ESTADISTICA Y DECISIONES
Vol 2 Multivariado

José Rodríguez

29 de julio de 2010

Índice general

PREFACIO

La estadística se confunde con el método científico: contrastar lo que uno ve con lo que uno cree. Habiendo aprendido en el Vol 1 cómo se implementa este programa para algunos temas de la normal univariada, extendemos la misma consideración a otros casos, algunos sencillos, otros complicados con muchas variables y otros válidos para distribuciones no normales.

La estadística actual exige estar preparados para analizar grandes volúmenes de datos, lo cual amerita el uso de software apropiado. Nuestra propuesta es apreder a utilizar dos paquetes que son no sólo profesionales sino también gratuitos. Uno es *Gnumeric*, para trabajo liviano y visto en el vol 1, y el otro *R*, para trabajo pesado y que introduciremos en el capítulo 1, en el cual corremos sobre *R* el tipo de problemas ya vistos en el primer volumen.

Consideramos que dosis estratégicas de teoría hacen la vida más interesante. Por ésa razón, los temas muy fundamentales son ventilados adecuadamente, lo cual nos permitirá avanzar inductivamente a grandes pasos sobre los temas operativamente más complicados pero que en definitiva son más de lo mismo.

José Rodríguez

Capítulo 1

Primero piense, luego exista

```
Sin R no hay experimentos
```

Nos complace proponer un imperialismo que nos hará bien a todos y que de una vez y para siempre pondrá en claro por qué hay que saber estadística muy bien:

Antes de llevar a cabo un experimento, diseñe la planilla de datos, llénelos ficticiamente y analícemos estadísticamente: *R* le será de mucha ayuda. Sería bueno hacer este juego varias veces. Así podrá estar seguro de que el experimento le arrojará luz sobre lo que busca.

Haga lo posible para que esta recomendación sea parte natural de la forma de ser de todo experimentador: que nunca se apruebe un proyecto a menos que este requisito se haya cumplido.

Capítulo 2

Refácil

```
El proyecto R también es nuestro.
```

1 ***Objetivo: R representa el fruto del esfuerzo continuado de una comunidad que se ha propuesto ofrecer una alternativa gratis a los paquetes estadísticos profesionales de alta calidad y que son supremamente costosos. Sus productos rivales pueden ser mejores en cuanto a la comunicación con el usuario, pero R les gana a casi todos en su riqueza de servicios que se pueden expandir con mas de 2500 paquetes. Hay versiones de R para Windows, Linux y Mac. En el presente capítulo aprenderemos a instalarlo y a manejar sus servicios básicos. En los capítulos siguientes, R será la herramienta de uso natural, aunque algunas cosas se harán con Excel o Gnumeric.***

2 ***Gnumeric y Linux***

Para análisis univariado, usar *R* puede resultar incómodo al menos al principio. Lo mejor es usar un paquete adecuadamente simple. Nuestra propuesta es **Gnumeric**, que es gratis y muy profesional. Sin embargo, no todos comparten la misma idea y hay cursos de estadística básica con *R*. Un ejemplo: *Elementary statistics with R* by Chi Yau (2010)

```
http://www.r-tutor.com/
```

Debido a que *R* representa un esfuerzo patriótico multinacional de liberarse de las grandes compañías productoras de Software y que pueden cobrar muy caro (varios miles de dólares por un producto con una licencia), *R* combina bien con **Linux** que es gratis y le falta poco para ser tan amigable para el usuario como Windows. Nosotros trabajamos con la versión **OpenSuse**, pero en general, no se espera que haya problemas delicados por causa del sistema que cada quien use. En realidad, hay una fuerte competencia por simplicidad y efectividad en todas las versiones de Linux y no es raro oír aplausos a una cualquiera. Fedora, Debian, Ubunto, Kubunto son otras versiones de Linux muy famosas.

2.1. ¿Qué es *R*?

El proyecto *R* es la versión de software libre de *S-plus*, y cuyo objetivo es producir software de alta calidad para quehaceres científicos centrados en la estadística.

Tanto *Gnumeric* como *R* vienen sin garantía. Eso quiere decir que pueden tener bugs, errores de programación que causará que algún resultado salga erróneo. Lo que uno tiene que hacer es mirar los datos a ver si la respuesta tiene sentido. O verificar la función deseada probándola con un ejemplo de un libro o que uno ya haya hecho a mano. O por medio de una simulación. Si hay discrepancia, lo más probable para este momento de la historia es que uno descubra que sus cuentas a mano están mal hechas y entonces uno estará muy agradecido, como en mi caso. Pero también es posible encontrar un bug, para lo cual uno lo reporta en el link que a propósito se tiene en la página web oficial de cada aplicación. Así fue como el autor de este material encontró y reportó para *Gnumeric* el bug 614746 sobre la documentación del ZTEST, el cual fue aceptado, y para *R* el bug 14316 sobre regresión polinomial, el cual no fue aceptado sino que se trataba de una falta mía de no haber leído bien la documentación.

Originalmente *R* se diseñó para ambientes Unix y Linux, dirigidos por instrucciones por consola, pero en la actualidad se trabaja en la versión orientada a ventanas, en la cual todo se hace por medio de un click y, como veremos al final del capítulo, ya tenemos versiones muy buenas. Hay versiones para Microsoft Windows, Linux y Mac.

2.2. Instalación

Para instalar *R*, búsquelo en Google o acceda a la siguiente dirección:
http://www.r-project.org/
Allí, busque un link que diga download *R*, después elija un servidor y a continuación una versión apropiada a su sistema, Linux, Windows o Mac. Si tiene problemas instalando una versión para 64 bits, elija una de 32 y vuelva a tratar. Si tiene problemas con la última versión disponible, elija una anterior.
Si su sistema es OpenSUSE Linux:

1. Conéctese a

   http://software.opensuse.org/search

2. Déle al buscador la orden de buscar

   R-base.

3. Después, haga click en un botón que diga

   1-Click Install.

   Siga las instrucciones y de la clave del Administrador cuando el sistema lo requiera.

4. El sistema debe terminar reportando una instalción exitosa.

   El sistema puso en algún lugar un directorio llamado RProject y ahí guardó todo lo referente al paquete.

En el suguiente link hay instrucciones visuales para instalar R sobre Windows y una potente GUI (graphic user interfase) llamada *Tinn-R* y que cada quien puede probar:

`http://bioinformatics.ualr.edu/resources/tutorials/Tinn-R_installation.html`

Nuestras instrucciones que están más abajo, son para otra GUI llamada *R Commander*, la cual sirve para todas las plataformas.

2.3. Iniciando *R*

El paquete *R* necesita un directorio o folder para trabajar. Por favor, primero cree un directorio o folder llamado *RProjectU1* (el sufijo *U1* significa usuario 1), donde pondrá todo lo referente a *R*. A continuación, cree otro directorio dentro de *RProjectU1* y póngale nombre *RWorkU1*.

Para iniciar *R*:

- Sobre Windows, *R* se inicia con un click sobre el ícono de *R*. Debería indicarle a *R* el camino para llegar a *RWorkU1*.
- En Linux, abra una consola o terminal y simplemente teclee *R*.

Cuando se abra la ventana o consola de *R*, éste saluda ofreciendo sus credenciales y dando algunas indicaciones. Por ejemplo que para salir de *R* se debe teclear q(). Después de saludar, *R* espera órdenes. Para ello, escribe el símbolo
>
al cual añade un cursor, que puede ser un bloquecito negro. Si el cursor de *R* es un bloquecito sin relleno, uno debe hacer click sobre la ventana que lo contiene para poder activar el editor, y el cursor se rellenará de negro. Después uno teclea lo que desee.

3 ***Ejercicio*** *Verificar que para salir de R basta teclear*

```
q()
```

Por favor, recuerde que siempre que se le de una orden a R, debe terminarse con

Enter

Por favor, salga de R y vuelva a entrar. Al tratar de salir, R preguntará si desea salvar algún archivo de trabajo. Responda con una n de not. En una futura ocasión, cuando haya trabajado y desea grabar su trabajo, responda y de yes.

4 ***Ejercicio*** *Corremos el demo de gráficas. Para ello, se escribe*

```
demo(graphics)
```

En respuesta, R pone un título y solicita permiso para seguir. Uno lo autoriza. R promete dibujar una gráfica, para lo cual abrirá una ventana, pero quizá no se vea nada. Uno corre la ventana recién creada para un lado, de tal manera que uno pueda ver la ventana de edición, la que debe tener el signo $>$.

R pedirá permiso para una segunda gráfica. Uno accede: para ello, uno hace click sobre la ventana de la terminal o consola, la ventana de R propiamente dicha y oprime Enter. La ventana graficadora se habrá minimizado. Dicha ventana se llama R Graphcis: Device 2. Uno la maximiza haciendo click sobre ella y uno deberá encontrar una gráfica. Así mismo se pueden ver las demás gráficas. Cuando R termine su demo reportará en la ventana de R:

```
par(oldpar)
```

Después de ésto, R estará listo para ejecutar otra orden.

5 ***Correr programas***

A *R* se le dan instrucciones verbales en un lenguaje apropiado, el cual hay que aprender trabajando. En el presente material hay muchos programas ya hechos. Para correrlos, no los teclee. Uno teclea sólo aquellos programas que uno mismo hace. Un programa ya hecho, se copia al clipboard y se pega a la consola de *R*. Algunos programas cortos pueden copiarse y pegarse desde este documento pdf, pero para programas largos surge un problema y es que las instrucciones para navegar en el documento también se insertan. Por ello, hemos preparado un archivo aparte *comandosVol2* que se puede bajar desde el mismo sitio que el codumento pdf y se puede abrir y editar con cualquier procesador de palabra. Seleccione el programa deseado y péguelo a la consola de *R*.

2.4. Comunicación con R

Aprendamos cómo se le hace llegar los datos a *R* y cómo se apodera uno de los resultados.

6 ***Definición local***

Uno puede definir directamente sobre *R* los datos. Hay que tener presente que toda definición se hace mediante la función c() que concatena o pone en cadena los datos y con la función $<-$ que asigna un nombre al conjunto de datos. Esta flechita se compone del signo menor que $<$ seguido de un guión corto $-$. Ejemplo:

```
#INTRODUCCION DE DATOS
x <- c(1,3.2,4,5,2.3,5,4,3,4,5,7,2)
# listar x
x
x[1] + x[4]
```

El programa anterior genera un vector de datos llamado x y que contiene los datos listados y los lista. Para listar un objeto simplemente se teclea su nombre. El comando x[1] + x[4] suma el primer elemento de x con el cuarto. La respuesta debe ser 6. Copie el programa anterior (que está en el archivo de *comandosVol2*) al clipboard y péguelo en *R*. Siempre que pegue un programa debe terminar con *enter*.

Verifique que se puede copiar todo el programa completo, pegar a *R* y ejecutar (*R* ejecuta en secuencia). Esto permite hacer librerías personales de comandos en *R*, las cuales se guardan en un archivo de texto y para usarlas se usa copiar y pegar.

Cuando uno copia y pega programas, quizá se tenga un programa hecho para un objeto x pero uno necesita correrlo sobre z. La solución es hacer un clon de z y llamarlo x:

```
#REUSO DE PROGRAMAS
#Tenemos z
z <- c(1,3.2,4,5,2.3)
#clonamos z sobre x
x<-z
#Programa que procesa x
x[1] + x[4]
```

Este programa, y casi todos, también está en el archivo de comandos. De allí se copia y se pega en la consola de *R*.

7 *Tecleando datos desde el editor*

Para hacerle llegar los datos a *R* también contamos con la siguiente instrucción que abre un editor de datos y asigna un nombre a los datos tecleados. Uno teclea los datos en la ventana emergente y para terminar sobre Linux hace click sobre *quit* o en Windows hace click derecho en el mouse y elige *cerrar*. Es posible que la manera de cerrar, la cual graba lo tecleado, cambie ligereramente de acuerdo a la versión.

La instrucción para abrir el editor sobre Linux:

```
#EDITOR DE DATOS SOBRE LINUX
#Abrir el editor (se graba al cerrarlo con quit)
sueldoA <-edit(data.frame())
#Publica sueldoA
sueldoA
```

La instrucción para abrir el editor sobre Windows:

```
#EDITOR DE DATOS SOBRE WINDOWS
#Abrir el editor
#Se graba cerrándolo con click derecho en el mouse + cerrar
sueldoA <-edit(as.data.frame(NULL))
#Publica sueldoA
sueldoA
```

Uno también puede ponerle títulos a cada columna, borrando el encabezado que dice $V1$ y poniendo el nombre deseado. Se puede hacer modificaciones de cualquier casilla pero antes hay que borrar todo lo que allí haya con la tecla *backspace*.

En un primer ejercicio, teclee simplemente una columna de datos. Si mas luego quiere teclear datos con muchas columnas, amplie la ventana del editor, para lo cual arrastrar su borde derecho es la mejor opción. También se puede navegar con el tabulador.

8 ***Corregir datos***

Si hemos usado el editor de datos para introducir unos datos y queremos hacer una corrección, podemos usar la siguiente idea:

```
#CORRECION DE DATOS
#Ya se tiene sueldoA
sueldoB <- edit(sueldoA)
t.test(sueldoA,sueldoB)
```

9 ***La media de un vector de datos***

Si queremos hallar la **media** de la variable *sueldoA*, que nos da una muestra aleatoria de sueldos de la persona A, usamos:

```
#LA MEDIA
#Programa en dos partes, corra primero una después la otra.
#Parte 1:
sueldoA <- c(1, 3.2, 4, 5, 2.3)
mean(sueldoA)
summary(sueldoA)
var(sueldoA)
#desviación
sd(sueldoA)
#dibuja sueldoA
plot(sueldoA)
#
#Parte 2
boxplot(sueldoA)
```

Detalle técnico: **boxplot** muestra la media y sus cuartiles vecinos. No muestra el error estándard. Para saber más, teclee

```
#AYUDA
#Teclee q para salir de la ayuda
help(boxplot)
boxplot.stats
```

Sobre la ayuda se avanza con la teclas para avanzar y retroceder sobre página. Para salir de la ayuda, teclee *q*.

10 ***Histogramas***

Los histogramas en R son muy naturales. En referencia al siguiente programa, el procedimiento *seq* tiene 3 parámetros: donde se empieza el histograma, donde se termina y el ancho de cada rango. El parámetro *prob* dice si la envolvente se hace en frecuencia absoluta o en relativa: puede tomar dos valores T y F. Uno cambia del uno al otro y se queda con el que más le guste. El comando *lines* presenta una curva que se ajusta al histograma. El grado de suavización se ajusta con *bw* (bandwidth): si uno quiere una curva con muchas arrugas, debe ser pequeño, pero si quiere una envolvente muy suavizante, debe ser grande.

```
#HISTOGRAMA
x<-c(1,2,1,3,2,3,2,4,3,2,4,2,2,1)
hist(x, seq(0.5, 4.5, 1), plot = TRUE, prob=T)
#Envolvente del histograma
lines(density(x, bw=1))
#Densitómetro de barras
#rug= representación unidimensional gráfica
rug(x, side=1)
```

Para utilizar un histograma en un documento sobre Windows: haga click derecho sobre la ventana de la gráfica, elija la opción de copiar al clipboard y péguela en el documento. Hay dos opciones, mire a ver cuál le gusta más. Sobre Linux es más complicado, al menos por ahora, y se explicará más abajo.

11 *Un test con la t*

Supongamos que queremos poner a prueba la idea de que el sueldo promedio de A es 1.2. Para ello se teclea

```
#TEST HO SOBRE LA MEDIA
sueldoA <- c(1,3.2,4,5,2.3)
t.test(sueldoA,mu=1.2)
```

Para uno aceptar o rechazar la hipótesis nula, de que la media es 1.2, uno mira el *p-value*. Si es más grande que 0.05, uno la acepta, pero si es menor que 0.05, R le está diciendo a uno que los datos que uno tiene son extremos con respecto a la H_o y que sería mejor buscarse otra explicación, por lo cual uno podría rechazar la H_o.

12 *Comparación de medias*

Si de antemano se sabe que las varianzas de las poblaciones de las cuales provienen los datos son iguales, para comparar medias poblacionales usamos

```
t.test(sueldoA,sueldoC, var.equal = T)
```

pero si dichas varianzas son diferentes, se puede especificar

```
t.test(sueldoA,sueldoC, var.equal = F)
```

Si los datos son apareados:

```
t.test(sueldoA,sueldoC, paired = T)
```

Si uno desea cambiar el nivel de confianza o hacer varias especificaciones:

```
t.test(sueldoA,sueldoC,conf.level = 0.99)
t.test(sueldoA,sueldoB,paired = T, conf.level = 0.99)
```

Si uno tiene más preguntas, uno puede pedir ayuda a ver qué logra:

```
help(t.test)
```

Para salir de la ayuda, teclear q de quit, que significa salir.

El siguiente programa tiene todo y está en el archivo de comandos:

```
#PRUEBA t PARA COMPARAR MEDIAS
sueldoA <- c(1, 3.2, 4, 5, 2.3)
t.test(sueldoA,mu=1.2)
sueldoB <- c(2.1, 1.3, 3.1, 2, 1.5)
t.test(sueldoA,sueldoB)
sueldoC <- c(2, 2.1, 3.1, 1.2, 1.5, 2.3, 1.7, 1.8)
t.test(sueldoA,sueldoC, var.equal = T)
t.test(sueldoA,sueldoC, var.equal = F)
t.test(sueldoA,sueldoC,conf.level = 0.99)
#Para un test de datos apareados, mismo número de elementos
t.test(sueldoA,sueldoC, paired = T)
t.test(sueldoA,sueldoB,paired = T, conf.level = 0.99)
#Avance sobre la ayuda con el navegador de páginas
#en el teclado.
#Para salir de la ayuda, teclee q
help(t.test)
```

13 *Test de varianzas*

¿Cómo se sabe que las varianzas poblacionales son iguales o diferentes? Por medio de un **test de varianzas** (ya no se puede decir test F porque la F servirá para muchas cosas diferentes) :

```
sueldoA <- c(1, 3.2, 4, 5, 2.3)
sueldoB <- c(2.1, 1.3, 3.1, 2, 1.5)
var.test(sueldoA, sueldoB)
var.test(sueldoA,sueldoB, conf.level = 0.99)
```

14 *Aumentando el rigor*

Los ejercicios sobre comparación de medias y varianzas entre dos columnas de datos se hacen más fácil en *Gnumeric* que nos permite comparar varianzas y diferenciar entre datos independientes y apareados. Pero si queremos aumentarle rigor a nuestros análisis, R puede comenzar a parecer interesante. Por ejemplo, para poder aplicar un test t debemos verificar que los datos vienen de una distribución normal. Para un **test de normalidad** es mejor tener bastantes datos. De momento, lo mejor es generarlos usando un generador interno. Usamos el **test de Kolmogorov-Smirnov**. Todo queda como sigue:

```
#TEST KOLMOGOROV SMIRNOV SOBRE NORMALIDAD
#Parte 1: Esto se corre primero
#Generamos 500 observaciones al azar de la normal,media = 40, sigma = 3.
w<-rnorm(500, mean = 40, sd = 3)
#Histograma de w
hist(w, seq(28, 52, 1), prob=T)
#Envolvente del histograma de datos
lines(density(w, bw=1))
#Envolvente según la Ho: función densidad de la normal, media = 40, sigma = 3.
#add = TRUE significa que la curva se añade a la gráfica anterior.
#add = FALSE significa que se crea una gráfica nueva.
curve((1/(2*pi* 9)^0.5) * (exp( -(x-40)^2/(2*3^2)) ) , 28,52, add = TRUE, col = "red")
#Densitómetro de barras
rug(w, side=1)
#
#Parte 2: Esto se corre (copia y pega)  después
#
#La gráfica K-S
qqnorm(w)
#La Ho (normalidad) daría una línea.
qqline(w)
#Test de Kolgomogorov - Smirnov para normalidad
ks.test(w, "pnorm", mean=40, sd=3)
```

Tengamos presente que sobre el dibujo, uno puede percibir anormalidad si los datos se desvían de la línea recta dibujada por el procedimiento *qqline()*. Esta metodología, de hacer muchos dibujos, es inmanente y necesaria a R y a todo investigador: en proyectos complejos está por descontado que uno puede equivocarse y por éso se requiren medios expeditos de retroalimentación. Nuestra manera predilecta será hacer gráficas y digerirlas al máximo: R se hizo precisamente para ésto, para que uno pueda estar seguro de lo que cree por estar apoyado en una base amplia de factores diversos. Pensando en ello, deberíamos hacer algo para mejorar nuestro programa anterior: hagamos el histograma y la envolvente de los datos según la H_o, la hipótesis de normalidad. Tengamos en cuenta que **la fórmula de la densidad de la normal** con media μ y desviación σ es:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

El contraste entre lo que se cree en la H_o, la normalidad, y lo que se ve en los datos se mide por un estadígrado llamado D y su significancia con dos colas es reportada como el p-value (la significancia es con dos colas porque D involucra un valor absoluto, que no distingue si hay exceso o hay defecto). El **p-value** es la significancia de nuestros datos, es decir, es la probabilidad de que por azar se halle un valor más extremo que el nuestro. Cuando los datos se ajustan a una normal, el p-value debe dar grande, mucho más grande que 0.05, lo cual dice que por puro azar y bajo la hipótesis de normalidad, es perfectamente corriente que se obtengan valores de p-value mucho menores, de datos mucho más extremos, es decir, tendiendo a ser anormales.

Si hacemos lo mismo con la distribución uniforme, obtenemos:

```
#TEST KS SOBRE DISTRIBUCION UNIFORME
#Esto se corre primero
#Generamos 500 observaciones al azar de la uniforme
w<-runif(500)
 #Histograma de w
hist(w, seq(-3, 3, 1), prob=T)
#Envolvente según la Ho: densidad de la normal
curve((1/(  sd(w) * ((2*pi)^0.5)  ))  *
  (exp( -(x-mean(w))^2/(2*(sd(w))^2)) ) ,
  -3,3, add = TRUE, col = "red")
#Densitómetro de barras
rug(w, side=1)
#
#Esto se corre después
#
#dibujo K-S
qqnorm(w)
#Si fuese perfecta normalidad daría una línea.
qqline(w)
#Test de Kolgomogorov - Smirnov para normalidad
ks.test(w, "pnorm", mean = mean(w), sd = sd(w))
```

15 *Tablas*

Si uno va a hacer una prueba t para comparar las medias de dos vectores, uno debe tener a la mano dos vectores de datos. Pero si uno desea comparar la media de 3 o más vectores de datos, entonces hay que tener en cuenta que los datos pesan más juntos que separados. Eso significa que cuando los datos se analizan juntos, las decisiones son más contundentes, que cuando se analizan por separado. Por tanto, cuando tenemos 3 o más columnas de datos, debemos organizarlos en un único objeto, una **tabla de datos**. Para poner tablas a disposición de R hay varios caminos .

1. Teclear la tabla con el editor interno, el cual sirve igual de bien para teclear un vector o una tabla.
2. Formar una tabla a partir de vectores existentes.
3. Leer la tabla de un archivo, para lo cual uno tiene que grabarlos previamente.

