factor como un caso de la regresión

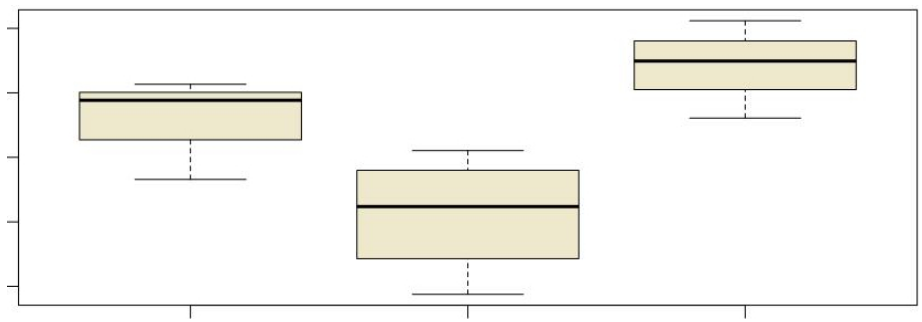
En el archivo Solo.Variables.Indicadoras se considera un modelo de regresión de la forma

$$Y = \beta_0 + \beta_1 v + \beta_2 w + \epsilon$$

en donde v y w son variables indicadoras que solamente toman los valores 0 o 1, pero no se da el caso de que $v = w = 1$. Note que

$$\begin{aligned} \mathrm{E}(Y) &= \beta_0 && \text{si } v = 0 \text{ y } w = 0, \\ &= \beta_0 + \beta_1 && \text{si } v = 1 \text{ y } w = 0, \\ &= \beta_0 + \beta_2 && \text{si } v = 0 \text{ y } w = 1. \end{aligned}$$

Cuando se varían los valores β_0, β_1 y β_2 se cambian los niveles del factor.



ANOVA con dos factores

Un acumulador de energía eléctrica se puede construir de tres tipos distintos de materiales (factor 1 con tres niveles) y trabajará a distintas temperaturas (factor 2). Es de interés el número de horas de servicio del acumulador. El acumulador será probado a tres temperaturas distintas (los tres niveles del factor 2). En la tabla se reportan las horas de servicio obtenidas al aplicar cada tratamiento (combinación de niveles de los factores) a 4 acumuladores elegidos de forma aleatoria.

Temperatura	15°	70°	125°
Tipo 1 de material	130, 155 74, 180	34, 40 80, 75	20, 70 82, 58
Tipo 2 de material	150, 188 159, 126	136, 122 106, 115	25, 70 58, 45
Tipo 3 de material	138, 110 168, 160	174, 120 150, 139	96, 104 82, 60

ANOVA con dos factores

Con los siguientes comandos se realiza el análisis de varianza con dos factores para el problema planteado en la página anterior.

```
> vida <- c(130, 74, 150, 159, 138, 168, 155, 180, 188, 126, 110,
+160, 34, 80, 135, 106, 174, 150, 40, 75, 122, 115, 120, 139, 20,
+82, 25, 58, 96, 82, 70, 58, 70, 45, 104, 60)
>material <- rep(c(1,1,2,2,3,3), 6)
>temp <- rep(c(15, 70, 125), each=12)
>dat <- data.frame(resp=vida, tipo=factor(material),
+temperatura=factor(temp))
>fit <- aov(resp ~ tipo*temperatura, data=dat)
>summary(fit)
```

Ver el archivo ANOVA.Acumulador.r .

Regresión logística

Con los siguientes comandos se simula una muestra en donde la variable respuesta solamente toma los valores 0 o 1.

```
>tt <- 50 # tamaño de la muestra
>x <- 20+c(1:tt)*50/tt
>b0 <- -10
>b1 <- 0.2

>y <- c()
>for (i in 1:tt) {
+EXP <- exp(b0+b1*x[i])
+p <- EXP/(1 + EXP)
+y[i] <- rbinom(1, size=1, prob=p)
+}

>plot(x, y)
```

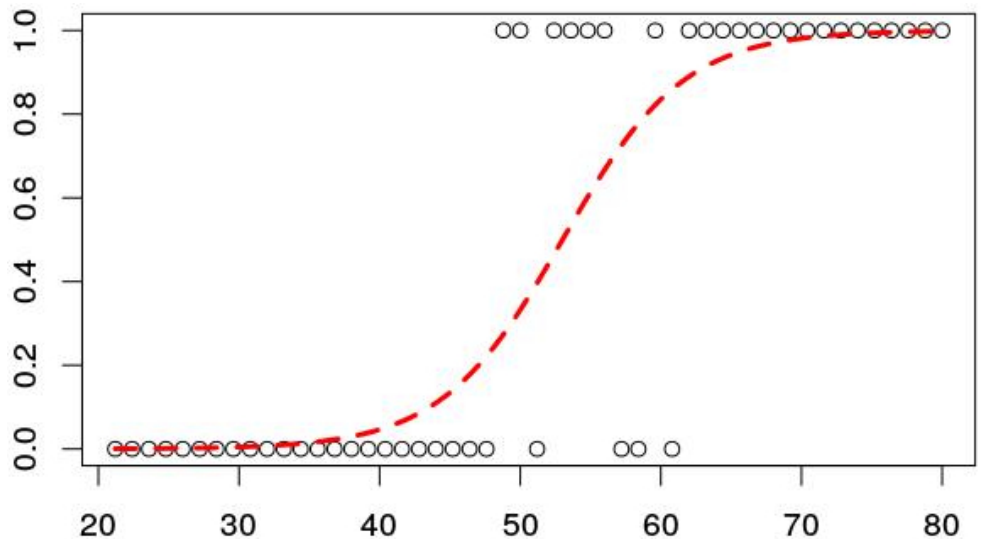
Ver el archivo Regresión.Logística.r .

Regresión logística

Una vez dada la muestra que se generó en la página anterior, los coeficientes del modelo se estiman con el siguiente comando.

>fit <- glm(y ~ x, family="binomial")
>summary(fit)

En la figura se reporta la muestra generada junto con el modelo ajustado (en linea roja punteada). El modelo provee la probabilidad de que un punto de la muestra haya tomado el valor 1. Cuando esta probabilidad es muy pequeña, entonces la variable respuesta tenderá a asumir el valor 0.



Índice de comandos

aov()	ANOVA.r
boxplot()	ANOVA.r
chisq.test()	Ji.Cuadrada.r
cor.test()	Prueba.de.Correlación.r
data.frame()	ANOVA.r
data.frame()	Data.Frame.r
factor()	ANOVA.r
for()	Gama.Momentos.r
for()	Ji.Cuadrada.r
function()	Distribución.Logarítmica.r
function()	EMV.Exponencial.r
function()	Ji.Cuadrada.r
function()	Método.del.Rechazo.r
function()	Prueba.de.Correlación.r
glm()	Regresión.Logística.r
if()	Método.del.Rechazo.r

Índice de comandos

if()	Teorema.Central.del.Límite
lm()	ANOVA.r
lm()	Regresión.Simple.r
lm()	Regresión.Múltiple.r
matrix()	Distribución.Logarítmica.r
oneway.test()	ANOVA.r
optim()	EMV.Exponencial.r
rep()	ANOVA.r
sapply()	Ji.Cuadrada.r
uniroot()	Distribución.Logarítmica.r
which()	Ji.Cuadrada.r
which()	Regresión.Variables.Indicadoras.r
while()	Método.del.Rechazo.r
while()	Distribución.Logarítmica.r

Índice de colores