Para convertir uno o más vectores de la misma dimensión, mismo número de datos, en una tabla, uno adecúa y ejecuta el programa siguiente:

```
#CREAR TABLA A PARTIR DE VECTORES
GastosA<-c( 1,3,4,2,3,5,6,4,3)
GastosB<-c( 2,3,1,2,4,3,4,2,1)
#Unimos los datos en un sólo objeto
z<-data.frame(GastosA,GastosB)
z#publicamos la tabla en la consola.
```

Para manejar los elementos de una tabla hay varias maneras:

```
#MANEJO DE TABLA NUMERICA
a<- c(2001,2002,2003,2007,2010)
b <- c(2001+1,2002+2,2003+1,2007+3,2011)
t <- data.frame(a,b)
#Diversas publicaciones
t
t[1]
t[1,1]
t[5,2]
```

```
t$a
#Sume
t[1,1] + t[5,2]
t$a[1] + t$b[5]
```

16 ***Cambio de tipo numérico a factor***

Cuando uno ha codificado una variable con números, *R* lo interpretará como variable cuantitativa, numérica. Pero quizá a uno le interese que sea entendido como factor, es decir como variable categórica. Por ejemplo, el año 2001 no es un número sino el año en que nos fue bien. Y el 2005 fue cuando tuvimos recesión. Con estas connotaciones, los años no son números sino factores.

Para cambiar una variable de tipo numérico a factor, se conserva la variable antigua pero se crea un duplicado de tipo factor.Para vectores:

```
#CAMBIO DE TIPO: DE NUMERICO A FACTOR  (CATEGORICO)
#Si uno tiene un vector
#Se copia un vector de números a otro de tipo factor
a<- c(2001,2002,2003,2007,2010)
b <- as.factor(a)
#Los números se pueden sumar
a[1]+a[2]
#Los factores no:
b[1]+b[2]
```

para cambiar el tipo de una columna dentro de una tabla:

```
#CAMBIO DE TIPO DE UNA COLUMNA DENTRO DE UNA TABLA
a <- c(2001,2002,2003,2007,2010)
b <- c(2004,2005,2006,2007,2009)
tabla <- data.frame(a,b)
tabla
#Cambiar  a de numérico a factor:
#se copia a en c, pero como factor.
#pero se deja a como numérico.
#La simbología padre$hijo
#denota el hijo de tal padre, en este caso
#c es hijo (columna) de tabla.
tabla$c <- as.factor(tabla$a)
tabla
tabla$a[1] + tabla$b[1]
tabla$a[1] + tabla$c[1]
```

17 ***Tablas de contingencia en Gnumeric***

Las **tablas de contingencia** tradicionales se estudian igual de fácil tanto en *Gnumeric* como en la *GUI* de*R*. En el prier caso: *tools* + *statistics* + *contingency tables* + i*ndependence test*. En el segundo caso: en *statistics* se busca *contingency tables* y luego *enter and analyze two way table*.

R a través de la consola ofrece poderosas facilidades para el estudio de tablas de contingencia como muestra el siguiente ejemplo:

Supongamos que uno desea probar que los perros más viejos son más talentosos que los jóvenes. Entonces uno registra en un vector *edad* y otro *puntaje* que bien puede ser cuantitativo o categórico. El puntaje de nuestro caso es cuantitativo. Se sobreentiende que *edad* y *puntaje* se aparean naturalmente. Si uno mira los datos, uno se da cuenta que en realidad los perros viejos tienen mejor puntaje. *R* nos ayuda de la siguiente forma: el particiona los datos en dos clases, conforme se lo pedimos en el procedimento *cat* (categorizar) y con el parámetro *breaks = 2* (número de particiones). En segundo lugar, las particiones se hacen de tal forma que la probabilidad de hallar dependencia entre los los dos factores *edad* y *puntaje* sea lo máximo posible. Los rangos de la particiones que *R* hace aparecen en la tabla. Cuando hay muchos datos queda bien tomar un número mayor de categorías. Después se ejecuta el test usual chi-cuadrado.

```
#TABLAS DE CONTINGENCIA DE 2 DIMENSIONES
edad   <- c(1,2,3,4, 5, 6, 7, 8)
puntaje<- c(8,7,8,10,20,20,16,17)
#Partimos edad en dos categorías
edad.cat<-cut(edad,breaks=2)
#Partimos puntaje en dos categorías
puntaje.cat<-cut(puntaje,breaks =2)
#Calculamos la tabla de contingencia
d<-table(edad.cat,puntaje.cat)
#Publicamos la tabla
d
#Hacemos un test chi-cuadrado de independencia
chisq.test(edad.cat,puntaje.cat)
```

La H_o de una tabla de contingencia es que los dos factores son independientes. Cuando R ejecuta este programa produce como output el p-value = 0.03389 que es menor que 0.05. Por lo tanto, nuestros datos son extremos en el mundo de la chi-cuadrado y podemos tomar la determinación con dicha significancia de no aceptar la H_o de la independencia de los factores y considerar que los factores edad y puntaje son dependientes: a pesar de tener pocos datos, predecimos que hay alguna relación entre edad e inteligencia. El test no nos dice más, pero si uno mira la tabla producida por R, uno ve que todos los perros jóvenes clasifican en el puntaje más bajo y todos los perros viejos lo hacen en el puntaje más alto. Por tanto, tenemos derecho para alegar que los perros más viejos son más inteligentes.

R dice también que no está muy seguro de su aproximación, lo cual es una forma educada de decir que hay un prerequisito que no se cumple. En este caso, que hay una o más casillas con menos de 4 elementos y lo mínimo recomenddo es 5. Además, hay que tener en cuenta que el medidor de discrepancia entre lo que se ve y lo que se cree para tablas de contingencia no se distribuye exactamente como una chi-cuadrado sino sólo aproximadamente. Para tablas 2×2, hay una corrección usual debida a Yates (Sokal, 1981) que es muy popular y que es utilizada por R.

Sokal RR, Rohlf FJ (1981). Biometry: The Principles and Practice of Statistics in Biological Research. Oxford: W.H. Freeman

18 *Grabando datos*

Una manera cómoda de hacerle llegar información a R es **grabár datos** desde una hoja de cálculo, quizá *Excel* o *Gnumeric*.

Si los datos están en *Excel*, se regraban como tipo *text (MS-DOS)*. Si están en *Gnumeric* se graban como tipo *txt* y se responde *OK* a todo lo demás.

Atención: R maneja variables cuantitativas (numéricas) y categóricas (factores). A veces se crea ambivalencia de manera inadvertida. Por ejemplo: hay una variable Prom (promoción) que indica el año en que salieron graduados unos profesionales. Es natural que uno codifque el año de Prom en números, digamos el 2001. Como consecuencia, R tomará la columna Prom como los datos de una variable cuantitativa. Cada vez que uno quiera hacer un análisis, es necesario decidir si la forma como uno ha codificado sus variables corresponde a los propósitos que uno tenga en el momento. La GUI que usaremos diferencia entre factores y variables cuantitativas muy claramente y a los factores les asigna el papel de variables explicativas y a las variables cuantitativas les asigna el papel de variables respuesta o de salida.

Para fijar ideas, consideremos un estudio que evalúa los sueldos promedios de unos profesionales. Los datos se presentan en la tabla siguiente. Una tabla como esta corresponde a una planilla de datos, tal como se llena en el campo, o en las encuestas o en los experimentos. Por éso, a este tipo de tablas las llamamos *naturales*. Para la *GUI*, las variables Prof, Univ y Gen son factores (variables explicativas) en tanto que Prom, Sueldo y estrato son posibles variables respuesta. La tabla es la siguiente:

```
Prof       Prom  Univ  Gen    Sueldo  Estrato
Civil     2006    U1    M     2       4
Ind       2004    U2    H     3       5
Ind       2003    U2    M     2       2
Elec      2001    U3    H     1       3
Sist      2003    U1    M     2       4
Civil     2001    U2    H     3       6
Sist      2004    U3    M     2       6
Elec      2003    U2    M     1       3
Sist      2001    U2    H     2       2
Civil     2002    U3    M     3       3
```

Si uno desea, uno puede teclear los datos y salvarlos como archivo *txt*. Pero si uno quiere ahorrarse la tecleada, la forma de hacerlo es la siguiente:

Busque estos datos en el archivo adjunto, el que contiene los comandos, cópielos al clipboard, péguelos en *Gnumeric* y grábelos con nombre *tsueldos* y de tipo *txt*. Para salvar el archivo en *Gnumeric* usamos la opción *File + Save as*. Cuando se despliegue un cuadro de diálogo, elija el tipo de archivo en *File Type* como *Text*. Déle un nombre a su archivo, por ejemplo, *tsueldo*. Grábelo. A las preguntas restantes responda de la manera más simple o ignórelas (haga click sobre Save).

Inmediatamente después de grabar, revise el camino para llegar a su archivo *tsueldo.txt* y anótelo. En Linux puede ser algo así:

```
AJose/RProjectU1/RWorkU1/tsueldo.txt
```

Sobre Windows, un camino podría ser al estilo

```
C:/AJose/RProjectU1/WorkU1/tsueldo.txt
```

19 *Leer datos en R*

Podemos ahora **leer los datos** desde *R*:

Sobre Linux:

```
#IMPORTACION DE DATOS DESDE UN ARCHIVO TIPO txt sobre LINUX
tsueldos = read.table(file("AJose/RProjectU1/RWorkU1/tsueldos.txt"), encoding="latin1")
#Forma sinónima
tsueldos = read.table(file("AJose/RProjectU1/RWorkU1/tsueldos.txt"))
tsueldos #para listar un objeto, se teclea su nombre
```

Sobre Windows:

```
#IMPORTACION DE DATOS DESDE UN ARCHIVO TIPO txt sobre WINDOWS
#header = T significa que cada variable tiene un título
#como encabezado; header = F, significa que no hay título.
tsueldos <-read.table(file
  ("C:/AJose/RProjectU1/RWorkU1/tsueldos.txt"), header = T)
#para listar un objeto, se teclea su nombre
tsueldos
```

Modifique esta instrucción teniendo en cuenta sus sistema operacional y cambiando el camino al archivo por el camino verdadero de su archivo *tsueldo*. Es mejor hacer las modificaciones o ediciones en algún procesador de palabra. Copie su nueva instrucción al clipborad, active con un click a *R*, y péguela. Oprima Enter. Verá a continuación que *R* despliega los datos que leyó.

20 *Exportando resultados*

Uno puede ver los resultados que produce R pero ¿qué hacer si uno los desea exportar a un documento personal?

Existen varias maneras. La primera es negrear con el cursor la zona que uno desee copiar, copiarla al clipboard y luego pegarla en el documento que uno previamente tiene abierto para incluir los resultados.

Otra manera es el menu de *R* y buscar en algún lugar una opción que le permita grabar el output. Puede ser el menu *scrollback* seguido de *save output*. Elegir estas opciones lo lleva a uno a crear un archivo, el cual se actualiza a voluntad. Quizá uno quiera utilizar el mismo archivo como mesón de trabajo: en ése caso es necesario tener presente que cada vez que se inicializa *R* y se ordena grabar el output, uno borrará todo lo anterior. O bien, cada sesión debe originar su propio archivo con su propio nombre.

21 ***Gráficas***

Si uno desea una única gráfica que despliegue información de varios vectores de datos:

```
#VARIOS VECTORES DE DATOS EN LA MISMA GRAFICA
#Generamos los datos de unos gastos
ProyA<-c( 1,3,4,2,3,5,6,4,3)
ProyB<-c( 2,4,1,3,4,3,4,2,1)
#Unimos los datos en un sólo objeto
z<-data.frame(ProyA, ProyB)
#Gráfica:
#Dibujamos los datos en pantalla
#col = colores, 2 = rojo, 3= verde.
matplot(z,axes=F,frame=T,type='b',ylab="",col = 2:3)
# Se añaden los Títulos
title(ylab="Gastos", xlab = "Meses")
title("Gastos por mes \n proyecto A  y B")
#Caja de letreros: posición, contenido, lty = line types, colores
legend(1, 5, c(paste(" = ", c("A","B"))), lty = 1, col = 2:3)
#Dibuje la escala en el eje X
axis(1)
#Dibuje la escala en el eje Y
axis(2)
```

Si uno desea que varias gráficas aparezcan en la misma página:

```
#VARIAS GRAFICAS POR PAGINA
#cargar paquete
data(LifeCycleSavings, package="datasets")
#Listar datos
LifeCycleSavings
#
#Todas las gráficas en la misma página,
#formando una matriz de  3  filas y 2 columnas.
#par = modifique parámetros
par(mfrow=c(3,2), pch=16)
#Dibuje histogramas:
#LifeCycleSavings es al archivo de datos que contiene
#las variables sr, ddpi, pop15, pop75, dpi:
hist(LifeCycleSavings$sr,  plot = TRUE, breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$ddpi,  plot = TRUE,  breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$pop15,  plot = TRUE, breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$pop75,  plot = TRUE,  breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$dpi,  plot = TRUE,  breaks="Sturges",
  col="darkgray")
#Retorne a modo gráfico 1 x 1
par(mfrow=c(1,1), pch=16)
```

22 ***Grabando gráficas***

Sobre Windows:
Se pueden copiar al clipboard y de allí pegar a donde uno desee.

Sobre Linux:
Grabar una gráfica es dibujar en un archivo la gráfica y por éso se usa exactamente las mismas instrucciones en ambos casos. El código para la consola es una adaptación del siguiente, pero éso también puede hacerse desde la GUI,

```
#GRABAR GRAFICAS
#Parte 1
#Generamos los datos
GastosA<-c( 1,3,4,2,3,5,6,4,3)
GastosB<-c( 2,3,1,2,4,3,4,2,1)
#Unimos los datos en un sólo objeto
z<-data.frame(GastosA,GastosB)
#Dibujamos los datos en pantalla
matplot(z,axes=T,frame=T,type='b',ylab="")
#Títulos
title(ylab="Gastos", xlab = "Meses")#label for y-axis;
title("Gastos por mes \n proyecto A  y B")
#Se inicializa un archivo con formato png
#que ocupa poco espacio.
png(file="AJose/RProjectU1/RWorkU1/graf1.png")
#Dibujamos los datos en el archivo
#matplot: función que dibuja funciones
matplot(z,axes=F,frame=T,type='b',ylab="")
#Títulos
title(ylab="Gastos", xlab = "Meses")#label for y-axis;
title("Gastos por mes \n proyecto A  y B")
#Cerramos el archivo
dev.off()
#
#Parte 2
#Ahora grabamos un histograma
HGastosA<-hist(GastosA)
#Dibujo en pantalla
plot(HGastosA,main = paste("Histograma de GastosA"))
#Se inicializa un archivo con formato png
#que ocupa poco espacio.
png(file="AJose/RProjectU1/RWorkU1/graf2.png")
#Dibujamos los datos en el archivo
plot(HGastosA,main = paste("Histograma de HGastosA"))
#Cerramos el archivo: device off
dev.off()
#Terminar
#El archivo se puede editar con un procesador de imágenes.
```

Este programa creará una figura en el folder mencionado. El nombre de la figura es *graf1.png* y se abre y edita con cualquier aplicación que procese gráficas.

2.5. R es tan fácil como Excel

Aprendamos a trabajar con R por medio de una potente GUI(Graphic User Interface). Una GUI es una interfase entre un programa y el usuario que ha sido diseñada para ser operada preferencialmente con un click. Podremos trabajar con R con tanta facilidad como se trabaja con *Excel* o con *Gnumeric*.

23 *Instalando la GUI*

La GUI que probaremos se llama *R Commander* y se instala desde *R*, dado que previamente uno se ha conectado a Internet. Para instalarla en Windows, uno busca en el menu de *R* que diga *instalar paquete*, elige un servidor y el paquete *Rcmdr* y lo instala. Para instalarlo en Linux, uno debe entrar como administrador antes de llamar a *R*, y una vez con los derechos adecuados, uno llama a *R* e inserta la siguiente orden, la cual también sirve para Windows

```
# INSTALAR EL R COMMANDER, UNA GUI PARA R
install.packages("Rcmdr", dependencies=TRUE)
```

Después de oprimir *Enter*, *R* empieza un arduo trabajo que en Linux puede llevar quizá media hora. Uno puede almorzar entre tanto. Una vez terminado el trabajo, R abre la GUI. Pero si no lo hace, se abre con el comando

```
#ENCENDER EL R COMMANDER
library(Rcmdr)
```

R tratará de montar el *R Commander* y seguramente alegará que falta algún archivo. Lo mejor es decirle que no, que no lo instale, a no ser que se necesite expresamente. O uno puede decirle que si a todo y ayudarle a escoger, por ejemplo, un servidor adecuado de donde bajar el archivo. Tiempo para un café. Con la versión que tengo actualmente de *R* sobre Linux, esta reclamadera lo hace siempre como si fuese la primera vez.

Cuando termine, uno debe buscar una nueva aplicación que estará abierta quizá detrás de la ventana de *R*. Y estaremos listos para experimentar. Uno puede explorar la barra de menus y darse cuenta de las opciones disponibles. Después de descubrir que tenemos la opción de trabajar con *R* con la misma facilidad con que se trabaja en *Excel*, uno se pregunta ¿para qué sirve acceder a *R* desde la consola de comandos si uno tiene una *GUI*? Pues hay tres casos importantes:

1. Cuando uno tiene tareas repetitivas y complejas: uno las resuelva una vez, graba el programa correspondiente (lo cual también puede hacerse con la *GUI*) y después invoca uno el programa cada vez que lo necesite.

2. Cuando uno requiera una modalidad de un paquete ya montado sobre la GUI pero que no esté implementado sobre ella.

3. Cuando uno necesite un paquete que no esté implementado en la *GUI*. Recordemos que *R* tiene como 2000 paquetes a nuestra disposición. Atención: si tiene pensado instalar *R* y sus paquetes en muchas máquinas, baje lo que necesite una sóla vez, guárdelo en una carpeta apropiada y de allí saque todas las copias que necesite. Si no hace éso, puede correr el riesgo de una penalización por demasiadas visitas innecesarias al sito de *R*, que se llama **cran** (cran significa *comprehensive R archive network*).

Mientras que uno trabaja con la *GUI*, uno minimaliza la ventana de *R*. Para salir de *R*, se teclea q(). Si uno mata a R sin haber cerrado la *GUI*, puede tener problema con ella que queda como espectro molesto e inútil.

24 *¡Y ahora qué hago?*

La primera tarea que uno puede hacer es introducir un vector de datos y sacarle la media. Para ello, uno activa el menu *Data* y señala *New data Set*. Se abre el editor de datos y uno le pone el nombre a sus datos, por ejemplo, *sueldoA*, y rellena una columna con los datos, que se ha desplegado. Se termina con *Quit*. Después de ello, uno va al menu *statistcs* y busca alli *summary* or *numerical summary* y lo ejecuta. En el output uno encontrará la media y seguramente la varianza. La GUI permite hacer correcciones a voluntad, para ello es suficiente hacer click en el botón *Edit data set* y preceder en correspondencia.

Dado que uno tiene un columna de datos, uno mira aquí y allá hasta que encuentre el menu apropiado para hacer una prueba t para poner a prueba la creencia de que la media vale tanto y con la significancia requerida.

Uno tiene la opción de editar varios objetos de datos. ¿Con cuál de todos trabajará *R*? Para elegir el objeto de trabajo, hay una casilla en donde aparece uno de los nombres de los objetos de datos. Si se hace click sobre dicha casilla, se despliega un diálogo en el cual uno puede activar uno de los objetos y con ése es que *R* trabajará.

Bueno, ¿qué hacer si queremos estudiar la regresión de un primer vector sobre otro? ¿O una comparación de medias?; ¿O una comparación de varianzas?

Atención: *R* Commander es una *GUI* inteligente, lo cual significa que ofrece sugerencias de acuerdo a los datos que uno tenga. La forma como hace las sugerencias es simple y directa: las opciones de menu que estén activada y que aparecen en negro reteñido son las opciones que el *R* Commander sugiere para hacer. Las opciones que no corresponden quedan desactivadas en gris claro.

Precaución: ser inteligente es algo difícil. Por ahora, la inteligencia de la *GUI* no es muy alta y es mejor confiar en uno mismo, o uno tiene que exponerse a hacer cosas sin sentido que la *GUI* y *R* le permitirán.

25 *Instalación de paquetes*

La CRAN (la casa madre de R) tiene paquetes para todo lo que uno se imagine, para hacer las cosas mucho mejor, y para mil cosas más. Hay alrededor de 2500 paquetes para escoger en la siguiente dirección:

```
cran.r-project.org
```

Para uno adueñarse de un paquete, primero se baja e instala el paquete escogido conjuntamente con sus dependencias. Luego viene la activación que debe hacerse en cada sesión y puede hacerse por comandos en la consola o por medio del *R-comander* en *tools* → *load packages*.

Para bajar paquetes sobre Windows:
se usa el menu de R: si el paquete aparece listado, se procede por inercia. Si el paquete no aparece listado, se baja la versión para Windows desde la CRAN. Para ello, uno busca *cran* en cualquier motor de búsqueda, digamos Google, y de allí se conecta a la página web de *R*. Sobre dicha página, uno busca un link que lo lleve a los paquetes (packages). Siguiendo dicho link, uno llega a una lista alfabética de paquetes y allí busca el paquete deseado. Al buscar, hay que tener presente que la lista distingue mayúsculas de minúsculas tanto en la primera letra como en las demás y que los números van primero que las letras. Uno puede examinar la lista de paquetes con una descripción de media línea sobre su función.

Cuando uno haya bajado el paquete y lo haya puesto en alguna carpeta, vuelve a usar el menu de R para instalarlo. Y después ya lo puede correr.

Para bajar e instalar paquetes en Linux:
Primero que todo, uno busca *cran* en cualquier motor de búsqueda, digamos Google, y de allí se conecta a la página web de *R*. Sobre dicha página, uno busca un link que lo lleve a los paquetes (packages). Siguiendo dicho link, uno llega a una lista alfabética de paquetes y allí busca el paquete deseado. Al buscar, hay que tener presente que la lista distingue mayúsculas de minúsculas tanto en la primera letra como en las demás. Los números van primero que las letras. La lista de paquetes viene con una descripción de media línea sobre su función.

Para instalar un paquete en Linux: cuando uno haya bajado y salvado su archivo en una carpeta apropiada, uno abre una terminal encima de dicha carpeta y con derechos *su* (superuser), uno teclea el comando de instalación.

Para fijar ideas, pensemos en montar la maquinaria para correr correlación canónica sobre *R*. Lo primero que hacemos es instalar el gcc-fortran, que normalmente viene con todas las distribuciones de Linux. Para ello se abre una terminal de Linux, se teclea *su* (superuser) se da la clave y se copia+pega el siguiente comando sobre la terminal:

```
zypper install gcc-fortran
```

Una vez que se ha instalado gcc.fortran, uno ya puede bajar e instalar los iguientes paquetes:
fda, spam, fields, catspec, CCA

Una vez bajados y puesto en la carpeta downloads, abrimos una terminal encima de downloads, tecleamos *su*, y la clave y después usamos copie+pegue sobre los siguientes comandos, en el orden respectivo (por favor, cambiar las versiones, las cuales vienen con el nombre de cada archivo):

```
R CMD INSTALL fda_2.2.2.tar.gz
R CMD INSTALL spam_0.22-0.tar.gz
R CMD INSTALL fields_6.01.tar.gz
R CMD INSTALL catspec_0.95.tar.gz
R CMD INSTALL CCA_1.2.tar.gz
```

Muchos paquetes son grandes proyectos que se van haciendo poco a poco y que reciben mejoras con frecuencia. Para actualizar los paquetes, *R* tiene la opción de actualizarlos todos con una sóla orden:

```
update.packages()
```

26 ***Para saber más sobre R***

- *R* permite hacer trabajo con mucho detalle, para expertos, incluyendo imponentes gráficas, que se pueden confeccionar a todo dar. Uno puede convencerse de éso echándole una ojeada al siguiente documento:

  http://math.illinoisstate.edu/dhkim/Rstuff/Rtutor.html

- Hay una exhuberante galería de gráficas con su código en el sitio siguiente:

  ```
  http://addictedtor.free.fr/graphiques/
  ```

- Un muy bien lugar para profundizar sobre las opciones de *R* en estadística:

  ```
  http://www.statmethods.net/advstats/index.html
  ```

- Es posible aunque improbable que *R* se atasque debido a un volumen excesivamente grande de datos, por ejemplo, al estudiar macroproyectos. Si éso llegase a pasar, es necesario tener en cuenta que manejar grandes cantidades de datos es de por sí una profesión y la cual se implementa con una base de datos. La predilecta por los usuarios de *R* parece ser una versión apropiada de **MySQL** (My es el nombre de una niña, hija de uno de los desarrolladores) que se llama, por supuesto, RMySQL y que se baja de la sección de paquetes de

  ```
  http://cran.r-project.org/
  ```

  Las instrucciones para conectar *RMySQL* a *R* pueden encontrarse en el link *R data Import/Export* de la ayuda en línea de *R* que se llama con

  ```
  help.start()
  ```

- Para saber más sobre la *GUI*, un puede conectarse al siguiente sitio:

  http://socserv.mcmaster.ca/jfox/Misc/Rcmdr/

  donde uno puede encontrar un link a un *Introductory manual*. En la ayuda de la GUI también hay un link al mismo documento.

- El poder de *R* es enciclopédico con una excelente bibliografía en línea. Para saber qué tiene *R* para determinado tema, se va a la página de la *cran* y se busca el menu *Search*. En la ventana emergente uno busca *R site search* y allí teclea el tema. En respuesta se desplegará toda una serie de opciones a elegir. Si no da nada, de seguro es que hay mala ortografía.

Capítulo 3

El método científico

`Contrastar lo que se ve con lo que se cree`

27 ***Objetivo****: Entender sobre un ejemplo el modus operandi de la ciencia moderna, la cual se identifica con la estadística.*

3.1. Ciencia en el seno familiar

La estadística moderna pueda quizá equipararse a la ciencia actual y en consecuencia luce muy sofisticada. Lo es tanto que es usual tener que usar la estadística sin tener la menor idea de de por qué hace las cosas que hace.

En realidad, la estadística moderna es una abstracción de la forma usual de ser de todos los humanos. Para demostrarlo, analicemos un caso en el cual todos estaremos listos a reconocernos. El tema lo adaptamos del Vol 1.

28 ***Ejemplo*** *Estadística por instinto.*

Patricia sale a las 2 de clase y va llegando a su casa a eso de las tres y media. Rara vez llega más tarde que las cuatro. Ella se queda un poco en la U después de clase para poder hacer en grupo algunas tareas. Hoy es jueves. Son las 4 y Patricia no ha llegado. Pero éso a veces pasa, sigamos esperando. Patricia no ha llegado y ya son las cuatro y cuarto. La mamá ya anda nerviosa, pero se despreocupa después de una llamada de Patricia que le confirma que está haciendo tareas. Llegan las 6 de la tarde pero Patricia no aparece y sus llamadas no tranquilizan a la mamá, la cual ya ha comenzado a llamar a todas las amigas de Patricia, piensa llamar enseguida a la policia, luego al papá y al tío Paco. Ya no le podemos creer que se trata tan sólo de hacer tareas. Es evidente que a Patricia algo le pasó o algo está ocultando. Reformulemos esa historia en términos oficiales:

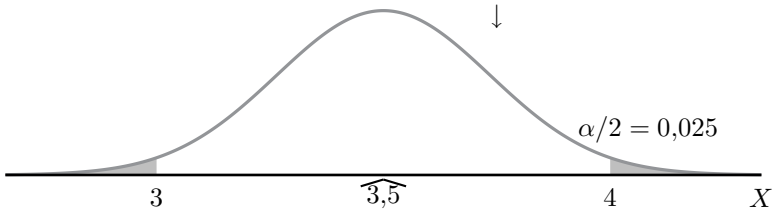

Figura 3.0. X es la hora de llegada de Patrica a su casa. Cuando $3 < x < 4$, la mamá toma una decisión: quedarse tranquila. Cuando $x < 3$ o $X > 4$ se toma otra: prender las alarmas.

Sea X la hora de llegada de Patricia a su casa. X tiene una distribución de tendencia central que caricaturizamos como una normal. La media está en 3.5 y cuadramos la desviación para que el 95 % de las veces, llegue entre las 3 y las 4. El 5 % de las veces llega antes de las 3 o después de las 4. Eso quiere decir que el 2.5 % llega antes de las tres y el otro 2.5 % después de las cuatro. Al mirar la tabla de la

Ψ, vemos que a $\Psi = 0{,}025$ le corresponde una $z = 2$. Eso significa que hay dos desviaciones estándar desde las tres hasta las tres y media y otras dos desde las tres y media hasta las cuatro. Por lo tanto, la desviación vale un cuarto de hora. En resumen: $X \sim N(3.5,\, 1/4)$. Si Patricia llega tarde hay problemas. Pero si llega a casa demasiado temprano, la mamá comienza a pensar si fue que tal vez ella no hizo las tareas. Así que mal si llega temprano, mal si llega tarde. Oficialmente decimos que trabajamos con *dos colas*.

El contexto nos dice que hay barreras que dividen lo usual, lo tolerable, de lo inusual, lo intolerable. Esas barreras son las 3 de la tarde y las 4 y dentro de esas barreras está el 95 % de las veces de llegada de Patricia a su casa. Nosotros hemos elegido una significancia del 0.05. Si un evento es normal de acuerdo con dicha significancia, tomamos una decisión: la mamá se queda tranquila. Pero si un evento se clasifica como anormal, la mamá toma otra decisión y en el caso de mucho retardo ella comienza a llamar a las amigas de Patricia, a la Policía, al papá y al tío Paco.

Hemos usado un procedimiento para decidir una de dos explicaciones. La primera: Patricia se quedó haciendo tareas. La segunda: a Patricia le pasó algo o algo está ocultando. Ahora aprendamos a formalizar todo éso:

Lo usual se denomina la H_o, la **hipótesis nula**. En este caso, lo usual es que Patricia se queda haciendo tareas, a veces se queda mucho, a veces se queda poco. Por éso, la hora de llegada X es aleatoria y $X \sim N(3.5,\, 1/4)$. El dato que queremos contrastar contra el transfondo de la H_o se llama **dato experimental**, $x = 6$. Tengamos en cuenta que pueden aparecer datos extremos que inviten a pensar que sucedió algo raro y que incentive la búsqueda de explicaciones alternas, que no forman parte del mundo habitual que se considera en la H_o. Esa opción se denomina la H_a, **hipótesis alterna**. En la H_a se especifica si la negación de la H_o se hace con la cola superior, con la cola inferior o con ambas colas.

La hipótesis nula dice que $X \sim N(3.5,\, 1/4)$. La H_a dice que si un evento es extremo con relación a una significancia dada, lo mejor es inventar otra explicación. La forma oficial de decirlo es:

$H_o : \mu = 3{,}5$ (dado que $\sigma = 1/4$, el dato es normal si está cerca de 3,5)
$H_a : \mu \neq 3{,}5$ (dato extremo con dos colas, invente otra explicación aparte de la H_o).

Con una significancia $\alpha = 0{,}05$, **la región de aceptación de la** H_o es el intervalo (3,4). **La región de rechazo de la** H_o es lo que queda por fuera: $x < 3$ ó $x > 4$.

Como el dato experimental es $x = 6$, rechazamos la H_o: somos partidarios de decir que hay algo anormal en lo que le está pasando a Patricia y que hay un factor extraño que ha causado un desajuste. Por ejemplo, Patricia se quedó encantada con un amigo hablando de la vida.

Aclaración: Si la H_o se rechaza con la cola superior, lo cual dice que el dato experimental se considera extremo del lado superior, entonces la H_a es

$H_a : \mu > 3{,}5$ (dato extremo con la cola superior, invente otra explicación aparte de la H_o).

Si la H_o se rechaza con la cola inferior, lo cual dice que el dato experimental se considera extremo del lado inferior, entonces la H_a es

$H_a : \mu < 3{,}5$ (dato extremo con la cola inferior, invente otra explicación aparte de la H_o).

29 ♣ ***Definición.*** *Una proposición es una* H_o***, hipótesis nula,*** *cuando cualquier diferencia entre los datos experimentales y la* H_o *puede ser explicada cómodamente por el azar. Decimos que una proposición es una* ***hipótesis alterna****,* H_a*, cuando ésta niega a la* H_o*. Si hay una explicación para algo, entonces debe ser explicado o bien por la* H_o *o bien por la* H_a*, pero no por ambas. Por eso, hay que decidir cuál de las dos hipótesis explica el fenómeno propuesto. Si se rechaza la* H_o*, se está diciendo que debe haber un factor que crea un efecto sistemático que hace que los resultados se desvíen de lo predicho por la* H_o*. En otras palabras, estamos prediciendo que en circunstancias similares la misma gran discrepancia entre resultados y la* H_o *serán observados con una alta probabilidad que se hará más real entre más datos se consigan.*

3.2. El azar

Hemos dicho que la H_o es lo usual, lo cual ha sido descrito por la hora de llegada de Patricia a su casa: $X \sim N(3.5,\ 1/4)$ que nos dice que ella llega a casa aproximadamente a éso de las tres y media. La ciencia ha adoptado una lectura e interpretación muy oficial de esto: la conducta de Patricia puede entenderse como el resultado de dos componentes, el primero es determinístico y si no hubiese nada más Patricia llegaría a las tres y media. La cosa es tan seria que cuando el papá tiene que intervenir le dice: *La señorita Patricia María Maldonado me llega a casa a las tres y media. Y si nó, se las tiene conmigo.*

No vamos a negar que el papá sea algo anticuado pero, éso si, de bruto no tiene nada y sabe que hay que ser un poco tolerable: hay factores que hacen que Patricia llegue un poquito antes o un poquito después. Nadie está en capacidad de controlar dichos factores, como que haya o no trancón. Estos factores definen el segundo componente que se denomina azar. Oficialmente, el **azar** consiste en todas las cosas que no controlamos y que hacen que los datos experimentales se desvíen de su valor promedio. El azar estaría descrito en nuestro caso por una normal con media cero y desviación 0.25.

Podemos enunciar la H_o como sigue: de no ser por el azar, Patricia llegaría a las tres y media. Aceptar la H_o significa que usamos el azar para explicar la discrepancia entre el dato observado y lo esperado. Rechazar la H_o significa que un evento es tan extremo con relación a lo usual que uno por instinto tiende a rehusarse a aceptar que sucedió por a azar y se empeña en buscar otra explicación causada por algún agente desconocido que causa o puede causar efectos sistemáticos.

3.3. Simulación en R

Ayudémonos con R para entender que es lo que las altas matemáticas hacen para implementar la parte operativa de la estadística, en este caso contrastar una hipótesis nula con un dato experimental. Nuestro ejemplo es el más sencillo posible pero toda la estadística por más sofisticada y compleja que parezca es simplemente más de lo mismo (mas altas dosis de creatividad operativa).

Al utilizar las tablas ya estamos usando las altas matemáticas. Por éso, lo que haremos será hacer un trabajo equivalente a sacar una tabla. En este caso, la tabla de interés es la tabla de la $\Psi(z)$ que da el área bajo la campana normal que queda hacia el lado izquierdo de z.

Nuestra hipótesis nula representa la conducta usual de Patricia cuya llegada a casa se describe como $X \sim N(3.5,\ 1/4)$ y tenemos un dato experimental $x = 6$. Lo que debemos hacer es decidir si ante este dato experimental debe seguirse haciendo lo que usualmente se hace, esperar con paciencia a Patricia, o si debe hacerse algo especial, como llamar a la policía. Nuestro criterio de decisión será

Aceptar la $H_o : \mu = 3{,}5$ (dado que $\sigma = 1/4$) si el dato experimental está cerca de 3,5. O bien, rechazar la H_o , buscarse otra explicación y prender las alarmas si el dato experimental está demasiado alejado de 3,5.

Pero, ¿qué signfica estar cerca o lejos 3,5? Para la distribución normal estar cerca de 3,5 significa estar más cerca que dos desviaciones estánard. Y estar lejos significa estar más lejos que 2 desviaciones. Si queremos aplicar los mismos conceptos a distribuciones que no son simétricas como la F, debemos reformular los mismos conceptos en términos que permitan una fácil generalización. Esto se logra trabajando con la **significancia** que da la proporción de eventos que son más extremos que los nuestros. *Lo usual es trabajar con una significancia del 0.05 que nos dice que prenderemos las alarmas ante cualquier evento que sea tan extremo que tan sólo el 5 % de los eventos sean más extremos que el nuestro.*

Jerga: a la **significancia** asociada a un conjunto de datos también se le llama **p-value.** La generalidad de paquetes estadísticos reportan el p-value y nada más para que uno pueda tomar las decisiones correspondientes.

Veamos como luce este programa visto desde R. Para simplificar, consideremos rechazar la H_o con la cola superior que es la más alarmante y que sucede cuando Patricia se demora demasiado para llegar.

```
#==============================
#ESTUDIO DE UNA HIPOTESIS NULA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(1,1), pch=16)
#Parte 1
#Cálculo de la Ho.
#Parámetros de la normal
media <- 3.5
sigma <- 0.25
#Simulación:
#Generamos n observaciones al azar de la normal,
# media = 3.5, sigma = 0.25.
n <- 5000
w<-rnorm(n, mean = media, sd = sigma)
#Los datos generados pueden listarse
# w
#Histograma de w:
#Definimos intervalo a considerar
limInferior <- media - 10*sigma
limSuperior <- media + 10*sigma
#Ancho de cada rango del histograma
paso <- 0.1
#Calculamos y dibujamos histograma
hist(w, seq(limInferior, limSuperior, paso), prob=T)
#Envolvente según la Ho: función densidad de la normal
#add = TRUE significa que la curva se añade a la gráfica anterior.
#add = FALSE significa que se crea una gráfica nueva.
K <- 1 /( sigma * (2*pi)^0.5 )
curve( K  * ( exp( -(x-media)^2 / (2*sigma^2)) ),
   limInferior, limSuperior, add = TRUE, col = "red")
#Densitómetro de barras
rug(w, side=1)
#
#Parte 2:
#Contrastamos el dato experimental x= 6 con lo usual:
#vemos por el histograma que
#el dato es muy extremo.
#Nuestro dato no se explica bien por lo usual, por azar, y
#aconsejamos que se prendan las alarmas.
#
#Forma ofical de decir que nuestro dato es muy extremo:
#calculamos la significancia o el p-value con la cola superior,
#es decir,
#calculamos el porcentaje de eventos más extremos por arriba
#que el dato experimental observado. Eso se puede hacer
#matemáticamente o usando nuestra simulación
#que produjo los datos guardados en w:
#chequeo con x = 4
datoExp <- 4
cuenta <- 0
for (j in 1:length(w))
{
 if ( w[j] > datoExp) cuenta <- cuenta + 1
```

```
}
cuenta
pvalue <- cuenta / n
pvalue
#
#Test con x = 6
datoExp <- 6
cuenta <- 0
for (j in 1:length(w))
{
 if ( w[j] > datoExp) cuenta <- cuenta + 1
}
cuenta
pvalue <- cuenta / n
pvalue
```

Nuestra simulación nos dice que el porcentaje de eventos que por azar son más extremos que 4 es 2.28 %. Por otra parte, el procentaje de datos más extremos que 6 es 0 dado que se hizo una muestra de 5000 datos al azar. Si se usa *Gnumeric*, basado en altas matemáticas, el porcentaje exacto de eventos más extremos que 4, que corresponde a 2 desviaciones, es 0.02275013194818. El porcentaje de eventos más extremos que 6, que queda a 10 desviaciones arriba de la media, es 0. En ambos casos, llegada a las 4 o las 6, los datos son extremos y es natural que uno prenda las alarmas.

3.4. La ciencia se viste de gala

Lo que hicimos queda muy bien para estudiar la media sobre datos que que se distribuyen normalmente. Cuando uno tiene en mente no sólo la media sino también la varianza de distribuciones arbitrarias, uno se pregunta cómo ha de generalizarse lo hecho para que pueda servir en todos los casos. Lo sorprendente es que tal generalización existe y empieza con la caracterización siguiente:

La ciencia es un contraste, la medida de una discrepancia, entre lo que se ve y lo que se cree, entre lo que se observa y lo que se espera. Lo que se espera es lo habitual, lo que pasa con la mayoría, es la H_o. Si la discrepancia entre lo que se ve y lo que se espera es pequeña, se acepta lo que se espera. Si la discrepancia es grande, se rechaza lo que se espera y se busca otra explicación. El número de colas de rachazo se consigna en la H_a.

Podemos seguir un protocolo muy oficial como sigue:

Todos los problemas se resuelven por exactamente la misma metodología: se codifica el problema dentro de la camisa de fuerza del método científico que dice: **la ciencia es un contraste medido en unidades apropiadas entre lo que uno observa (los datos) y lo que uno cree (la hipótesis nula).** Luego se resuelve el problema codificado y su solución se reinterpreta en términos de los enunciados del problema. Se siguen los pasos siguientes:

PASO 1: Se estudia el enunciado del problema para formular la hipótesis nula y la alterna. La hipótesis nula dice lo que se cree. La H_o siempre contiene una igualdad, por ejemplo que una diferencia de medias vale cero. La hipótesis alterna especifica con cuantas colas se trabaja. Para decidir si se trata de una cola o de dos colas se mira el contexto del problema:

1. Si se enfatiza una igualdad, la H_o es la igualdad y la H_a es la desigualdad, la cual es con dos colas, pues hay dos maneras de ser desigual.
2. Si el enunciado pregunta sobre una desigualdad (ser mejor, más alto, más grande, más rico), la hipótesis nula es la igualdad y la alterna contiene el símbolo de ser mayor que y es con una cola pues dado un número hay sólo una manera como otro número puede ser mayor.

3. Si se dice: yo tengo el doble de puntaje promedio que Usted, se trata con dos colas, la hipótesis nula es la igualdad: mi promedio es dos veces el tuyo. La hipótesis alterna es la desigualdad.

4. Si se dice: yo tengo más del doble de puntaje promedio que Usted, se trata con una cola, la hipótesis nula es la igualdad: mi promedio es dos veces el tuyo. La hipótesis alterna contiene el símbolo de ser mayor que.

PASO 2: Se especifica lo que se ve, lo observado. Por ejemplo, que la diferencia de las medias de dos muestras es 8. Se ruega encarecidamente homogenizar las dimensiones: si el problema habla de días, meses y años, reducir todo, por ejemplo, a meses.

PASO 3: Se mide la discrepancia entre lo que se ve y lo que se cree, entre lo observado y lo esperado. Para ello se usa un estadígrafo de contraste adecuado, eligiendo el caso adecuado, por ejemplo:

1. Comparar un dato con una media, o una media muestral con una media poblacional sabiendo la varianza poblacional.

2. Comparar dos proporciones.

3. Comparar dos varianzas.

4. Comparar dos medias. Para ello, primero hay que comparar las varianzas. Si las varianzas poblacionales resultan iguales, se aplica un procedimiento, si son diferentes, otro.

5. Comparar las medias de datos apareados.

PASO 4: Se decide si la discrepancia medida es grande o pequeña. La discrepancia se define como pequeña cuando es más pequeña que el valor crítico de la discrepancia por puro azar, la cual se mira en las tablas. La discrepancia es grande si es mayor que la discrepancia crítica. Si uno usa el *p-value*, uno acepta la H_o cuando el p-value es más grande que la significancia de trabajo y uno rechaza la H_o cuando el p-value de los datos es menor que la significancia lo cual implica que nuestro dato es extremo conforme a la significancia escogida.

La discrepancia crítica depende de la significancia α que dice cuando un valor del estadígrafo considerado es ó bien extraño ó bien común y corriente. Si el problema no da la significancia, uno puede tomar la que uno quiera, de acuerdo a las tablas que se tengan a mano. Es usual tomar $\alpha = 0{,}05$ cuando no se menciona en el problema.

PASO 5: Para terminar se toma un veredicto: Si la discrepancia dada por el estadígrafo de contraste es pequeña, la hipótesis nula se acepta. Pero si la tal discrepancia es grande, la hipótesis nula se rechaza. Luego se reescribe la decisión en los términos literarios del problema. **Un problema sin veredicto no vale nada** pues hay que recordar que lo que importa son las decisiones y no los cálculos.

En resumen: la hipótesis nula dice que la población de datos no es homogénea y por tanto un proceso de muestreo aleatorio dará a veces un resultado y a veces otro. Por consiguiente, la diferencia entre lo esperado en la H_o y lo observado es un resultado del muestreo y no de un proceso sistemático. Decimos que la discrepancia resultante se debe al azar. La hipótesis alterna dice que aunque el azar en el muestreo nunca deja de funcionar, si la discrepancia entre lo esperado según la H_o y lo observado es demasiado grande, entonces es más cómodo interpretar dicha enorme diferencia como el resultado de un proceso sistemático, es decir como el resultado de la actividad que se haya hecho. Ejemplo: uno a veces se siente fuerte y a veces débil. Si uno come y se siente fuerte, quizá no sea por la comida sino porque a veces uno se siente fuerte y a veces débil y esta vez uno se sintió fuerte por puro azar y la comida no tuvo nada que ver. Pero si uno definitivamente se siente muy fuerte, uno tiende a pensar que es por la comida y no hay nadie que lo haga cambiar de opinión.

30 ***Ejemplo: estudiante engreído:*** *Un estudiante exhibe con orgullo su 3.2 en un examen en el cual las notas se distribuyen normalmente con media 2.3 y desviación 0.3. ¿Se justifica su jactancia?*

Paso 1: Formulamos lo que se espera (esperamos que el mundo se comporte como de costumbre, o como la gran mayoría). $H_o : \mu = 2{,}3$ y los datos experimentales deben estar por ahí cerca, sabiendo que la desviación vale 0.3. Es decir, todo el mundo debería sacar 2.3 de nota excepto por cosas del azar.

$H_a : \mu > 2{,}3$: además del azar, debe haber algún efecto sistemático que hace que al estudiante le vaya mejor que a la gran mayoría de la clase.

Paso 2: Dato experimental: la nota en el examen fue de 3.2. La desviación poblacional: $\sigma = 0{,}3$.

Paso 3: Medimos la discrepancia entre lo encontrado experimentalmente y lo esperado según la H_o. Para ello usamos una z:

$$z = \frac{x-\mu}{\sigma} = \frac{3{,}2-2{,}3}{0{,}3} = \frac{0{,}9}{0{,}3} = 3.$$

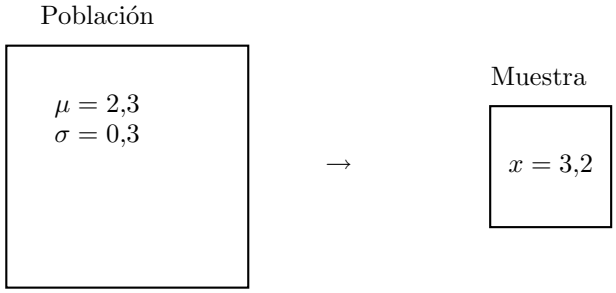

Figura 3.1. Hay una discrepancia entre la media de la población μ y el valor x: ¿se debe dicha discrepancia al azar, a la variación intrínseca de la población, o a algo sistemático?

Paso 4: Decidimos si la discrepancia 3 es pequeña o grande.

Para ello tomamos la significancia 0.05 con la cola superior que es 1.6. Si la discrepancia está por encima de ese valor, la discrepancia es grande y se rechaza la H_o: no es cómodo explicar el dato experimental diciendo que de no ser por el azar sería 2.3. Si la discrepancia es menor que 1.6, se considera fruto del azar.

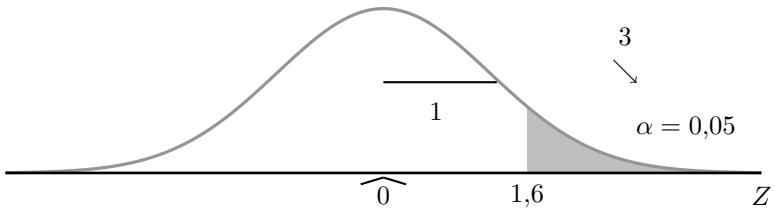

Figura 3.2. Si $H_o : \mu = 0$ en $Z \sim N(0,1)$, con la cola superior y $\alpha = 0{,}05$, la zona de aceptación de la H_o es lo que está abajo de 1.6. Como el dato experimental con $z = 3$ está afuera de tal zona, rechazamos la H_o.

Paso 5: Veredicto: Como la discrepancia $z = 3$ está por encima de 1.6, decidimos que la discrepancia es grande y la H_o se rechaza. Verbalicemos todo muy explícitamente. Tenemos que elegir entre dos plausibles explicaciones del dato experimental cuya discrepancia con respecto a la media es 3:

1. H_o : de no haber sido por el azar, el estudiante habría sacado el promedio de la clase, 2.3.
2. H_a : hay un factor sistemático que hace que al estudiante le vaya mejor que a la gran mayoría de la clase.

Por supuesto que entre las dos explicaciones elegimos gustosos la segunda: al muchacho le fue definitivamente bien, mejor que a la gran mayoría. Tolerémosle su jactancia.

Veamos como luce este proyecto visto desde una simulación en *R*:

```
#==============================
#LA Z COMO ESTADIGRAFO DE CONTRASTE
#Limpia la memoria
rm(list = ls())
par(mfrow=c(3,1), pch=16)
#Parte 1
#Cálculo de la Ho que dice que de no ser por el azar,
#cada evento debería ser la media.
#Parámetros de la normal
media <- 2.3
sigma <- 0.3
#Simulación:
#Generamos n observaciones al azar de la normal,
# media = 3.5, sigma = 0.25.
n <- 10000
w<-rnorm(n, mean = media, sd = sigma)
#Los datos generados pueden listarse
# w
#Histograma de w:
#Definimos intervalo a considerar
limInferior <- media - 5*sigma
limSuperior <- media + 5*sigma
#Ancho de cada rango del histograma
paso <- 0.1
#Calculamos y dibujamos histograma
hist(w, seq(limInferior, limSuperior, paso), prob=T)
#Envolvente según la Ho: función densidad de la normal
#add = TRUE significa que la curva se añade a la gráfica anterior.
#add = FALSE significa que se crea una gráfica nueva.
K <- 1 /( sigma * (2*pi)^0.5 )
curve( K  * ( exp( -(x-media)^2 / (2*sigma^2)) ),
    limInferior, limSuperior, add = TRUE, col = "red")
#Densitómetro de barras
rug(w, side=1)
#
#Parte 2:
#
#Definimos la discrepancia entre lo que se ve x
#y
#la media que es lo que se cree bajo Ho, :
#discrep (x) = (x - media)/sigma
#
#Parte 3:
#Estudiamos la distribución de la discrepancia.
#Inicialización
Discrep <- w
#Cálculo de la discrepancia
for (j in 1:length(w))
{
  Discrep[j] <- (w[j]   - media)/sigma
```

```
}
#Histograma de la discrepancia bajo la Ho
hist(Discrep, seq(-5,5, 0.1/sigma), prob=T)
#
#Envolvente según la Ho (altas matemáticas)
K <- 1 /(2*pi)^0.5
curve( K  * ( exp( -(x)^2 / 2) ), -5, 5, add = TRUE, col = "red")
#
#Parte 4
#
#Contrastamos el dato experimental x= 3.2 con lo usual:
#calculamos la discrepancia entre lo que se espera
#bajo la Ho y lo observado:
x <- 3.2
discrepExp <- (x   - media)/sigma
discrepExp
#Vemos por el histograma de la discrepancia bajo la Ho que
#el dato es muy extremo.
#Nuestro dato no se explica bien por lo usual, por azar, y
#aconsejamos que se prendan las alarmas.
#
#Forma ofical de decir que nuestro dato es muy extremo:
#calculamos la significancia o el p-value con la cola superior,
#es decir, sobre las discrepancias
#calculamos el porcentaje de eventos más extremos por arriba
#que la del  dato experimental observado. Eso se puede hacer
#matemáticamente o usando nuestra simulación
#que produjo los datos guardados en Discrep:
#
#Test con x = 3.2 cuya discrepancia es discrepExp
cuenta <- 0
for (j in 1:length(Discrep))
{
 if ( Discrep[j] > discrepExp) cuenta <- cuenta + 1
}
cuenta
pvalue <- cuenta / n
pvalue
par(mfrow=c(1,1), pch=16)
```

El p-value de la cola superior según nuestra simulación es 0.0013. El p-value de la cola superior según *Gnumeric* es 0.00134989803163.

Nuestra simulación dice que entre 10000 eventos al azar sólo 13 son más extremos que el observado. Rechacemos la H_o y que se prendan las alarmas: ¿ a qué se debe que ese estudiante sean tan bueno? ¿se copió? ¿es muy inteligente? ¿es muy estudioso? ¿qué cosas buenas debo yo imitar para lograr lo mismo o ser mejor?

Por complicado que sea un experimento, uno lo puede estudiar siguiendo los mismos pasos que hemos visto aquí. Se puede hacer matemáticamente o con simulaciones para lo cual R es grandioso.

Capítulo 4

Anovas

```
Anova = analysis of variance
```

31 ***Objetivo****: generalizar el test t para comparar medias entre cualquier número de columnas de datos.*

32 ***La gran idea***

Ahí está María. sentada, tranquila, toma el sol, toma un café. Los muchachos pasan, como que la miran, como que no. Ella ni lo nota. Pero de pronto va llegando Luis y ella se alborota, se altera, se pone colorada, hace que no ve, se rasca la cabeza, saca sus electrónicos de su morral, los vuelve a meter, se para un momento, se sienta y todo eso antes de que Luis haya alcanzado a llegar. Todo el mundo se da cuenta que Luis es para María terriblemente importante y que el resto de muchachos no tanto. Saquemos en claro que la forma como hemos logrado saberlo es la siguiente: la conducta de María se modifica notablemente cuando Luis va llegando en tanto que María permanece inalterada cuando los otros muchachos pasan.

Concluimos que un factor influye sobre un sistema cuando existe un **descriptor**, una variable medible y observable, que se llama **variable respuesta** y que cambia, que varía, cuando el factor cambia. En nuestro caso, el **factor** o **variable experimental** es la presencia de Luis, que puede tomar dos valores, presente o ausente, y el sistema es María a la cual le medimos la cantidad y extensión de movimientos que ejecuta por unidad de tiempo. Esta gran idea, de usar la variación de un descriptor para estimar el efecto de un factor sobre un sistema es de sentido común pero su depuración como fina y poderosa herramienta en estadística se debe a Fisher, y se denomina **análisis de varianza**. Es usual usar el anglicismo anova (analysis of variance) para designar el análisis de varianza. Podemos entender de qué se trata si estudiamos cómo se implementa para comparar varias medias de la variable respuesta.

4.1. Anova unifactorial

33 ♣ ***Definición.*** *Consideremos un* ***experimento*** *que estudia el comportamiento de una variable de respuesta de un sistema ante cambios de las condiciones en que se mantuvo y /o mantiene el sistema estudiado. Cuando se toman condiciones específicas que se comparan entre ellas tenemos* ***tratamientos****, que pueden ser, por ejemplo, cambios de nivel de una factor dado. Una* ***anova unifactorial*** *es es el análisis de la variación en las respuestas promedio ante cambios de tratamientos. Se propósito es evidenciar diferencias entre medias debidas a diferencias entre tratamientos. La idea es que si un sistema influye sobre un factor, entonces cuando uno varíe el tratamiento, la respuesta promedio del sistema debe variar. La anova estudia esta variación con el ánimo de determinar si el efecto de cambio de tratamiento es más fuerte que lo que llamamos azar o fluctuaciones del sistema y que son los cambios en la variable de respuesta debidos a todo lo que no controlamos sea porque no podamos o porque no queremos o porque nos sale demasiado caro.*

34 ***Ejemplo de la mora.*** *Consideremos un cultivo de mora que crece a diversas temperaturas. Medimos la cosecha en kg por metro cuadrado, suponiendo la misma densidad de plantas sobre tierra abonada con los mismo nutrientes. Si la temperatura afecta la producción de la planta, entonces debemos*

observar una variación de la producción promedio para diversos niveles de temperatura. Lo datos son los siguientes:

Kgs de mora por m^2		
9^o C	18^o C	25^oC
1	3	1
2	4	1
1	5	2

Aunque son pocos datos, inmediatamente entramos a sospechar que a 18^o hay más producción promedia que a 9^o o a 25^o pues la media en los dos últimos casos es 1.3 mientras que en el primero es 4 y además todos los valores de la producción a temperatura 18^o están por encima de los valores para los otros dos tratamientos. Quizá sea cierto lo que pensamos, pero probarlo no es tan fácil. El problema radica en que para cada temperatura hay una variación que se explica diciendo que hay factores no controlados que influyen sobre el sistema, por ejemplo, las variaciones genéticas que hacen a cada planta diferente de las demás. Por tanto, podemos formular la siguiente objeción: los cambios observados en la producción entre tratamientos no se deben al efecto de la temperatura sino al efecto de los factores no controlados que por azar generaron la impresión de que la temperatura influía sobre la producción. Haremos un estudio de la temática de forma general, y después lo especializamos a nuestro ejemplo.

35 ♣ *Especificaciones detalladas del diseño anova de una vía o unifactorial.* *Un diseño experimental tipo anova estudia el efecto de los diversos niveles o tratamientos de un factor que puede ser cualitativo o categórico sobre un sistema al cual se le mide una variable respuesta o de salida y que debe ser cuantitativa Y. Consideramos que hay g tratamientos y n_g representa el número de datos colectados para el tratamiento g. La suma de todos los datos da un n total: $n = \sum_{j=1}^{g} n_j$. Para denotar cada dato, usamos la notación matricial con dos subíndices, el primero se refiere al número que ocupa el dato dentro de su columna y el segundo denota la columna o tratamiento. Por ejemplo, y_{23} denota el dato 2 de la columna 3 y en general y_{ij} denota el dato i dentro del tratamiento j.*

Es natural suponer que hay muchos factores no controlados que inciden sobre el experimento. Como consecuencia se predice una variabilidad en los datos para cada tratamiento. Vamos a asumir que los factores que crean dicha variabilidad y el factor de estudio no se interfieren mutuamente y como consecuencia la varianza observada en los datos de cada uno de los tratamientos puede considerarse como una estimación de una varianza ideal, poblacional, que notaremos σ^2. Para el caso del ejemplo de la mora, pensamos que el principal factor no controlado es la variación génetica que hace que todas las plantas sean diferentes, al igual que todas las personas lo son. Adicionalmente asumimos que la forma como los genes se las arreglan para crear una planta no depende del factor estudiado, en nuestro caso la temperatura. Como consecuencia podemos decir que el dato y_{ij} se explica como la combinación de dos efectos: el efecto promedio del nivel j del factor de estudio, μ_j, y el efecto de todo lo demás no controlado, ϵ_{ij}. Lo notamos así:

$$y_{ij} = \mu_j + \epsilon_{ij}$$

El promedio μ_j es fijo, por supuesto, pero ϵ_{ij} es debida al azar, el cual no influye sobre la producción promedio, es una v.a. con media cero y varianza σ^2 que se considera independiente del tratamiento.

Lo que nosotros intuímos es que el factor estudiado influye sobre el valor promedio de la variable de respuesta, pero para probarlo necesitamos resolver la objeción planteada por σ^2 que en general no es cero: la diferencia observada entre las medias por columna no se deben necesariamente al efecto de cambiar de tratamiento sino al puro azar causado por los factores no controlados y porque tenemos un número finito de datos. Empaquetamos la objeción en la hipótesis nula que dice que de no ser por el azar las medias observadas sería iguales a la media ideal, poblacional:

$$H_o : \mu_1 = \mu_2 = ... = \mu_g$$

La hipótesis alterna dice que hay al menos un par de medias que sean diferentes, lo cual será la evidencia de que el factor influye el promedio de la variable respuesta.

Vamos a estudiar nuestra H_o y de paso estableceremos el lenguaje usado en análisis de varianza que permea toda la estadística.

Relacionemos los estadígrafos del experimento con los valores ideales, poblacionales. Primero que todo, hay una producción promedio global, μ que se estima por $\bar{y}_{\bullet\bullet}$, el promedio global:

$$\bar{y}_{\bullet\bullet} = \frac{1}{n}\sum_{j=1}^{g}\sum_{i=1}^{n_j} y_{ij}$$

donde se promedia sobre todos los datos, n. La primera suma corre sobre todos los g tratamientos y la segunda sobre los n_j datos recolectados para cada tratamiento.

En segundo lugar, tenemos μ_j, el promedio sobre el tratamiento j. Podemos estimar dicho promedio por los datos a mano que dan $\bar{y}_{\bullet j}$, el promedio de la columna j:

$$\bar{y}_{\bullet j} = \frac{1}{n_j}\sum_{j=1}^{n_j} y_{ij}$$

El significado de $\bullet$ se refiere a la matriz de datos y depende de su posición: si está en el segundo subíndice dice que el segundo subíndice ha dejado de ser una variable porque, por ejemplo, se ha promediado sobre ella y si está en ambos lugares es porque se ha promediado tanto sobre el filas, el primer subíndice, como sobre columnas, el segundo.

Ahora especifiquemos el azar o el ruido causado por los factores no controlados. Eso se logra mediante el siguiente truco:

$$y_{ij} = y_{ij} + 0 = y_{ij} + \bar{y}_{\bullet j} - \bar{y}_{\bullet j} = \bar{y}_{\bullet j} + (y_{ij} - \bar{y}_{\bullet j})$$

que nos dice que el dato y_{ij} es el resultado de un efecto promedio del tratamiento j mas una fluctuación o desvación de dicho promedio causada por el azar. Estamos diciendo oficialmente que $y_{ij} - \bar{y}_{\bullet j}$ es un estimador de ϵ_{ij}.

Por otra parte, nuestra H_o puede predecir $\hat{y}_{ij}$, el valor esperado promedio de nuestra observación y_{ij}. Para ello tomamos el promedio a ambos lados de la ecuación

$$y_{ij} = \mu_j + \epsilon_{ij}$$

para obtener

$$\hat{y}_{ij} = \bar{y}_{ij} = \mu_j + \bar{\epsilon}_{ij} = \mu_j + 0$$

pues el ruido o azar se llama azar porque en promedio no produce ni fu ni fa.

Ahora, pondremos atención a la variación de los datos.

Como definición de **variación total observada** tenemos un múltiplo de la varianza total observada:

$$\textit{variación total observada} = SST = \sum_{j=1}^{g}\sum_{i=1}^{n_j}(y_{ij} - \bar{y}_{\bullet\bullet})^2$$

donde hemos adoptado la costumbre de llamar a esta variación como suma de cuadrados totales que se nota SST (del inglés *total sum of squares*).

Por otro lado, la variación predicha por nuestra H_o es

$$\textit{variación total predicha por la } H_o = SSA = \sum_{j=1}^{g}\sum_{i=1}^{n_j}(\hat{y}_{ij} - \bar{y}_{\bullet\bullet})^2 = \sum_{j=1}^{g}\sum_{i=1}^{n_j}(\bar{y}_{ij} - \bar{y}_{\bullet\bullet})^2$$

donde *SSA* viene del inglés *sum of squares among groups* y que representa la variación entre las medias de los grupos o columnas ponderada al número total de datos por cada columna. También

hemos aplicado que $\bar{y}_{ij} = \bar{y}_{\bullet j}$, de acuerdo a la predicción de la H_o. Como en la columna j todos los n_j términos son iguales a $\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}$, tenemos

$$SSA = \sum_{j=1}^{g} \sum_{i=1}^{n_j} (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet})^2 = \sum_{j=1}^{g} n_j (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet})^2$$

Hemos demostrado que SSA, la variación predicha por el modelo, mide el efecto observable del factor sobre la variable de respuesta. En efecto, si un cambio de tratamiento influye sobre el promedio de la variable de respuesta, deberá haber una variación consistente entre las medias, y entonces la variación entre medias será no cero y de una magnitud tal que sobrepase la variación causada por el ruido. Si la H_o es cierta, SSA deberá ser oscurecida por la variación causada por el ruido: ¿pero cuál es esa variación? Es la que se observa cuando no hay cambio de tratamiento, dentro de cada columna, y que se debe a la única fuente de variabilidad, el ruido, y es la siguiente:

$$\textit{Variación causada por el ruido} = SSE = \sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet j})^2$$

donde SSE viene del inglés *error or total within group sum of squares* y que cuantifica el efecto de todo lo que no se controla en el experimento, y que se llama genéricamente **azar** y mide la variación de cada término con respecto a su media local, dentro de cada grupo.

Nuestro modelo divide el mundo de los factores en dos partes: el factor de estudio y todo lo demás no controlado. Por consiguiente deberemos esperar que la variación total se descomponga en dos partes, la variación debida al factor más la variación debida al azar. Es decir, deberíamos tener que

$$SST = SSA + SSE$$

¿será eso cierto?
Sí lo es, pero veamos cómo se aplican los supuestos para poder demostrarlo:

$$\begin{aligned}
&SST = \sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet\bullet})^2 \\
&= \sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} + 0 - \bar{y}_{\bullet\bullet})^2 \\
&= \sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} + (\bar{y}_{\bullet j} - \bar{y}_{\bullet j}) - \bar{y}_{\bullet\bullet})^2 \\
&= \sum_{j=1}^{g} \sum_{i=1}^{n_j} ((y_{ij} - \bar{y}_{\bullet j}) + (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}))^2 \\
&= \sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet j})^2 + \sum_{j=1}^{g} \sum_{i=1}^{n_j} (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet})^2 + 2 \sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet j})(\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) \\
&= SSE + SSA + 0
\end{aligned}$$

donde hemos expandido un binomio $(a+b)^2 = a^2 + b^2 + 2ab$ y además hemos aplicado el siguiente resultado:

$$\sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet j})(\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) = 0$$

lo cual se prueba así:

$$\begin{aligned}
&\sum_{j=1}^{g} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet j})(\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) \\
&= \sum_{j=1}^{g} (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) [\sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\bullet j})] \text{ (lo que no depende de i se puede sacar como constante)} \\
&= \sum_{j=1}^{g} (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) [\sum_{i=1}^{n_j} y_{ij} - \sum_{i=1}^{n_j} \bar{y}_{\bullet j}] \text{ (la sumatoria se reparte)} \\
&= \sum_{j=1}^{g} (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) [\sum_{i=1}^{n_j} y_{ij} - n_j \bar{y}_{\bullet j}] \text{ (pues } \bar{y}_{\bullet j} \text{ es constante y sale fuera, queda un 1, } n_j \text{ veces)} \\
&= \sum_{j=1}^{g} (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) [n_j \bar{y}_{\bullet j} - n_j \bar{y}_{\bullet j}] \text{ (porque } \bar{y}_{\bullet j} = (1/n_j) \sum_{i=1}^{n_j} y_{ij}, \text{ y por tanto } n_j \bar{y}_{\bullet j} = \sum_{i=1}^{n_j} y_{ij}) \\
&= 0.
\end{aligned}$$

Al probar que $SST = SSA + SSE$ estamos diciendo que

1. Nuestro modelo descompone la variación total en dos fuentes, el efecto de la variable experimental o bajo control mas el efecto del ruido, del azar, de todo lo que no se controla.

2. No hay interacción entre el ruido y la variable experimental responsable de SSA, estos dos factores son independientes.

Tratemos de entender mejor el significado de que el ruido y el factor estudiado son independientes, es decir, no tienen efectos conjuntos. El efecto del ruido se mide por los términos de la forma $y_{ij} - \bar{y}_{\bullet j}$, las desviaciones de cada dato con respecto a su media local, y los efectos del factor de estudio se miden por los términos de la forma $\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}$, las desviaciones de las medias locales con respecto a la media global. Como consecuencia de dicha independencia, la covarianza $\sum_{i=1}^{n_j}(y_{ij} - \bar{y}_{\bullet j})(\bar{y}_{.j} - \bar{y}_{\bullet\bullet})$ vale cero para cada i. Ahora bien: decir que el ruido y la variable experimental son independientes es una propiedad estadística que puede ser cierta o no pero nosotros hemos demostrado que siempre es válida. Lo que éso segnifica es que nuestro experimento anova unifactorial es realmente un instrumento de medida que no está capacitado para registrar una posible dependencia o interacción entre el azar y la variable o factor experimental. Pero en el momento de relacionar nuestro modelo con la naturaleza, la independencia entre el experimento y el azar es un postulado que se debe verificar y que no es tan secillo. Por ejemplo, si consideramos el efecto de los precios del fríjol sobre la nutrición de los niños, no es tan fácil creer que dicho precio es independiente de todo lo demás que pueda influír sobre la nutrición de los niños. Es suficiente pensar en el precio del maíz, de la papa y de la habichuela para darse cuenta que es improbable que sean independientes del del fríjol.

Recalquemos un punto que ha llamado mucho la atención de los filósofos de la ciencia:

36 ***Theory-ladeness (preconcepciones teóricas)***

La ciencia es un contraste entre lo que se ve y lo que se cree. Lo que se cree viene empaquetado en los modelos a priori que queremos verificar o especificar. Como consecuencia, los conceptos de verdad o falsedad no dependen de la naturaleza solamente sino de la interacción entre lo que que se cree, los modelos con los que atacamos al mundo, y lo que observamos. Como consecuencia, no es legítimo hablar de verdad o ciencia en términos absolutos sino tan sólo en relación a los modelos que se han estudiado. En nuestro ejemplo, el modelo predice que el experimento y el azar no están correlacionados y por lo tanto tal consideración entra como un supuesto al momento de especificar la descomposición $SST = SSA + SSE$. Este fenómeno se llama oficalmente **theory-ladeness** y significa que uno siempre llega a estudiar al mundo con modelos preseleccionados a priori, los cuales tienen supuestos necesarios que hay que tener en cuenta para que no haya otra fuente de contradicciones.

El caso más extremo de theory-ladeness es la ciencia moderna con sus paradigma oficiales, como la teoría evolutiva: hoy en día dicha teoría no es una teoría científica, que se pueda poner a prueba para aceptar o rechazar, sino que es el modelo socialmente obligante con que se observa toda la naturaleza, desde el neutrino hasta los huecos negros. Es exactamente lo mismo que pasaba hace 4 siglos: el lente que se usaba para mirar al universo en aquel momento era la teología bíblica con sus interpretaciones aceptadas en el momento. En ambos casos, hoy con la evolución o ayer con la teología de la Inquisición, se comete el gravísimo error de no tener en cuenta que toda verdad humana depende del modelo con que se empiece la discusión.

Volveremos sobre este tema. Por ahora, continuemos con la discusión de la anova unifactorial:

Hemos demostrado que $SST = SSA + SSE$, y que nos dice que el ruido y los tratamientos son independientes y que no producen efectos registrables conjuntos, de interacción. Ahora podemos entrar a medir de algún modo la discrepancia entre lo que se ve y lo que se cree según la H_o. Comenzamos recordando que SSA es un término sensible a los cambios de tratamiento. Si la H_o es cierta, SSA deberá quedar oscurecido por el ruido, SSE. Así que debemos comparar SSA con SSE. Pero el estadígrafo que mida la relación entre esos dos términos debería tener una distribución reconocida. Eso se logra transformando las sumas de cuadrados en varianzas y estudiando su relación en forma de quebrado para que de una F. Pero hay que cuadrar además un detalle técnico: debemos predecir cuánto debe valer el quebrado buscado bajo la suposición de que la H_o es cierta: de esa forma mediremos la discrepancia entre lo que se ve con lo que se cree. Dicho requerimiento se logra de manera espectacular pues podemos transformar SSE y SSA en estimadores insesgados de la misma varianza poblacional que no puede ser otra que la de σ^2, la varianza debida al ruido, pues bajo la H_o no puede haber más.

Decimos que un **estadígrado es insesgado** cuando el promedio da el valor teórico. Ejemplo:
Si tenemos una media poblacional igual a μ, el estadígrafo $\bar{X}_n = \frac{1}{n}\sum X_i$ es un estimador de μ. Pero al repetir el muestreo muchas veces y hacer el histograma de $\bar{X}_n$, uno podrá sacarle la media a dicho histograma y coincidirá con la media poblacional μ. Ese es el mensaje del teorema del límite central, el cual nos garantiza que $\bar{X}_n$ es un estimador insesgado de la media poblacional. Algo tan directo como lo encontrado para la media no se repite con la varianza. Para que la varianza de una muestra sea estimador insesgado de la varianza poblacional se requiere definir la varianza como

$$s^2 = \frac{\sum(x_i - \bar{X})}{n-1}$$

y se le da el nombre de varianza muestral para distinguirla de la varianza normal, la que divide por n, y que es insesgada.

Por igual motivo, al tranformar SSA y SSE en varianzas, se toma $SSA/(g-1)$ y $SSE/(n-g)$. Al último término se le llama **varianza error** que cuantifica el efecto del azar sobre el sistema. Ahora lo que tenemos que ver es que estas expresiones son estimadores de la misma σ^2, la varianza debida al ruido, al azar. Veámos: SSW mide la variación dentro de cada grupo que es efecto natural de la varianza debida al azar. Por tanto, todo lo que se requiere para hallar un estimador insesgado de σ^2 es dividir dicho término por un número apropiado, que resulta ser $n-g$. Por otro lado, SSA mide la variación debida a los cambios entre tratamientos. Pero bajo la H_o, ellos no influyen para nada, y por tanto SSA también tiene su origen en el azar. Como consecuencia, podemos dividir por un número apropiado para que nos de un estimador insesgado de la varianza σ^2. El número apropiado resulta ser $g-1$.

En resumen, tomamos el quebrado:

$$R_{exp} = \frac{SSA/(g-1)}{SSE/(n-g)}$$

que mide la relación observada entre los efectos del experimento y los del ruido. Pero, atención, para obtener una F nosotros debemos suponer que los datos tanto del numerador como del denominador vienen de un muestreo al azar de una población normalmente distribuída. Lo cual implica que ϵ, que representa el efecto del ruido, tenga una distribución normal con media cero y desviación σ^2. Si ése es el caso, y como tenemos estimadores insesgados de la misma varianza, podemos decir que la H_o predice que el valor esperado del quebrado R debe ser 1. Por tanto, el quebrado que mide la relación entre lo que se ve, R_{exp} y lo que se cree, $R = 1$, es igual a

$$F_{exp} = \frac{R_{exp}}{1} = R_{exp}$$

Puesto que venimos de poblaciones normales, la F_{exp} se distribuye como una F, pero hay que determinar los grados de libertad. Quisiéramos, por supuesto, decir que los grados de libertad de la F son $g-1$ en el numerador y $n-g$ en el denominador. Pero para poder decir eso, tenemos que demostrar que no hay redundancia de información de los dos grupos de datos, los que producen el numerador de los que producen el denominador. Pero eso lo garantizamos al decir que la covarianza entre los factores del experimento y el ruido son independiente, con covarianza cero. Si no tuviésemos dicha covarianza igual a cero, tendríamos que hacer cuentas engorrosas, quizá imposibles, para determinar cuántos grados de libertad se pierden.

Con todos los supuestos dichos, la F_{exp} se distribuye como una F con $g-1$ en el numerador y $n-g$ en el denominador.

Si ningún cambio de tratamiento influye sobre el valor promedio de la variable de respuesta, tendremos un valor de F_{exp} por ahí cerca de uno. Pero en la medida en que el experimento influya sobre la variable de respuesta del sistema, en dicha medida será grande el numerador de la R_{exp} y por tanto de la F_{exp} y por consiguiente cuando este estadígrafo sea mucho más grande que su valor crítico, nos sentiremos autorizados para creer que el experimento si influye sobre el valor promedio de la variable de

respuesta. Se deduce que la H_o, de que los factores controlados no influyen sobre la variable respuesta, se rechaza mediante F con una cola, la superior.

¿Qué hacer cuando queremos probar que un factor afecta el desempeño promedio de una variable de respuesta aunque débilmente? Propondremos una respuesta general a esta pregunta en el teorema fundamental del método científico, expuesto más abajo. Por ahora, resolvamos el ejemplo de la mora.

37 ***Ejemplo de la mora, análisis de varianza***

La tabla de datos es la siguiente:

9^o C	18^o C	25^oC
1	3	1
2	4	1
1	5	2

A una tabla con esta se le llama **tabla de conteo**.
Las medias por columnas o tratamientos son:
$\bar{y}_{\bullet 1} = (1+2+1)/3 = 1{,}33$.
$\bar{y}_{\bullet 2} = (3+4+5)/3 = 4$.
$\bar{y}_{\bullet 3} = (1+1+2)/3 = 1{,}33$.

La media global es la suma de todo, 20, dividido por 9 datos:

$\bar{y}_{\bullet\bullet} = 20/9 = 2{,}222$.

$SSA = \sum_{j=1}^{g} n_j(\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet})^2$
$= 3(1{,}33-2{,}22)^2 + 3(4-2{,}22)^2 + 3(1{,}33-2{,}22)^2 = 3(0{,}89)^2 + 3(1{,}78)^2 + 3(0{,}89)^2 = 14{,}25$

$SSE = \sum_{j=1}^{g}\sum_{i=1}^{n_j}(y_{ij} - \bar{y}_{\bullet j})^2$
$= (1-1{,}33)^2+(2-1{,}33)^2+(1-1{,}33)^2+(3-4)^2+(4-4)^2+(5-4)^2+(1-1{,}33)^2+(1-1{,}33)^2+(2-1{,}33)^2 = 3{,}33$

Teniendo en cuenta que hay 3 grupos y 9 datos, $g = 3$, $n = 9$

$F_{exp} = \frac{SSA/(g-1)}{SSE/(n-g)} = \frac{14{,}25/2}{3{,}33/6} = 12{,}83$

Los grados de libertad de la F son el numerador 2 y en el denominador 6.

Es muy sencillo lograr que *Excel* o *Gnumeric* nos hagan el análisis de varianza para los datos de la mora. El resultado en *Excel* luce como sigue:

Anova for three temperatures						
Origin of variation	SS	DF	Mean Square	F-Exp	Probab	Critical F
Among groups	14,22222222	2	7,111111111	12,8	0,0068453	5,143249382
Within groups	3,333333333	6	0,555555556			
Total	17,55555556	8				

Para usar *Gnumeric*, uno teclea los datos sobre una planilla, los negrea con el cursor, va a *tools*, luego a *statistical analysis*, luego a *anova*, y allí escoge *one way anova*. *Gnumeric* produce la misma tabla aunque la titula algo diferente, por ejemplo, en vez de *among groups* escribe *between groups*. Los valores de p-value y del valor crítico difieren por allá en el 4 decimal. *Gnumeric* tiene una fama de ser muy exacto y *Excel* no. Por eso, si se requiere mucha precisión, uno puede preferir las respuestas de *Gnumeric*, el cual acompaña la tabla con los valores de la media y de la varianza por columna.

¿Qué decisión tomamos con respecto a la mora? ¿Es o nó sensible a la temperatura? La tabla nos dice que el F crítico vale 5 y pico en tanto que el valor F_{exp} es mayor que 12. Vemos que nuestro experimento arroja datos que son extremos en un mundo donde todo se deba al azar. La probabilidad de que haya un valor más extremo que el nuestro es 0.0068 que es casi 10 veces menor que 0.05. Nos sentimos autorizados a creer que la prodicción promedio de mora es sensible a la temperatura. Y que a los 18 grados de temperatura promedio la mora parece sentirse muy bien.

A mucha gente le parece bonito justificar la decisión tomada sacándole provecho a todos los datos de la tabla de la anova. Para nuestro caso, podemos decir que la variación entre grupos en unas 4 veces mayor que la variacion debida al ruido, dentro de grupos. Y si promediamos por los grados de libertad, el efecto de la temperatura es 12 veces mayor que el efecto del ruido. Por supuesto, tiene toda la razón quien diga que la producción promedio de mora es muy sensible a la temperatura.

38 *Aumentando el rigor*

En algunos círculos les gusta ser rigurosos, lo cual significa que los supuestos para el análisis de varianza deben probarse. Hagamos, pues, una lista de los supuestos que hemos utilizado para desarrollar nuestro análisis de varianza:

1. Todo lo que afecta la variable de respuesta, que debe ser cuantitativa, se puede dividir en dos partes: lo que se controla en el experimento y lo que nó, la causa de ruido.
2. El efecto del ruido debe producir una variable aleatoria que se debe leer a lo largo de la lista de datos.
3. El efecto del ruido sobre la variable respuesta viene distribuido normalmente y tiene media cero y una varianza que no depende del tratamiento. Por tanto, los datos sobre cada tratamiento vienen al azar y tienen una distribución normal.
4. El ruido y el experimento no producen efectos conjuntos: la covarianza entre el efecto del ruido y el efecto del experimento es cero. Por lo tanto, las varianzas de todas las columnas son iguales.

Estos supuestos pueden ser violados. Por ejemplo:

1. Si queremos investigar los abusos del poder, deberemos probar que el mundo se puede dividir en dos partes: los que abusan y los que son abusados. En general, es difícil hacer eso porque los que abusan tienen tentáculos por todo lado y no es posible aíslarlos del resto del mundo. Es mucho más fácil hacer dicho estudio cuando ha pasado algún tiempo después de que el poder ha pasado a líderes con ideologías contrarias a los que estaban antes.
2. Hacer encuestas sobre los temas de actualidad permite que los sesgos creados por los medios de comunicación sean reproducidos masivamente y la aleatoriedd se pierda. La madurez de los medios se ve reflejada en los debates que armen, por lo cual los periodístas se arriesgan a la muerte.
3. Para que el efecto del ruido tenga una distribución normal se requiere que haya simetría en la distribución y una desviación por defecto sea tan probable como una por exceso. Pero en un estudio de damnificados hecho por el Gobierno que promete ayudas, lo más probable es que todo el mundo quiera aparecer como muy necesitado.
4. Se estudia el comportamiento de las mamás en su forma de juzgar un jardín infantil. Como regla general, las mamás con poca experiencia tendrán pocos elementos de juicio y produciran poca variabilidad en las respuestas, pero la mamás que ya tienen varios hijos pueden matizar sus respuesta de muchas maneras que un investigador no entrenado puede pasar por alto. En este caso, el número de hijs aumentará la varianza.

Pretender mucho rigor en un análisis de varianza queda fuera de lugar: debido a que un test anova es terriblemente eficiente para rechazar la hipótesis nula de igualdad de varianzas y demostrar así que el experimento funcionó, las anovas se distinguen porque funcionan con experimentos pequeños. Pero cuando uno tiene pocos datos, uno no puede rechazar otras hipótesis nulas, por ejemplo, que no tengamos

normalidad o que las varianzas no sean iguales. Por eso, es usual no detectar ninguna violación a los supuestos por escases de evidencias y quedarse tranquilos. Pero no haber podido rechazar la hipótesis nulas es muy distinto de decir que los supuestos han sido probados. Este malentendido puede ser grave pues si uno va a aplicar los resultados, allí no tendrá unos pocos datos sino miriadas y las consecuencias quizá sean menores quizá nó.

Para evitarse tantos problemas, mucha gente prefiere definitivamente la estadística no paramétrica, que es mucho más costosa en datos pero más segura en sus decisiones. Y sin embargo, es preciso reconocer que la ciencia que conocemos muy pocas veces se ha basado en el rigor para progresar. Para verlo, es suficiente pensar en la fisiología que describe un organismo vivo compuesto por órganos que dependen extremadamente unos de otros para su funcionamiento: ¿Cómo ha sido posible asignarles una función cuando se pueden separar sólo por abstracción?

39 *Aumentando la eficiencia*

Pensando en el ejemplo de la mora, es impresionate cuán eficiente es el análisis de varianza para comparación de medias. La razón de su eficiencia se haya en que para estimar la varianza causada por el ruido usa todos los datos al mismo tiempo.

De igual modo, existe un test que compara un grupo de varianzas al mismo tiempo par adistribuciones normales. Se llama el test de Bartlett. Si además, las columnas tienen igual número de elementos, se pueden usar los tests de Hartley y de Cochran. Estos test permiten demostrar rigursoamente la homogeneidad o igualdad de varianzas que es un prerequisito para hacer las anovas.

¿Qué hacer cuando hay una violación a alguno de los postulados? A veces alguien se permite decir que hay unas violaciones que no son tan graves como otras. Por ejemplo, una anova generalzia un test t (para dos tratamientos, un test t y una anova coinciden) y el test t se basa en el teorema del límite central, el cual es aproximadamente válido para muchos tipos de distribuciones y no necesariamente la normal. Por ello, la violación de la normalidad no se castiga tan duro.

Dado que los test de comparación de varianzas son muy sensibles a la normalidad, es usual que no se tolere tal violación. El remedio es usar análisis no paramétricos que son válidos para todas las distribuciones pero tienen el costo de ser muy ineficientes: se necesitan muchos datos para poder rechazar las hipótesis nulas y así probar que el experimeto funcionó.

En la vida real hay muchos compliques. Por ejemplo, supongamos que una niña quiere hacer el experimento para su tesis en el lote de su casa, del cual ha logrado negociar con su mamá y su hermano menor que le dejen para ella un espacio de $6 \times 3m^2$. En ese terreno, ella piensa hacer unas eras para sembar zanahoria a diversos tratamientos. A ella le parece que podría hacer 18 eras. Pero no sabe si hacer un experimento con 3 tratamientos y 6 eras por tratamiento, o si sería mejor otro con 6 tratamientos y 3 eras por tratamiento, o quizá si sería mejor hacer 2 tratamientos con 9 eras cada uno, o tal vez 18 tratamientos, uno por era. ¿Usted qué haría?

Este tipo de preguntas generan un arte delicado que se llama diseño de experimentos. Pero existe la contraparte: dado un experimento hecho en un esquema fuera de serie, ¿Qué podemos hacer para estudiarlo? Podemos hacer simulaciones, que consisten en la construcción de un mundo virtual con exactamente el mismo esquema que el real y correrlo todas las veces que queramos para sacar conclusiones estadísticas.

40 *Comparación de pares de medias*

Una vez que uno sabe que hay por lo menos un par de medias que no son iguales, es posible que a uno le interese saber exactamente cómo es la relación entre medias. La solución inmediata es correr un test t para comparar medias cuando las varianzas son iguales (lo cual resulta de que los efectos del ruido y del experimento tienen covarianza cero). Pero, atención: en vez de correr un test t común y corriente y que tiene en cuenta las varianzas de las columnas en comparación únicamente, podemos recordar que los datos pesan más juntos: en la anova unifactorial hay sólo una varianza y es la creada por el ruido y tenemos derecho a estimarla como más nos convenga. Eso lo logramos cuando involucramos todos los datos de la tabla, de todas las columnas, para obtener

$$s_F^2 = SSW/(n-g).$$

En nuestro caso de la mora, tenemos:

$SSW = 3{,}33$
$s_F^2 = SSE/(n-g) = 3{,}33/6 = 0{,}55$
$s_F = \sqrt{0{,}55} = 0{,}74$

con la cual podemos aplicar una test t con varianzas iguales para comparar la media poblacional de la columna 1 con la columna 2. El test lo podemos leer sobre el intervalo de confianza para la diferencia entre medias: si el cero pertenece a dicho intervalo, no es juicioso reclamar que hay una diferencia entre medias no nula, pero si el cero está por fuera del intervalo, sí lo es. El intervalo de confianza está dado por la fórmula siguiente (ver Vol 1):

$$(\bar{X_m} - \bar{Y_n}) - t_{\alpha/2}\widehat{\sigma_J}\sqrt{\tfrac{1}{m}+\tfrac{1}{n}} < \mu_X - \mu_Y < (\bar{X_m} - \bar{Y_n}) + t_{\alpha/2}\widehat{\sigma_J}\sqrt{\tfrac{1}{m}+\tfrac{1}{n}}$$

en donde podemos reemplazar $\widehat{\sigma_J}$ por s_F. Para una significancia del 0.05 y gl $= n - g = 9 - 3$ la t crítica con dos colas es 2.44 y obtenemos el intervalo de confianza:

$$(4 - 1{,}33) - (2{,}44)(0{,}74)\sqrt{\tfrac{1}{3}+\tfrac{1}{3}} < \mu_X - \mu_Y < (4 - 1{,}33) + (2{,}44)(0{,}74)\sqrt{\tfrac{1}{3}+\tfrac{1}{3}}$$
$$2{,}77 - 1{,}47 < \mu_X - \mu_Y < 2{,}77 + 1{,}47$$
$$1{,}3 < \mu_X - \mu_Y < 4{,}24$$

Como cero no pertenece a este intervalo, concluimos que hay una diferencia estadísticamente significativa, repetible con el 95 % de confianza, entre la producción promedio de mora a 9 grados y la que da a 18.

Este test se debe a **Fisher** y se llama el *test LSD* (least significant difference). Pero modernamente se han elaborado otras opciones. Una de ellas es el **test de Tukey** y es la que adoptaremos más abajo para correr sobre *R*.

41 *Los dos primeros teoremas fundamentales del método científico*

Habíamos dejado para después el problema de detectar efectos que por su debilidad son fácilmente tapados por el ruido. Pero antes de seguir, demos un ejemplo en el cual tal efecto sea importante para la ciencia. El caso es el de la radiación de fondo, que es una radiación de ondas de radio que vienen del cielo y que atraviezan la tierra así que se pueden registrar viniendo tanto del cielo como del suelo. Esa radiación debe tener características precisas para que apoye la teoría cosmológica del Big Bang. Su estudio completo exige mediciones muy finas, pero hay un problema: los equipos electrónicos que se usan para su registro producen un ruido interno que no puede apagarse y que puede ser más fuerte que las señales que deben captarse y que no se remedian si se hace el estudio desde satélites. ¿Qué se puede hacer? Lo único que se puede hacer es tomar en serio la definición de ruido (con media cero): el azar en promedio no es ni fu ni fa y por tanto cuando se acumulen muchos datos, sus efectos se autoaniquilarán.

Esta ley de autoaniquilación del azar es un teorema de la teoría de probabilidad conocido como la ley de los grandes números y cuya versión parafraseada dice:

Ley de los grandes números: *sea X una v.a. con media μ y una desvación σ, que denota la magnitud de su carácter aleatorio. En general, al tomar una muestra al azar de tamaño n y promediar para obtener $\bar{X}_n$, uno obtendrá un número diferente de μ cada vez que lo haga. Pero cuando el número de datos aumenta hacia el infinito, $\bar{X}_n$ tenderá a parecerse a μ. Sea $d = \|\bar{X}_n - \mu\|$. Cuando $n \to \infty$ la probabilidad de que d tienda a cero tiende a uno. Es decir, cuando uno le permite al azar autoexpresarse, éste se autoaniquila.*

En el siguiente link hay hermosas simulaciones que ilustran este teorema, junto con explicaciones y comentarios:

http://en.wikipedia.org/wiki/Law_of_large_numbers

De lo dicho se deduce inmediatamente que

Primer teorema del método científico: Entre más datos tenga un experimento que haga referencia a valores esperados, sea de la media o de la varianza, más autoridad tiene y más confianza genera, pues la incertidumbre debida al azar tiende a autoaniquilarse.

Este teorema tiene sus implicaciones en todo lado. Para ilustrarlo, examinemos el intervalo de confianza para la diferencia de medias cuando las varianzas son iguales:

$$(\bar{X_m} - \bar{Y_n}) - t_{\alpha/2}\widehat{\sigma_J}\sqrt{\tfrac{1}{m} + \tfrac{1}{n}} < \mu_X - \mu_Y < (\bar{X_m} - \bar{Y_n}) + t_{\alpha/2}\widehat{\sigma_J}\sqrt{\tfrac{1}{m} + \tfrac{1}{n}}$$

En este expresión, hay 2 fuentes de incertidumbre que se pueden controlar por medio del número de datos. La primera es el término $\sqrt{\frac{1}{m} + \frac{1}{n}}$. Cuando el número de datos aumenta en ambas muestras, este término decrece. Desafortunadamente decrece lentamente, pues la raíz crece despacio y eso implica que la precisión científica cuesta caro y entre más alta, más costosa. Pero de todos modos, entre más datos, la incertidumbre disminuye, los intervalos de confianza se contraen y hay más confianza en la toma de decisiones. Por otra parte, cuando el número de datos aumenta, los grados de libertad de la t también aumentan proporcionalmente y cuando eso pasa, la t, que es una campana parecida a la z pero más explayada, se va cerrando contra la z y así la incertidumbre en la evaluación de la diferencia de medias también disminuye. Tenemos entonces que:

Entre más datos tenga un experimento, los intervalos de confianza de la media son más pequeños.

En realidad, el ejemplo de la t no prueba que éso sea cierto en todos los casos. Pero es natural suponer que si alguien se inventa un determinado test, por fuerza de su propósito científico debe hacerlo de tal manera que el test aumente su **eficiencia** con el número de datos, es decir, el número de datos para rechazar la hipótesis nula con una determinada confianza debe disminuir.

Una implicación práctica del primer teorema es la siguiente:

Segundo teorema del método científico: *los datos pesan más juntos que separados.*

que dice que los grandes proyectos experimentales que se analizan como un todo son más eficaces que varios proyectos pequeños que luego se discuten juntos. Por ésa razón, las anovas que comparan todas las medias al tiempo son tan eficientes, muchísimo más que comparaciones por pares y que se hace con una t. El mismo principio se aplica para comparar varianzas, pero lo veremos después.

42 *Taller en R*

Para hacer la anova de la mora en R hay que ponerle los datos como fueron registrados en el campo. La idea es que uno tiene viveros con termostatos que gradúan y conservan la temperatura y que para evitar sesgos, por ejemplo de posicionamiento, los tratamientos se asignan al azar a los viveros. Esto es parte del protocolo para garantizar que los datos vengan al azar, así que especifiquémoslo mejor:

Diseño unifactorial para estudiar la influencia de la temperatura sobre la producción promedio de mora:

Tenemos un terreno en el cual se pueden hacer 9 viveros cuadrados cada una de un metro de lado. Se numeran los viveros por orden natural según el terreno. Queremos registrar la producción a tres temperaturas diferentes de 9, 18 y 25 grados, lo cual nos da tres tratamientos y podemos repetir cada tratamiento 3 veces. Entonces se toma un vivero al azar y se le asigna el primer tratamiento. De los viveros que quedan se toma otro al azar y se le asigna el segundo tratamiento. Después se asigna un tercer tratamiento. Luego se repite el procedimiento otras dos veces.

Supongamos pues que la lista de datos de las producciones fue

Tratamiento	Prod
nueve	1
docho	3
nueve	2
vcinco	1
nueve	1
vcinco	1
docho	4
vcinco	2
docho	5

donde hemos usado las convenciones *docho = dieciocho*, *vcinco = veinticinco*.

El programa en *R* para la anova podría ser:

```
#==============================
#ANOVA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Los tratamientos a los 9 viveros
Trat<-c( "nueve", "docho", "nueve","vcinco", "nueve", "vcinco", "docho", "vcinco","docho")
#Producción de los viveros
Prod<-c( 1,3,2,1,1,1,4,2,5)
#Se hace una tabla a partir de dos vectores
Mora<-data.frame(Trat,Prod)
Mora
#Tabla bidimensional de frecuencias absolutas
z<-table(Mora$Prod,Mora$Trat)
z
#analysis of variance
w<-aov(Mora$Prod ~ Mora$Trat, projections = TRUE )
summary(w)
#Comparación de medias por pares según Tukey
#Se ordenan las diferencias ntre medias antes del test
tt <- TukeyHSD(w, ordered = TRUE, conf.level = 0.99)
plot(tt)
```

Este programa arroja un output casi idéntico al de *Gnumeric* y al de *Excel*. Añade la convención de indicar la significancia de los datos por medio de estrellas: dos estrellas, **, dicen que la significancia es menor que 0.01, tres, que es menor que 0.001.

Es posible que alguien prefiera usar una codificación numérica para los tratamientos. En tal caso, es posible que se le ocurra usar un programa del siguiente estilo:

```
#==============================
#PROGRAMA PERNICIOSO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Los tratamientos a los 9 viveros
Trat<-c( 9, 18, 9,25, 9, 25, 18, 25,18)
```

```
#Producción de los viveros
Prod<-c( 1,3,2,1,1,1,4,2,5)
#Se hace una tabla a partir de dos vectores
Mora<-data.frame(Trat,Prod)
Mora
#Tabla bidimensional de frecuencias absolutas
z<-table(Mora$Prod,Mora$Trat)
z
#analysis of variance
w<-aov(Mora$Prod ~ Mora$Trat, projections = TRUE )
summary(w)
```

Sucede que R toma en serio los números y produce lo que no debe. Para que proceda como queremos, hay que decirle que los números son sólo símbolos que indican una clasificación categórica, como blanco y rojo, y no numérica:

```
#==============================
#ANOVA: NUMEROS QUE CODIFICAN FACTORES
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(1,1), pch=16)
#Los tratamientos a los 9 viveros
trat<-c( 9, 18, 9,25, 9, 25, 18, 25,18)
#Tome los números como categorías o factores
Trat <- as.factor(trat)
#Producción de los viveros
Prod<-c( 1,3,2,1,1,1,4,2,5)
#Se hace una tabla a partir de dos vectores
Mora<-data.frame(Trat,Prod)
Mora
#Tabla bidimensional de frecuencias absolutas
z<-table(Mora$Prod,Mora$Trat)
z
#analysis of variance
w<-aov(Mora$Prod ~ Mora$Trat, projections = TRUE )
summary(w)
#Comparación de medias por pares según Tukey
#Se ordenan las diferencias ntre medias antes del test
tt <- TukeyHSD(w, ordered = TRUE, conf.level = 0.99)
plot(tt)
```

En un experimento real o en una encuesta, lo más seguro es que uno haya tecleado los datos en *Excel* o en *Gnumeric*. Para hacerlos llegar a R se regraban en un archivo tipo text, el cual es entendible por R. Para fijar ideas, consideremos la tabla que estudia los sueldos entre unos profesionales, reportada en el capítulo anterior y que seguramente ya fue grabada. Para hacerla llegar a R por medio del *GUI*, uno va a *Data* + *Import data* + *from text file* y allí localiza el archivo correspondiente. Al abrirlo se abre un diálogo y hay que darle un nombre, el nombre que llevará la tabla en R, y uno deja el resto sin tocar y abre el archivo. Puede luego mostrarlo o hacerle diversas correcciones. Y uno notará que esta tabla activa la casilla de las anovas (que se busca en el área de estadística y de medias). Y puede uno entonces hacer diversas comparaciones de medias.

Si uno prefiere usar la consola de R, uno podría teclear alguna adaptación del programa siguiente, que se ejecuta sobre la tabla *tsueldos*, la misma del capítulo dos. El siguiente programa para Linux debe adecuarse teniendo encuenta el camino al archivo que contiene los datos de *tsueldos*:

```
#==============================
```

```
#PROGRAMA PARA  LA ANOVA DE UNA VIA EN LINUX
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
tsueldos <-  read.table(file("AJose/RProjectU1/RWorkU1/tsueldos.txt"), header = T)
tsueldos
#Tabla bidimensional de frecuencias absolutas  sueldo  vs prof
z<-table(tsueldos$Sueldo,tsueldos$Prof)
z
#analysis of variance
w <- aov(tsueldos$Sueldo ~ tsueldos$Prof, projections = TRUE)
summary(w)
#Comparación de medias por pares según Tukey
#Se ordenan las diferencias entre medias antes del test
tt <- TukeyHSD(w, ordered = TRUE, conf.level = 0.99)
plot(tt)
```

Para Windows, uno puede adaptar una adaptación de la siguiente versión:

```
#==============================
#PROGRAMA PARA  LA ANOVA DE UNA VIA SOBRE WINDOWS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
tsueldos <-read.table(file("C:/AJose/RProjectU1/RWorkU1/tsueldos.txt"), header = T)
#Publicamos en consola el archivo
tsueldos
#
#Tabla bidimensional de frecuencias absolutas  sueldo  vs prof
z<-table(tsueldos$Sueldo,tsueldos$Prof)
z
#analysis of variance
w <- aov(tsueldos$Sueldo ~ tsueldos$Prof, projections = TRUE)
summary(w)
w <- aov(tsueldos$Sueldo ~ tsueldos$Prom, projections = TRUE)
summary(w)
tt <- TukeyHSD(w)
plot(tt)
```

Si uno desea comparar la salida de R con la de *Gnumeric*, es necesario darle a *Gnumeric* la siguiente tabla que reporta los sueldos individuales por profesión:

Civil	Eléct	Ind	Sist
2	1	3	2
3	1	2	2
3			2

Uno puede modificar el programa anterior para comparar el sueldo promedio por promoción. Sin embargo, si uno desea hacer el mismo análisis con ayuda de la GUI, uno verá que no puede. La razón es que R considera que el año de la promoción no es un factor sino una variable respuesta por ser numérica. Por tanto, hay que indicarle que haga el cambio de perspectiva. Para ello, uno se asegura que el conjunto de datos activado es el que uno desea. Luego uno va a *Data* y luego a *manage variables in active data set* y allí elige *convert numeric variables to factor*. Después se despliega un diálogo en el cual uno tiene la opción de cambiar el nombre numérico del año por un nombre con letras, por ejemplo, a *2001* le damos el nombre de *uno*. O uno tiene la opción de dejar el mismo 2001, pero como palabra no como número. y así con todo lo demás. Después ya podrá comparar los sueldos de acuerdo a la promoción.

Como se ve, codificar una variable categórica por medio de números es algo que crea muchos problemas y que no siempre abre opciones. Por ello, lo mejor podría ser cerrar dicha posibilidad: en vez de codificar el año 2001 comoo 2001, uno podría codificarlo como 2001A y así con los demás años y así se ahorra inconvenientes. O quizá a uno le convenga tener las dos opciones, conservar el año como número (para hacer una regresión que alega que entre más temprano fue la promoción, más antiguedad y mayor sueldo) y también como palabra, como nivel de un factor, para poder estudiar la incidencia de la promoción sobre el sueldo. Todos estos cambios pueden salvarse: en *Data + active data set* uno encuentra la opción para salvar.

Si hay ambivalencia entre ser número y ser palabra (ser factor o variable cuantitativa), es necesario quitarla, pues de lo contrario uno obtendrá resultados que corresponden a la interpretación como número de todo lo que esté codificado en números sólamente. Cuando *R* ve números considera que se pueden ordenar en cambio no hace lo mismo con las palabras. Como consecuencia, esta ambivalencia cambia los grados de libertad, los estadígrafos y por tanto las decisiones.

43 ***Múltiple comparación de varianzas***

Uno de los supuestos para nuestro ataque a las anovas es que las varianzas son homogeneas, es decir que no varían con los tratamientos. Una forma de revisar la validez de dicho supuesto para nuestros datos es hacer una comparación por pares usando una F. Pero como los datos pesan más juntos que separados, es más eficiente hacer una sóla comparación múltiple. En cuanto a varianzas, el **test de Bartlett** para datos cuantitativos es muy popular. Corre únicamente sobre tablas naturales. Se puede correr desde la *GUI* y también desde la consola haciendo una adaptación al siguiente comando:

```
bartlett.test(tsueldos$Sueldo ~ tsueldos$Prom)
```

Como tenemos muy pocos datos, este comando se frustra. Podemos correrlo sobre un conjunto de datos más grande que tomamos de la librería que *R* tiene. Para listar los archivos de *data* datos a disposición, usamos el comando

```
data()
```

Para saber la información sobre un archivo de datos, por ejemplo *CO2*, tecleamos el nombre del archivo y del paquete en el cual está:

```
data(CO2, package = "datasets")
help("CO2")
```

Para listar los datos, escribimos el nombre del archivo:

```
CO2
```

Para correr el test de Bartlett escogemos la variable respuesta, que siempre se lista de primera y en segundo lugar el criterio de comparación, que ha de ser un factor.

```
bartlett.test(uptake ~ Treatment, data = CO2)
```

Como el p-value nos da 0,4, no hay razón para dudar de la homogeneidad de las varianzas. Lo haríamos si el p-value fuese menor que 0,05.

4.2. Anovas con bloqueo

Se trata de preparar un laboratorio sobre las mitocondrias, las fábricas de energía de la célula. Queremos medir como variable de respuesta la producción de ATP, la molécula cargada de energía en un trifosfato. Queremos enseñar que el pH de la solución en la que se guarden las mitocondrías es muy importante. Pero como usamos mitocondrias de hígado de ratón, un mamífero de sangre caliente, nosotros podemos intuír que la temperatura es muy importante y que es mejor que la controlemos.

La idea es hacer un experimento tipo *anova unifactorial con bloqueo* de la forma 3×3. Con éso queremos decir que tenemos, por ejemplo, 3 valores de pH y 3 de temperatura. La forma de hacerlo es preparar 3 soluciones a los pH determinados. Con cada solución se llenan con una medida exacta 3 pequeños recipientes a los cuales se les echa una determinada cantidad de mitocondrias. Luego se toma un recipiente al azar y se le asigna la primera temperatura. Luego a un segundo recipiente tomado al azar entre los que quedan se asigna la segunda temperatura, y luego el tercero. Y se repite lo mismo otras dos veces. Las asignaciones se hacen al alzar porque por más cuidadoso que uno sea, el pH de los recipientes puede variar así sea porque uno respire más cerca de uno que del otro. Luego se toman todos

los recipientes asignados a la misma temperatura y se meten en un horno con control de temperatura. Y lo mismo con los otros dos grupos.

Vemos que una **anova con bloqueo** estudia el efecto promedio de diversos tratamientos o niveles de una variable de estudio pero controlando una variable presumiblemente importante tomada de entre las variables de fondo que responden por el ruido. En la literatura a este diseño se le llama diseño por bloques (block design, randomized block design). Me parece posible que este nombre tan raro se haya heradado de la bioquímica y afines donde puede quedar cómodo poner en un sólo y mismo bloque todos los especímines que tengan un mismo valor de la variable que se quiere controlar o bloquear.

Bloquear variables es algo muy difícil. Por ejemplo, uno puede graduar y estabilizar la temperatura del horno para luego meter allí todos los vasos de Petri que estén asignados a la temperatura elegida. Pensará uno que ya quedó bloqueada la variable temperatura al valor que marca el horno. En realidad no hay tal: a no ser que sean hornos de precio muy elevado, ellos tienen varios microambientes que pueden hacerse sentir en el plazo de unas cuantas horas. Y que ni se diga de lo que puede pasar en días o semanas.

Para calcular una anova con bloqueo, comencemos haciendo el truco del cero:

$$y_{ij} = y_{ij} + 0 = y_{ij} + 2(\bar{y}_{\bullet\bullet} - \bar{y}_{\bullet\bullet}) + (\bar{y}_{i\bullet} - \bar{y}_{i\bullet}) + (\bar{y}_{\bullet j} - \bar{y}_{\bullet j})$$
$$= \bar{y}_{\bullet\bullet} + (\bar{y}_{i\bullet} - \bar{y}_{\bullet\bullet}) + (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) + (y_{ij} - \bar{y}_{\bullet j} - \bar{y}_{i\bullet} + \bar{y}_{\bullet\bullet})$$

Esos términos se interpretan así: $\bar{y}_{\bullet\bullet}$ es un estimador insesgado de la media global, μ. El término $(\bar{y}_{i\bullet} - \bar{y}_{\bullet\bullet})$ estima una desviación promedio con respecto a la media global explicada por el tratamiento i de la variable de bloqueo, cuyo equivalente poblacional es β_i. Por otro lado, $(\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet})$ es una desviación promedio con respecto a la media global explicada por el tratamiento j de la variable de respuesta, que estima α_j. Por último $(y_{ij} - \bar{y}_{\bullet j} - \bar{y}_{i\bullet} + \bar{y}_{\bullet\bullet})$ es una fluctuación debida a todo lo que no se controla, al azar que se idealiza en ϵ_{ij}. Tenemos que estamos asumiendo el siguiente modelo:

$$y_{ij} = \mu + \beta_i + \alpha_j + \epsilon_{ij}$$

Hay g tratamientos y b bloqueos. El número total de datos es $n = bg$. Hay una única fuente de incertidumbre y está en ϵ_{ij} que se supone normalmente distribuida con media cero y varianza σ^2. La normalidad nos permitirá aplicar estadígrafos que relacionen varianzas y que se distribuyan como una F.

La hipótesis nula dice: ni el factor principal, ni el factor de bloqueo tienen algún efecto promedio sobre la variable respuesta y si los estimadores de dichos parámetros no dan cero, éso se debe a que el azar puede generar por pura casualidad efectos no zero pero no repetibles. Esta hipótesis se falsifica con dos colas y puede ser debido a que el factor principal si incide o a que el factor de bloqueo lo haga.

Especificando todo completamente, tenemos que el promedio global, μ, se estima por $\bar{y}_{\bullet\bullet}$:

$$\bar{y}_{\bullet\bullet} = \frac{1}{bg} \sum_{i=1}^{b} \sum_{j=1}^{g} y_{ij}$$

donde se promedia sobre todos los datos, $n = bg$. La primera suma corre sobre los b bloqueos y la segunda sobre todos los g tratamientos. Para estimar las betas tenemos en cuenta que son desvaciones promedio con respecto a la media global y que se estiman por:

$$\hat{\beta}_i = \bar{y}_{i\bullet} - \bar{y}_{\bullet\bullet}$$

donde

$$\bar{y}_{i\bullet} == \frac{1}{g} \sum_{j=1}^{g} y_{ij}$$

Similarmente:

$\hat{\alpha}_j = \bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}$
$\bar{y}_{\bullet j} = \frac{1}{b}\sum_{i=1}^{b} y_{ij}$

El significado de $\bullet$ se refiere a la matriz de datos y depende de su posición: si está en el primer subíndice dice que éste ha dejado de ser una variable porque se ha promediado sobre él. Si está en el segundo subíndice dice que el segundo subíndice ha dejado de ser una variable porque, por ejemplo, se ha promediado sobre el. Y si está en ambos lugares es porque se ha promediado tanto sobre el filas, el primer subíndice, como sobre columnas, el segundo.

Ahora especifiquemos el azar o el ruido causado por los factores no controlados: existe una única fuente de incertidumbre y está en ϵ que se estima por

$\hat{\epsilon}_{ij} = y_{ij} - \bar{y}_{\bullet j} - \bar{y}_{i\bullet} + \bar{y}_{\bullet\bullet}$

la suma de las desviaciones da cero:

$\sum \alpha_i = \sum \beta_j = 0$

lo cual se puede suponer siempre y combina bien con los estimdores.

Ahora, pondremos atención a la variación de los datos. Como definición de **variación total observada** tenemos un múltiplo de la varianza total observada:

variación total observada $= SST = \sum_{i=1}^{b}\sum_{j=1}^{g}(y_{ij} - \bar{y}_{\bullet\bullet})^2$

donde seguimos adoptado la costumbre de llamar a esta variación como suma de cuadrados totales que se nota SST (del inglés *total sum of squares*). Ahora reeplazamos y_{ij} por nuestra expresión sacada del truco del cero y obtenemos:

$SST = \sum_{i=1}^{b}\sum_{j=1}^{g}[\bar{y}_{\bullet\bullet} + (\bar{y}_{i\bullet} - \bar{y}_{\bullet\bullet}) + (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) + (y_{ij} - \bar{y}_{\bullet j} - \bar{y}_{i\bullet} + \bar{y}_{\bullet\bullet}) - \bar{y}_{\bullet\bullet}]^2$
$= \sum_{i=1}^{b}\sum_{j=1}^{g}[(\bar{y}_{i\bullet} - \bar{y}_{\bullet\bullet}) + (\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet}) + (y_{ij} - \bar{y}_{\bullet j} - \bar{y}_{i\bullet} + \bar{y}_{\bullet\bullet})]^2$

Se aplica la expansión de un trinomio al cuadrado $(a+b+c)^2 = a^2 + b^2 + c^2 + 2ab + 2ac + bc$, se identifican las covarianzas que se postulan iguales a cero, lo que quiere decir que el investigador conoce demasiado bien su terreno y puede asumir que el factor principal, el factor de bloqueo y el conjunto de todo lo demás son independientes. Y obtenemos que la variación total SST se divide en tres términos: una contribucón del factor principal y debida a la variación entre (Among) medias, SSA, una contribución del factor de bloqueo SSB, y una contribución debida al ruido, o variación error SSE:

$SST = SSA + SSB + SSE$

donde especificamos cada término de dos maneras, la oficial y la que se usa para hacer cálculos:

$SST = \sum_{i=1}^{b}\sum_{j=1}^{g}(y_{ij} - \bar{y}_{\bullet\bullet})^2 = \sum_{i=1}^{b}\sum_{j=1}^{g}(y_{ij})^2 - gb(\bar{y}_{\bullet\bullet})^2$

$SSA = \sum_{j=1}^{g}\sum_{i=1}^{n_j}(\bar{y}_{\bullet j} - \bar{y}_{\bullet\bullet})^2 = b\sum_{j=1}^{g}(\bar{y}_{\bullet j})^2 - gb(\bar{y}_{\bullet\bullet})^2$

$SSB = \sum_{j=1}^{g}\sum_{i=1}^{n_j}(\bar{y}_{i\bullet} - \bar{y}_{\bullet\bullet})^2 = g\sum_{i=1}^{b}(\bar{y}_{i\bullet})^2 - gb(\bar{y}_{\bullet\bullet})^2$

SSE = SST -SSA -SSB.

Ahora tranformamos SST, SSA y SSE en varianzas que representen estimadores insesgados de la única varianza poblacional en juego: la varianza del azar, la cual es la única quee existe bajo la hipótesis nula. Por éso se toma $SSA/(g-1)$, $SSS/(b-1)$ y $SSE/[(b-1)(g-1)]$. Estas expresiones son estimadores

de la misma σ^2, la varianza debida al ruido, al azar. La razón es que SSE mide la variación dentro de cada grupo que es efecto natural de la varianza debida al azar. Por tanto, todo lo que se requiere para hallar un estimador insesgado de σ^2 es dividir dicho término por un número apropiado, que resulta ser $(b-1)(g-1)$. Por otro lado, SSA mide la variación debida a los cambios entre tratamientos, los niveles del factor principal. Pero bajo la H_o, eso no influye para nada, y por tanto SSA también tiene su origen en el azar. Como consecuencia, podemos dividir por un número apropiado para que nos de un estimador insesgado de la varianza σ^2. El número apropiado resulta ser $g-1$. Y lo mismo se hace con SSB.

Consideremos el quebrado:

$$R_{A,exp} = \frac{SSA/(g-1)}{SSW/[(g-1)(b-1)]}$$

que mide la relación observada entre los efectos del experimento y los del ruido. Pero bajo la hipótesis nula, ambos términos del quebrado son estimadores insesgados de la misma varianza, entonces podemos decir que la H_o predice que el valor esperado del quebrado $R_{A,exp}$ debe ser 1. Por tanto, el quebrado que mide la relación entre lo que se ve, $R_{A,exp}$ y lo que se cree, $R = 1$, es igual a

$$F_{A,exp} = \frac{R_{exp}}{1} = R_{A,exp}$$

Puesto que venimos de poblaciones normales, la $F_{A,exp}$ se distribuye como una F, pero hay que determinar los grados de libertad. Quisiéramos, por supuesto, decir que los grados de libertad de la F son $g-1$ en el numerador y $(b-1)(g-1)$ en el denominador. Pero para poder decir eso, tenemos que demostrar que no hay redundancia de información de los dos grupos de datos, los que producen el numerador de los que producen el denominador. Pero eso lo garantizamos al decir que la covarianza entre los factores del experimento y el ruido son independiente, con covarianza cero. Si no tuviésemos dicha covarianza igual a cero, tendríamos que hacer cuentas engorrosas, quizá imposibles, para determinar cuántos grados de libertad se pierden.

Con todos los supuestos dichos, la $F_{A,exp}$ se distribuye como una F con $g-1$ en el numerador y $(b-1)(g-1)$ en el denominador.

También podemos cuantificar el efecto del bloqueo por medio de:

$$R_{B,exp} = \frac{SSB/(b-1)}{SSW/[(g-1)(b-1)]}$$

que mide la relación observada entre los efectos del bloqueo y los del ruido. Pero bajo la hipótesis nula, ambos términos del quebrado son estimadores insesgados de la misma varianza, la del ruido, pues no hay más fuentes de variabilidad. Entonces podemos decir que la H_o predice que el valor esperado del quebrado $R_{B,exp}$ debe ser 1. Por tanto, el quebrado que mide la relación entre lo que se ve, $R_{B,exp}$ y lo que se cree, $R = 1$, es igual a

$$F_{B,exp} = \frac{R_{B,exp}}{1} = R_{B,exp}$$

La $F_{B,exp}$ se distribuye como una F con $b-1$ en el numerador y $(b-1)(g-1)$ en el denominador. La H_o dice que los factores controlados no inciden sobre el valor promedio de la variable respuesta y por tanto los valor esperados de las medias bajo los diversos tratamientos coinciden. Cuando la H_o es cierta, el valor de F_{exp} es cercano a uno. La H_a dice que hay al menos un par de medias diferentes. La H_o se rechaza mediante la F_{exp} con una cola, la superior, pues en la medida en que los factores influyen, la SSA se hace más grande y como está en el numerador de la F_{exp}, ésta crece y se aleja de uno.

44 *Cálculos*

Procesemos el experimento de la mitocondria:

Mitochondria			
ATP production			
	pH 4	pH 6	pH 8
30^o	5	8	6
36^o	7	10	8
42^o	4	7	3

Uno puede procesar estos datos en *Gnumeric:* uno elige el menu *Tools*, luego *statistics*, luego *anova* y después escoge *Two factors + one row per sample.*

Uno puede implementar el algoritmo para estudiar una anova con bloqueo siguiendo los siguientes pasos:

Tenemos $g = 3$ tratamientos y $b = 3$ niveles de bloqueo, $n = bg = 9$.
GS_i= Great Square under treatment $i = (\sum_{j=1}^{n_i} X_{ij})^2$
$GS_1 = (5 + 7 + 4)^2 = 16^2 = 256$
$GS_2 = (8 + 10 + 7)^2 = 25^2 = 625$
$GS_3 = (6 + 8 + 3)^2 = 17^2 = 289$
$T = \sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{ij} = 5 + 7 + 4 + 8 + 10 + 7 + 6 + 8 + 3 = 58$
GS= Great Square = $(\sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{ij})^2$
$GS = T^2 = (5 + 7 + 4 + 8 + 10 + 7 + 6 + 8 + 3)^2 = 58^2 = 3364$

Among Columns SS = $\sum_{i=1}^{k} \frac{GS_i}{l} - \frac{GS}{N}$ =(256+ 625+ 289)/3 - 3364/9= 390 -373.7 = 16.3
Among columns DF= Among columns degrees of freedom= k-1= 3-1=2.
Among columns Mean square= $\frac{AmongColumnsSS}{AmongcolumnsDF} = 16{,}3/2 = 8{,}1$
We repeat the procedure with blocks or rows or levels of blockade:
$BGS_1 = (5 + 8 + 6)^2 = 19^2 = 361$
$BGS_2 = (7 + 10 + 8)^2 = 25^2 = 625$
$BGS_3 = (4 + 7 + 3)^2 = 14^2 = 196$
$T = \sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{ij} = 5 + 7 + 4 + 8 + 10 + 7 + 6 + 8 + 3 = 58$
GS= Great Square = $(\sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{ij})^2$
$GS = T^2 = (5 + 7 + 4 + 8 + 10 + 7 + 6 + 8 + 3)^2 = 58^2 = 3364$

Among Rows SS = $\sum_{i=1}^{k} \frac{GS_i}{k} - \frac{GS}{N}$ =(361+ 625+ 196)/3 - 3364/9= 394 -373.7 = 20.3
Among rows DF= Among rows degrees of freedom= l-1= 3-1=2.
Among rows Mean square= $\frac{AmongRowsSS}{AmongrowsDF} = 20{,}3/2 = 10{,}1$
Now we calculate the square of each entry.

X_{ij}^2		
pH4	pH6	pH8
25	64	36
49	100	64
16	49	9

Total SS= Total Sum of Squares= $\sum_{i=1}^{k} \sum_{j=1}^{n_i} (X_{ij}^2) - \frac{GS}{N}$
25 + 64 +36 +49 + 100 +64+16 + 49+9
$=(25 + 64 + 36 + 49 + 100 + 64 + 16 + 49 + 9) - \frac{3364}{9} = 412 - 373{,}7 = 38{,}3$
Error SS=error sum of squares= Total SS - Among columns SS -Among row SS= 38.3-16.3-20.3= 1.7
Error DF= Total DF - Among groups DF - Among rows DF= $8 - 2 - 2 = 4$.
Error Mean square= $\frac{ErrorSS}{errorDF} = 1{,}7/4 = 0{,}43$

$F - experimental\ for\ columns = \frac{Among\ columns\ mean\ square}{Error\ Mean\ square} = \frac{8{,}1}{0{,}43} = 18{,}8$
$F - experimental\ for\ rows = \frac{Among\ rows\ mean\ square}{Error\ Mean\ square} = \frac{10{,}1}{0{,}43} = 23{,}4$

Ahora comparamos estos valores con los críticos para los correspondientes dados por *Gnumeric*. Dicho valor es 6.94 para ambos casos. Como los valores experimentales de F son más grandes que los críticos, concluimos que el pH y la temperatura son factores que influyen en la producción promedio de ATP por la mitocondria.

45 *El tercer teorema fundamental del método científico*

Si uno corre una anova unifactorial normal, sin bloqueo, sobre los datos de la mitocondria para dilucidar el efecto del pH, uno encuentra que el p-value es 0.19, mayor que 0.05, lo cual dice que con los datos a mano no tenemos razón para alegar que el pH es importante. Sin embargo, nuestro análisis de los mismos datos asumiendo bloqueo nos dieron que teníamos datos de sobra para alegar que tanto el pH como la temperatura eran factores que influían la producción promedio de ATP por la mitocondria.

Esto no es una coincidencia. Sucedió así porque en un experiento el bloqueo de un factor de fondo importantante siempre hace ver más nítido el efecto del factor en estudio:

Tercer teorema fundamental del método científico: *entre más factores se controlen en un experimento, más contundentes serán las decisiones.*

La forma de argumentar esta proposición es la siguiente: hacemos un experimento con bloqueo y calculamos el p-value de la $F_{A,exp}$, la cual es sensible a los cambios promedio de la variable respuesta entre tratamientos. El p-value depende de la $F_{A,exp}$ y de los grados de libertad. Así que estudiamos cómo cambia la $F_{A,exp}$ cuando se pasa de no control a control. Esperamos que el control aumente la $F_{A,exp}$ y que dismuya la correspondiente F-crítica. Veamos:

Para una anova con bloqueo tenemos

$$F_B = F_{A,exp} = \frac{SSA/(g-1)}{SSE_B/[(g-1)(b-1)]} = \frac{SSA}{SSE_B}\frac{1/(g-1)}{1/[(g-1)(b-1)]}$$

Para una anova sin bloqueo tenemos $n = bg$ y

$$F = F_{A,exp} = \frac{SSA/(g-1)}{SSE/(n-g)} = \frac{SSA/(g-1)}{SSE/(bg-g)} = \frac{SSA/(g-1)}{SSE/(b(g-1))} = \frac{SSA}{SSE}\frac{1/(g-1)}{1/(b(g-1))}$$

Ahora, comparemos estas dos efes, F y F_B:

Tenemos que $g-1 < g$ y por tanto $(g-1)(b-1) < g(b-1)$ puesto que $b-1$ es positivo. Pero los recíprocos se relacionan al revés:

$$\frac{1}{(g-1)(b-1)} > \frac{1}{g(b-1)}$$

Por ejemplo, $2 < 3$ pero $1/2 < 1/3$. Similarmente

$$\frac{\frac{1}{1}}{(g-1)(b-1)} < \frac{\frac{1}{1}}{g(b-1)}$$

Multipliquemos a ambos lados por $\frac{1}{g-1}$, un número positivo,

$$\frac{1/(g-1)}{(g-1)(b-1)} < \frac{1/(g-1)}{g(b-1)}$$

Por ejemplo, para $b = g = 3$ obtenemos:

$$\frac{1/2}{(2)(2)} < \frac{1/(2)}{3(2)}$$

$$\frac{1/2}{\frac{1}{4}} < \frac{1/(2)}{\frac{1}{6}}$$

$$2 < 3$$

Punto en contra, pues esto implicaría que $F_B < F$, contrario a lo que esperamos. Esto implica que bloquear de por sí no es una panacea: si un investigador está interesado en bloquear, más le vale que escoja bien la variabale o factor que va a controlar.

Pero por otro lado, en una anova normal

$SST = SSA + SSE$
$SSE = SST - SSA$

en tanto que en una anova con bloqueo

$SST = SSA + SSB + SSE_B$
$SSE_B = SST - SSA - SSB$

y como SST y SSA son los mismos en ambos casos, tenemos que SSE_B es menor, quizá mucho menor, que SSE. Por tanto,

$\frac{SSA}{SSE} < \frac{SSA}{SSE_B}$

Punto a favor: bloquear con ciencia, con conocimiento del sistema estudiado, es productivo.

Adicionalmente, consideremos el efecto de los grados de libertad: para la anova normal son $g-1$ y $n-g = bg-g = g(b-1)$, para la anova con bloqueo son $g-1$ y $(b-1)(g-1)$. Así que los grados de libertad del denominador son mayores para la anova normal que para la anova con bloqueo. Pero sucede que cuando los grados de libertad del denominador aumentan, la F crítica disminuye. Por lo tanto, la F crítica sin bloqueo sería menor que la F crítica con bloqueo. Punto en contra: esperábamos que fuese al revés.

Concluimos que bloquear no necesariamente aumenta las F_{exp}. Ni tampoco disminuye la F crítica. Sino que dependiendo de qué tan bién se elija la variable a bloquear, entonces la F_{exp} puede aumentar majestuosamente como en el ejemplo de la mitocondria.

En suma: el tercer teorema fundamental del método científico es verdadero sólo para los profesionales: personas que entienden bien y a fondo el tema de estudio y que además confían mucho en sí mismos. Como consecuencia, es arriesgado aprobar proyectos a novatos. Es sensato aprobar proyectos a grupos de trabajo con trayectoria. Lo malo es que los grupos consagrados son en general presa fácil de las ideologías dominantes: si quiere producir cosas nuevas, empléese sólo medio tiempo (creo que este consejo se debe a Crick, el del premio Nobel por la estructura del ADN).

46 *Taller en R : Anova con bloqueo*

Ya sabemos cómo procesar una anova con bloqueo en *Gnumeric*, si los datos vienen como en la tabla siguiente:

Mitochondria			
ATP production			
	pH 4	pH 6	pH 8
30^o	5	8	6
36^o	7	10	8
42^o	4	7	3

Como *Gnumeric* hace las cosas tan fáciles, R no es necesario para este tipo de casos. R se vuelve interesante cuando uno tiene una tabla natural, como la siguiente y de la cual pudo haber venido la tabla citada:

pH + Temp → Prod		
pH	Tem	Prod
4pH	30G	5
4pH	36G	7
6pH	36	10
8pH	42G	3
6pH	30G	8
4pH	42G	4
8pH	30G	6
6pH	42G	7
8pH	36G	8

Podemos usar la consola para hacer que *R* nos haga la anova con bloqueo:

```
#============================
#ANOVA CON BLOQUEO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
pH <- c(4,4,6,8,6,4,8,6,8)
pH <-as.factor(pH)
temp <- c(30,36,36,42,30,42,30,42,36)
temp <- as.factor(temp)
prod <- c(5,7,10,3,8,4,6,7,8)
Mitoc <-data.frame(temp,pH,prod)
Mitoc
w<-aov((Mitoc$prod ~ Mitoc$temp  + Mitoc$pH), projections = TRUE )
summary(w)
#
#Comparación de medias por pares según Tukey
#Se ordenan las diferencias entre medias antes del test
tt <- TukeyHSD(w, ordered = TRUE, conf.level = 0.95)
plot(tt)
```

El output coincide con el de *Gnumeric* pero además los nombres de los factores aparecen listados de forma natural. La *GUI* se queda corta para analizar este problema, seguramente porque éste es demasiado pobre en datos.

4.3. Anova bifactorial

Venimos considerando que un **experimento** estudia la respuesta de una variable específica, llamada **variable respuesta**, ante los estímulos ofrecidos por el mundo tanto externo como interno al sistema. La acción del mundo sobre la variable respuesta se describe a través de variables que pueden ser cuantitativas o categóricas. A las **variables cuantitativas** también se les llama **numéricas** y las **categóricas factores**.

La perspectiva del **análisis de varianza** es que la forma de cuantificar el efecto de una variable estímulo sobre el sistema es midiendo la variación que experimenta la variable respuesta cuando variamos los niveles de la variable explicativa. La **variación total** de la variable respuesta se puede particionar en dos partes: la que se explica por las variables que tenemos controladas mas el resto debido a las variables que no controlamos y cuyo efecto, medido en las variaciones de la variable de respuesta, se denomina ruido o azar. La variación debida al ruido se denomina **variación error** y la varianza correspondiente **varianza error**. Debido a que la única fuente de incertidumbre es el azar, un experimento es más exacto entre más pequeña sea la varianza error.

El **diseño de un experimento** tiene como objetivo optimizar la relación entre el costo de control y la minimización de la varianza error. Para ello contamos con varias estrategias. La más inmediata

es reunir más y más datos porque la naturaleza del azar es que entre más se le permite expresarse más se autoaniquila (**Primer teorema fundamental del método científico**). La segunda es hacer experimentos grandes en vez de varios pequeños porque los datos pesan más juntos que separados (**Segundo teorema fundamental del método científico**). En tercer lugar, entre más controlemos las variables que puedan afectar al sistema, la varianza total se conserva pero se particiona en más partes y por tanto menor será la varianza error y así el experimento será más contundente (**Tercer teorema fundamental del método científico**).

En un experimento de tipo **anova unifactorial** estudiamos el efecto de una variable estímulo que llamamos **variable experimental** sobre el promedio de la variable respuesta. Con frecuencia uno no tiene una variable experimental clara y definida sino una situación experimental compliacada para describir. Ejemplo: cuando llueve se vende menos helados. Eso de llover o no llover es fácil de decir, pero sucede que eso causa tantos cambios en el medio que tratar de especificarlos no se logra con una variable dicotómica (de dos valores). A las situaciones experimentales se les llama **tratamientos**.

El objetivo básico en un experimento tipo anova es dilucidar si la variable experimental influye o no sobre el promedio de la variable respuesta. El próximo paso en refinación después del diseño unifactorial es una **anova con bloqueo** en la cual se estudia el efecto de la variable experimental sobre el promedio de la variable respuesta pero controlando otra de las **variables de fondo**, las que producen el ruido o el azar. Y, ¿qué sigue después?

Una vez que uno ha considerado un exprimento con bloqueo, se presentan dos opciones: controlar otras variables de fondo o bien esclarecer la relación de interacción entre las variables que ya han sido estudiadas. En particular, la **anova bifactorial** estudia el efecto de dos variables experimentales sobre el promedio de la variable respuesta, dilucidando el efecto de cada una y el efecto de su interacción. ¿que qué es eso de la interacción? Para entederlo, considermos el siguiente caso:

Una mamá le está inculcando a su hija las buenas maneras de una mujer en su hogar. Pero la mamá ha cambiado mucho: al principio con tender la cama era suficiente, después se ponía contenta si la niña barría, o si trapeaba el baño, o si lavaba la loza. Pero de un tiempo para acá ya no se pone contenta a menos que se hagan todas las tres cosas. La descripción científica de esta historia es así: el sistema es la mamá. Las variables estímulos son los diversos componentes del comportamiento de la hija. Cuando la niña era pequeña, con que tendiera la cama era suficiente. Teníamos un factor binomial, con dos valores: tender o no tender. Similarmente, barrer o no barrer, trapear o no trapear, lavar la loza o no lavar también pueden ser considerados binomiales. En realidad, éso era al comienzo. Ahora hay que tender la cama muy bien, y trapear muy bien y no se le puede dejar nada de grasa a la loza. Evidentemente, ahora las descripciones binomiales no son suficientes. Ahora estamos en la época de las variables cuantitativas o numéricas y que pueden tomar toda suerte de valores intermedios. Y como se fuese poco, hay que hacer todas las cosas y muy bien hechas porque si no, ¡que cantaleta! Vemos que a medida que pasa el tiempo, la interacción entre las variables se torna más pesada que el efecto de las variables por sí sólas. Ni nos dimos cuenta a qué horas pasamos de nada a un experimento multifactorial fuertemente dependiente de interacciones. Y, ¿ el ruido? ¿en dónde está el ruido? El ruido está compuesto por todas las fluctuaciones en el proceso de aprendizaje: la niña está tan ocupada en sus sueños, sus muñecas y su telenovela favorita que se crea severa interferencia con la realidad, es decir, con barrer y trapear. A veces hay más interferencia, a veces hay menos.

Por el ejemplo anterior sabemos que las interacciones entre factores son situaciones comunes y corrientes. Pero, ¿cómo se describen y cómo se detectan? Y, ¿por qué no salen directamente de un experimento con bloqueo?

Un diseño con bloqueo es un instrumento para detectar el efecto de una variable experimental auque también puede detectar el efecto de la variable bloqueda. Pero no incluye el efecto de las interacciones entre las dos variables o factores. Para poder detectar la interacciones debemos primero que todo postular que existen, luego predecir cómo será su efecto observable, y después fabricar el instrumento de medida que las detecte. A todo el conjunto se le llama **diseño bifactorial**.

Paso uno: postulamos su existencia, como en el modelo siguiente que dice que todo registro y_{ijh} se debe al efecto combinado de una media global, μ, mas una desviación promedio de la media global debida al efecto del primer factor, α_j, mas una desviación promedio de la media global debida al efecto

del segundo factor, β_i, mas una desviación promedio de la media global debida a la interacción de los dos factores, $(\alpha\beta)_{ij}$, mas una desviación particularizante que se debe a todo lo que crea ruido o azar, ϵ_{ijh} .

$$y_{ijh} = \mu + \alpha_j + \beta_i + (\alpha\beta)_{ij} + \epsilon_{ijh}$$

Aclaremos por qué se necesitan 3 subíndices. La razón es que con dos subíndices tendríamos:

$$y_{ij} = \mu + \alpha_j + \beta_i + (\alpha\beta)_{ij} + \epsilon_{ij}$$

y ahora nos preguntamos: ¿Qué diferencia operativa hay entre el efecto de la interacción y el del azar? ¡No hay ninguna! En cambio en el modelo con 3 subíndices si puede distinguir claramente el efecto de la interacción del del azar y de los efectos de los factores por separado. La primera implicación de nuestra descomposición es que el registro de datos ya no cabe en una tabla que indique el efecto del factor uno contra el dos. Ahora tenemos datos que exigen una tabla de 3 dimensiones, es decir, para el tratamiento i del factor uno y el tratamiento j del factor dos, hay que hacer varias repeticiones, y no como antes que con un sólo dato teníamos.

Ya tenemos el modelo que postula la existencia de una interacción entre los dos factores y también la forma de hacer el experimento: por cada par de tratamientos (i, j) hay que hacer repeticiones. Nos falta entender qué predicciones observables podemos tener del efecto de las interacciones. Para verlo, tomamos el promedio sobre las repeticiones para el par (i, j) primero en el modelo antiguo, con bloqueo, y luego en nuestro nuevo modelo. En el diseño con bloqueo tenemos que

1. $\bar{y}_{i\bullet} = \mu + \beta_i$
2. $\bar{y}_{\bullet j} = \mu + \alpha_j$

La primera identidad dice que todos las celdas de la misma fila tienen el mismo promedio, pero que puede haber variaciones de fila a fila. La segunda identidad dice que todas las celdas de la misma columna tinen el mismo promedio aunque puede haber variaciones de columna a columna. Eso es lo que pasa en el diseño con bloqueo. En cambio, en el diseño bifactorial podemos tomar el promedio sobre las h repeticiones y así el tercer subíndice no aparece:

$$\bar{y}_{ij\bullet} = \mu + \alpha_j + \beta_i + (\alpha\beta)_{ij} + \bar{\epsilon}_{ij\bullet}$$

lo cual nos obliga a especificar la naturaleza del ruido: ϵ_{ijh} es una v.a. de media cero y desvación σ^2 que no depende ni de i ni de j ni mucho menos del número de la repetición h. A propósito, cuando todos los datos caben en un tabla de 3 dimensiones y el número de repeticiones es el mismo para todo par (i, j), decimos que tenemos un diseño **balanceado**. De lo contrario se dice que es **no balanceado**.

Con nuestras suposiciones estamos diciendo que todo lo que afecta a la variable respuesta se descompone en dos partes: los dos factores en estudio, que pueden tener interacciones, y todo lo demás que está totalmente desligado de los factores bajo control. Con dicha suposición tenemos que

$$\bar{y}_{ij\bullet} = \mu + \alpha_j + \beta_i + (\alpha\beta)_{ij}$$

Si promediamos de nuevo, sea sobre i, sea sobre j es preciso tener en cuenta que las alfas y las betas son desviaciones de la media y como tales sus sumas y premedios dan cero. Así tenemos las siguientes predicciones observables:

1. $\bar{y}_{i\bullet\bullet} = \mu + \beta_i + (\alpha\beta)_{i\bullet}$
2. $\bar{y}_{\bullet j\bullet} = \mu + \alpha_j + (\alpha\beta)_{\bullet j}$

Eso quiere decir que en el diseño bifactorial esperamos que los promedios cambien tanto a lo largo de las filas como de la columnas.

Lo demás es aplicar el truco del cero y seguir hasta encontrar las 3 efes, la f de las columnas, la de las filas y la de la interacción. Todo esto se hace hoy día con ayuda computacional. Cuando uno tiene una **tabla! de conteo**, que tabula los datos por tratamiento, se puede usar *Gnumeric*, pero si tiene una tabla natural, se puede usar *R*.

La H_o dice que los factores controlados no inciden sobre el valor promedio de la variable respuesta y por tanto los valor esperados de las medias bajo los diversos pares de tratamientos coinciden. El estadígrafo usado para juzgar la H_o es una F. Cuando la H_o es cierta, el valor de F_{exp} es cercano a uno. La H_a dice que hay al menos un par de medias diferentes. La H_o se rechaza mediante la F_{exp} con una cola, la superior, pues en la medida en que los factores influyen, la F_{exp} se hace más grande y se aleja de uno.

47 *Theory-ladeness, ciencia y filosofía*

Discutimos con motivo de la anova unifactorial el fenómeno llamado **theory-ladeness**, el cual hace referencia a que debido a que la ciencia es fundamentalmente un contraste entre lo que se cree y lo que se ve, uno no puede hacer ciencia a no ser que se tenga un modelo con qué mirar el mundo. Sucede, que los modelos pueden tener supuestos quizá difíciles de sacar a flote y que convierten a los modelos en instrumentos de medida sesgados hacia sus propias preconcepciones.

En relación con una **anova con bloqueo** tenemos que ésta no puede registrar una posible interacción entre la variable experimental cuyo efecto se estudia y la variable de bloqueo cuyo efecto quiere filtrarse para que la varianza error disminuya y la resolución del modelo aumente. ¿Por qué? Porque la falta de interacción entre los dos factores citado es realmente un supuesto del modelo. Es decir, cuando implementamos un experimento con diseño anova con bloqueo estamos sobrecargando nuestra ciencia con el supuesto de que los dos factores, el fundamental y el que se bloquea, no pueden tener interferencia. Para descargar la ciencia de tal suposición, todo lo que se necesitaba era formular un modelo donde las interacciones puedan existir para después ponerlo a prueba. Este vacío fue llenado por el modelo **anova bifactorial**.

Lo expuesto parece implicar que la ciencia es un programa en el que modelos cada más complejos ejecutan una cada más depurada limpieza de supuestos. ¿Hasta donde puede llegar este programa? ¿Hasta la verdad absoluta? Necesitamos dar un rodeo para responder:

En la filosofía de la ciencia al **método científico** como lo venimos exponiendo se le llama **método hipotético-deductivo**: se plantean modelos, se calculan sus predicciones bajo diversas valoraciones de sus parámetros que constituyen sus hipótesis nulas, y se ponen a prueba con los experimentos o con las observaciones. El modelo y sus supuestos quizá formen un conjunto lógico coherente, que no contiene ni produce contradicciones. Hasta 1931 era natural pensar que la lógica lo cubría todo y que por lo tanto toda verdad podía discutirse dentro de la lógica. Pero en ése año, un joven matemático que venía de Viena y que se llamaba Kurt Gödel a sus 25 años presentó unos muy extraños resultados de contabilidad en el mundo de las proposicones lógicas. Ellos indicaban que un discurso lógico que pretenda describir y estudiar un universo complejo tiene limitantes muy fuertes y había que afrontar una disyuntiva: o uno era totalmente consecuente y libre de contradicciones o uno tenía que conformarse con vacíos, con representaciones del universo inevitablemente parciales. Supóngase que tenemos diversas descripciones coherentes y parciales pero que entre todas cubran todos los temas: ¿qué pasa si las juntamos? Inevitablemente se producen contradicciones entre ellas.

Como resultado de la insuficiencia de la lógica para describir un universo complejo, la ciencia no tiene una relación clara con la verdad, la cual se define como lo que el universo (todo lo que existe) es en sí mismo. A uno le queda la incertidumbre de si lo que uno cree refleja fielmente el universo o si es una conclusión lógica de sus modelos pero que terguiversa al universo real, tal como él es.

Como si fuese poco, nosotros no tenemos acceso a la realidad tal cual ella es sino que todo lo que conocemos de la realidad es porque nuestros sentidos nos lo han enseñado. Los sentidos son instrumentos de medida cuyo diseño obedece a un modelo que se puede dilucidar a posteriori, por ejemplo, la arquitectura de un ojo se rige por el modelo de la óptica geométrica, pero sus células sensibles, los bastones y conos, se rigen por el modelo semiclásico de la interacción materia-energía que representa a

las moléculas como elementos cuánticos y a la luz como elemento clásico. Por otro lado, la conciencia no ve lo que los sentidos le presentan, sino que el cerebro produce para la conciencia un informe en el que predominan los patrones o modelos de alto nivel y de ésa form los detalles se interpetan en favor de los modelos vencedores, que son los que más poder organizativo tienen.

En suma, toda ciencia es vanidad: si buscas la verdad, en ella no la encontrarás. O si uno prefiere, la ciencia es un arte delicado. Y con esfuerzo y mucha atención uno puede encontrar estupendas aproximaciones a la verdad. De hecho, la supervivencia de tantas especies de seres vivos durante tantos millones de años nos indican que los modelos en que se basan los sentidos y el proceso de fabricación de la realidad virtual que ven sus conciencias es de muy alta calidad cuando se la mide por tasas de supervivencia.

Y a todas esta, ¿cómo es posible que existen personas que mueran por su verdad? ¿De dónde les sale tanta seguridad de que lo que creen tiene una relación fiel con la verdad, la realidad de lo que existe? Conocí yo a una niña con la hermosura de los lilios del campo y que no creía que el hombre haya surgido por evolución a partir del chimpancé, sino qu creía que Dios lo había hecho directamente y como tal tenía derecho a juzgarlo. Ella trabajaba y estudiaba su posgrado en un instituto de investigación. Y como no podía dar una evidencia científica de sus creencias en tanto que sí había muchas, muchísimas de la evolución, al fin cedió ante la presión de la jefatura y renunció. Dadas las circunstancias políticas y económicas en que vivía, tal decisión significaba la renuncia a una profesión.

Es interesante notar que en muchas ocasiones los poseedores de la verdad no tienen poder para tomar decisiones represivas sino tan sólo para presionar. Pero la forma de hacerlo puede ser tan efectiva que al fin el presionado se fatiga hasta el límite y renuncia. Tal fue precisamente el caso de Giordano Bruno: a él no lo quemaron por sus creencias sino porque se negó a sequirle el juego de presión de la jefatura que en represalia lo clasificó como hereje, para los cuales el castigo del momento era la hoguera. Mas detalles en:

`http://galileo.rice.edu/chr/bruno.html`

48 ♣ *Análisis bifactorial de varianza* *Ejemplo sobre Excel*

Nosotros ya habíamos trabajado un ejemplo en el cual estudiábamos la dependencia sobre la producción de ATP de cambios en la temperatura y el pH. Asumimos que la temperatura y el pH no tienen efectos de interacción. Por supuesto, tal suposición es falsa: el pH mide la concentración de protones y su mobilidad es más fuerte a medida que aumenta la temperatura. Por consiguiente, ellos pueden ejercer un mayor efecto sobre la mitocondria. La razón es que en la producción de ATP se involucran muchos complejos proteícos que funcionan como motores que operan por gradiente de protones y por lo tanto son muy sensibles a los protones y a su mobilidad. Por lo tanto, un modelo sin interación es abusivo. Corrijámos dicho error.

Consideremos la siguiente tabla que nos presenta el efecto de la temperatura y del pH sobre la producción de ATP:

Mitocondrias			
ProducciOn de ATP			
	pH 4	pH 6	pH 8
30^o	5	8	6
	4	9	6
	5	7	5
36^o	8	10	8
	7	9	9
	8	11	7
42^o	5	7	2
	4	7	3
	6	7	1

Las hipótesis nulas son 3:

1. Un cambio en pH no produce un cambio en la producción promedio de ATP.

2. Un cambio en temperatura no produce un cambio en la producción promedio de ATP

3. Un cambio sincrónico en la temperatura y el pH no produce un cambio promedio en la produccin de ATP.

Las hipótesis alternas son las negaciones correspondientes. Presentamos enseguida el output de Excel sobre los datos dados.

Anova para pH y Temperatura						
Origin of variation	SS	DF	Mean Square	Fisher	Probab	Critical F
Among rows (T)	69.56	2	34.78	52.17	3.23 E-08	3.55
Among columns (pH)	49.56	2	24.78	37.17	4.06E-07	3.55
Interaction	15.56	4	3.89	5.83	0.0034	2.93
Within groups	12	18	0.67			
Total	146.67	26				

De acuerdo a esta tabla, el estadígrafo de Fisher entre filas es 52.17 que sobrepasa por mucho el valor crítico que es 3.55. Por lo tanto, consideramos que un cambio en temperatura causa un claro corrimiento de la producción promedio de ATP. El efecto de un cambio de pH tiene una lectura muy similar: un cambio en pH causa un cambio en la producción promedio de ATP. Además, vemos que la temperatura y el pH interactúan para crear efectos cooperativos sobre la producción promedio de ATP.

Aprendamos a reproducir a mano estos mismos resultados:

Ahora tenenos 3 subíndices X_{ijl}. El primero es para el tratamiento a lo largo de las filas, el segundo para los tratamientos sobre columnas y el tercero es para considerar las réplicas para cada par de valores ij de tratamientos. Ejemplos: $X_{111} = 5$, $X_{112} = 4$,$X_{321} = 7$,$X_{131} = 6$,$X_{333} = 1$. En nuestro ejemplo, los ubíndices corren desde 1 a 3, pero en general corren así: i de 1 a a, j de 1 a b y k de 1 a c. $N = abc = 27$.

calculamos varios tipos de sumas: las verticales para un pH dado, las horizontales para la temperatura. Todo esto debe ser indicado directamente en la tabla siguiente:

Mitocondria				
ProducciOn de ATP				
	pH 4	pH 6	pH 8	$\sum$
30^o	5	8	6	19
	4	9	6	19
	5	7	5	17
$\sum$	14	24	17	$\sum\sum$ =55
36^o	8	10	8	26
	7	9	9	25
	8	11	7	26
$\sum$	23	30	24	$\sum\sum$ =77
42^o	5	7	2	14
	4	7	3	14
	6	7	1	14
$\sum$	15	21	6	$\sum\sum$ =42
$\sum\sum$	52	75	47	$\sum\sum\sum$ =174

También calculamos la matriz de cuadrados, cuyas entradas son los cuadrados de las entradas de la matriz original:

Cuadrados - Mitocondria ProducciOn de ATP				
	pH 4	pH 6	pH 8	$\sum$
30^o	25	64	36	125
	16	81	36	133
	25	49	25	99
$\sum$	66	194	97	357
36^o	64	100	64	228
	49	81	81	211
	64	121	49	234
$\sum$	177	302	194	673
42^o	25	49	4	78
	16	49	9	74
	36	49	1	86
$\sum$	77	147	14	238

Ahora procesamos estas tablas, pero usamos inglés pues ya es estándar el uso de la terminología en dicha lengua:

1. T =Total sum (original matrix)= $T = 55 + 77 + 42 = 174$

2. Q =Total sum of squares (matrix of squares): $Q = 357 + 673 + 238 = 1268$

3a. L =Total sum of squares of columns sums (original matrix)= $L = 14^2 + 24^2 + 17^2 + 23^2 + 30^2 + 24^2 + 15^2 + 21^2 + 6^2 = 196 + 576 + 289 + 529 + 900 + 576 + 225 + 441 + 36 = 3768$

3b. I =Each case was replicated 3 times, hence we get $I = L/3 = 3768/3 = 1256$

4. A =Averaged great square = $A = T^2/N = 174^2/27 = 30276/27 = 1121{,}33$

5. Total SS= $Q - A = 1268 - 1121{,}33 = 146{,}67$

6. Among columns SS = $I - A = 1256 - 1121{,}33 = 134{,}67$

7. Within columns SS= Total SS- Among columns SS= $146{,}67 - 134{,}67 = 12$

8. R =Sum of squares of total rows (original matrix) and averaged over the number of data per row= $R = \frac{55^2+77^2+42}{9} = 3025 + 5929 + 1764 = \frac{10718}{9} = 1190{,}89$

9. C =Sum of squares of total colums (original matrix) and averaged over the number of data per column=$\frac{52^2+75^2+47^2}{9} = \frac{2704+5625+2209}{9} = \frac{9908}{9} = 1170{,}89$

10. Among Rows SS= $R - A = 1190{,}89 - 1121{,}33 = 69{,}56$

11. Among columns SS= $C - A = 1170{,}89 - 1121{,}33 = 49{,}56$

12. Interaction SS= Among columns SS -Among Rows SS- Among columns SS =134.67-69.56-49.56=15.55

13. Degrees of freedom: Among rows= 3-1=2; Among columns= 3-1=2; Interaction= $2 \times 2 = 4$; within (error)= rc(n-1)=$9 \times 2 = 18$.

14. Mean SS:

For rows = $\frac{Among\ Rows\ SS}{DF} = \frac{69{,}56}{2} = 34{,}78$

For columns = $\frac{Among\ columns\ SS}{DF} = \frac{49{,}56}{2} = 24{,}78$

For interaction = $\frac{Interaction\ SS}{4} = \frac{15{,}55}{4} = 3{,}89$

For error= = $\frac{Within\ columns\ SS}{18} = \frac{12}{18} = 0{,}67$

16. Fisher values:

For rows = $\frac{Mean\ SS\ for\ Rows\ SS}{Mean\ SS\ for\ error} = \frac{34{,}78}{0{,}67} = 51{,}9$

For columns = $\frac{Mean\ SS\ for\ columns\ SS}{Mean\ SS\ for\ error} = \frac{24{,}78}{0{,}67} = 36{,}98$

For interaction = $\frac{Mean\ SS\ forInteraction\ SS}{Mean\ SS\ for\ error} = \frac{3{,}89}{0{,}67} = 5{,}81$

Como podemos ver, nuestros resultados que ha sido calculados con 2 cifras significativas coinciden practicamente con los de Excel. Comparando con los valores críticos, podemos concluir que un cambio en la temperatura o en el pH produce un cambio en la producción promedio de ATP. Además podemos asegurar que la temperatura y el pH tiene sinergia produciendo efectos cooperativos,

49 ***Ejemplo de anova bifactorial en Gnumeric*** *Tools + statistical analysis + ANOVA + two factor*

Primero uno teclea la siguiente matriz de datos que reporta el puntaje dado por la mamá a su niña que está aprendiendo a comportarse como la ama de su casa. Vemos que al cambiar los niveles de cada factor, hay cambio en los promedio del puntaje. Por tanto, esperamos que cada factor sea importante, es decir, sus p-value deben ser muy pequeños. De manera semejante, los promedios por par de tratamientos son diferentes tanto a lo largo de las filas como de las columnas. Por consiguiente, esperamos que la interacción sea estadísticamente significativa. La tabla es la siguiente:

Puntaje para tareas caseras		
	Trapear	No trapear
barrer	5	2.1
	4.3	2.2
	4.4	2.4
	4.6	2.1
no barrer	2.2	−1
	2.2	−2
	2.1	−1
	1.9	−2

Es hermoso ver el despligue que hace *Gnumeric* al analizar esta tabla de conteo.

50 ***Ejemplo en R***

Cuando uno tiene una tabla natural, uno usa R. Para lo cual uno adpata el siguiente programa a su archivo específico. El programa toma un paquete de datos que viene con R y le hace una anova bifactorial.

```
#==============================
#ANOVA BIFACTORIAL
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#
#Parte 1
#Cargar paquete
data(CO2)
#Publicar  información sobre el paquete
#teclear q para salir de la ayuda
#help("CO2")
#
#Parte 2: correr después
#Listar datos
CO2
#Anova bifactorial
# uptake = variable respuesta
#Treatment, type = variables explicativas
AnovaModel <- (lm(uptake ~ Treatment*Type, data=CO2))
#Anova(AnovaModel)
summary(AnovaModel)
w <-CO2
#Tabla bidimensional de frecuencias absolutas
z<-table(w$Treatment,w$Type)
z
```

Podemos verificar que un modelo bifactorial es igual a un modelo con bloqueo pero cuando se incluyen las interacciones. El símbolo para la interacción de dos factores es dos puntos. El nuevo programa es:

```
#============================
#ANOVA BLOQUEO MAS INTERACCIONES
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#
#Parte 1
#Cargar paquete
data(CO2)
#Publicar  información sobre el paquete
#teclear q para salir de la ayuda
help("CO2")
#
#Parte 2
#Listar datos
CO2
#Anova bifactorial
# uptake = variable respuesta
#Treatment, type = variables explicativas
AnovaModel.1 <- (lm(uptake ~ Treatment+Type + Treatment : Type, data=CO2))
#Anova(AnovaModel.1)
w <-CO2
#Tabla bidimensional de frecuencias absolutas
z<-table(w$Treatment,w$Type)
z
```

Los dos últimos programas deben dar exactamente el mismo output. Eso ilustra que

$$A * B = A + B + A : B$$

lo cual dice que el operador estrella indica análisis bifactorial y contiene los efectos de los factores por separado y también sus interacciones.

4.4. MANOVAS = Anova multifactorial

La palabra **manova** es una contracción de *multivariate analysis of variance* y es una extensión natural de la filosofía optimizante del diseño de experimentos: si los datos pesan más juntos que separados, entonces los diseños deben generalizarse al máximo. Hemos visto cómo se generalizan incluyendo mas y mas variables experimentales, que se controlan, lo mismo que sus interacciones. Pero falta incluír en esta generalización lo que las manovas si tienen: tratamiento conjunto no de una sino de varias variables respuesta y también con sus interacciones.

El formalismo matemático de las manovas pasa de expresiones algebraícas de números a expresiones algebraícas con matrices. No se considera que se tenga una teoría matemática madura de las manovas pues todo lo que tenemos son aproximaciones discutibles, como la de Wilks o de Pillai que tratan de ajustar estadígrafos relacionados con determinantes de las matrices. La moda actual en la ciencia es preferir las cosas sencillas y claras a las muy finas pero engorrosas. Sin embargo y debido a que las manovas prometen hacer un análisis exhaustivo y eficiente, éstas han permanecido, siguen siendo importantes y se les declara larga vida.

La hipótesis nula de una manova es que las variables experimentales, las que se controlan, no tienen ninguna incidencia ni sobre el promedio de cada una de las variables respuesta ni sobre sus interacciones. El estadígrafo de contraste es en definitiva una F. La H_o se rechaza con la cola superior. La hipótesis alterna dice que algún subconjunto de factores experimentales o de sus interacciones inciden sobre algún subconjunto de variables respuesta o sobre sus interacciones.

En el caso de que la hipótesis nula se rechace, uno quisiera saber exactamente cuáles variables son las responsables de qué. Eso se logra haciendo análisis de varianza multivariado ordinario, con una sóla variable respuesta. Pero atención: existe la posibilidad de que la hipótesis nula de la manova se rechace y sin embargo ninguna anova de rango menor sea significativa. Éso es lo que significa que los datos juntos pesan más que los datos por separado. Sin embargo, uno tiene la opción de guiarse por las anovas ordinarias y los p-value: los más p-values pequeños muestran las variables más efectivas.

En los archivos de datos adjuntos a *R* hay suficientes opciones para experimentar. Todo éso queda muy fácil si se usa la *GUI*, pues las anovas multifactoriales tienen un lugar preferencial en la *GUI* en la sección de medias.

Las siguientes instrucciones publican la ayuda que a cerca de las manovas viene con *R*:

```
help(manova)
help(summary.manova)
#Ejemplo dado por R
```

El siguiente es un ejemplo que viene con *R* al cual el autor le ha añadido algunos comentarios. Explicamos las cosas nuevas antes de verlo:

Sabemos utlizar el procedimiento *c()* (de concatenar) para crear un vector de datos. Si los datos son números, se sobreentiende que el vector es numérico. Para unir varios vectores en una tabla, hemos usado el procedimiento *data.frame()*. Este procedimiento se generaliza en otro que pega no sólo vectores sino también tablas y se llama *cbind()*. Para cambiar de tipo un tabla numérica se usa *factor()* que la convierte en categórica.

El procedimiento

```
uso <- factor(gl(2,10), labels=c("Bajo", "Alto"))
```

crea un vector, *uso*, de tipo factor o categórico que tiene 2 niveles o categorías, *Bajo* y *Alto*, y que se listan en réplicas seguidas de a 10. Dicha instrucción produce el vector:

Bajo Bajo Bajo Bajo Bajo Bajo Bajo Bajo Bajo Bajo Alto Alto Alto Alto Alto Alto Alto Alto Alto Alto

El procedimiento

```
aditivo <- factor(gl(2, 5, length=20), labels=c("Bajo", "Alto"))
```

crea un vector de tipo factor o categórico que tiene 2 niveles o categorías, *Bajo* y *Alto*, y que se listan en réplicas seguidas de a 5. Las réplicas se alternan hasta llegar a un total de 20 datos. Dicha instrucción produce el vector:

Bajo Bajo Bajo Bajo Bajo Alto Alto Alto Alto Alto Bajo Bajo Bajo Bajo Bajo Alto Alto Alto Alto Alto

La siguiente es una tabla natural que presenta los datos de un experimento que estudia la dependencia de 3 variables respuesta (*degaste, brillo, opacidad*) ante dos variables experimentales: *uso* y *aditivo (cantidad de aditivo protector)* que se le da a una cinta plástica.

```
#==============================
#TABLAS GRANDES
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
## Tabla natural de un experimento
#Una tabla natural  se define por sus 3 columnas
desgaste <- c(6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
              6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6)
brillo <- c(9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
               9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2)
```

```
opacidad <- c(4.4, 6.4, 3.0, 4.1, 0.8, 5.7, 2.0, 3.9, 1.9, 5.7,
                  2.8, 4.1, 3.8, 1.6, 3.4, 8.4, 5.2, 6.9, 2.7, 1.9)
#El procedimiento cbind (pegar) forma grandes tablas a partir de otras mas pequeñas
Y <- cbind(desgaste,brillo, opacidad)
Y
#El procedimiento factor transforma un vector numérico
# en otro de tipo factor (categórico)
#gl(n,k) significa que hay n niveles con k réplicas
#gl(2,10) significa que hay 2 niveles con 10 réplicas
#labels=c("Bajo", "Alto") significa que los dos niveles son Bajo y Alto
uso <- factor(gl(2,10), labels=c("Bajo", "Alto"))
uso
aditivo <- factor(gl(2, 5, length=20), labels=c("Bajo", "Alto"))
aditivo
Tabla <- cbind( uso, aditivo, Y)
Tabla
```

El análisis de todo experimento, con datos y manova, es el siguiente:

```
#==============================
##MANOVA: multivariate analysis of variance (comparación de medias)
## Ejemplo sobre la producción de cinta plástica tomado de Krzanowski (1998, p. 381)
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Una tabla natural  se define por sus 3 columnas
desgaste <- c(6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
               6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6)
brillo <- c(9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
                9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2)
opacidad <- c(4.4, 6.4, 3.0, 4.1, 0.8, 5.7, 2.0, 3.9, 1.9, 5.7,
                  2.8, 4.1, 3.8, 1.6, 3.4, 8.4, 5.2, 6.9, 2.7, 1.9)
#El procedimiento cbind (pegar) forma grandes tablas a partir de otras mas pequeñas
Y <- cbind(desgaste,brillo, opacidad)
Y
#El procedimiento factor transforma un vector numérico
#en otro de tipo factor (categórico)
#gl(n,k) significa que hay n niveles con k réplicas
#gl(2,10) significa que hay 2 niveles con 10 réplicas
#labels=c("Low", "High") significa que los dos niveles son Bajo y Alto
uso <- factor(gl(2,10), labels=c("Bajo", "Alto"))
uso
aditivo <- factor(gl(2, 5, length=20), labels=c("Bajo", "Alto"))
aditivo
#MANOVA para estudiar la incidencia de uso y aditivo sobre
#las variables desgaste, brillo, opacidad y sus interacciones:
fit <- manova(Y ~ uso * aditivo)
#Tablas anova  multifactorial (una variable respuesta a la vez)
summary.aov(fit)
#Tabla de estadígrafos de  Pillai
#
summary(fit)
#Tabla de estadígrafos de Wilks
summary(fit, test="Wilks")
```

Al ver la salida de este programa, uno se da cuenta que se interpreta de igual forma que todo los

visto hasta ahora. En particular, las dos variables experimentales fueron seleccionadas sabiamente y dió lo que se esparaba, que un cambio entre sus diversos niveles cambia significativamente las variables respuesta. Más detalladamente, ambas variables influyen sobre el desgaste, el brillo se ve influenciado por el régimen de uso pero no por el de aditivo, y en cuanto a opacidad, el experimento no revela sensibilidad a los cambios sufridos en las variables experimentales. Significancia por defecto = 0.05.

51 ***Estudio de supuestos con R***

Las hipótesis nulas de las manovas se calculan asumiendo que las variables de salida tienen una distribución normal, pero recordemos que las manovas toman los datos en su conjunto no separadamente. Por consiguiente, el presupuesto de normalidad debe cumplirse no sólo para las variables por separado sino para todas ellas tomadas en conjunto. Por tanto, se asume **normalidad multivariada** lo cual significa que se generalzia de una normal, una campana de Gauss a una montaña en n-dimensiones que se ve desde todo lodo como una campana. También se asume que la variabilidad es homogénea, y eso implica que las varianzas y las covarianzas son las mismas independoentemente de los tratamientos que se tomen.

Se ha observado que todo esto es demasiado pesado para los que aman el rigor pues casi nunca se cumplen tales supuestos. Buscando remedios al problema, se considera que lo primero que debe hacerse es quitar de la muestra a todos los outliers, aquellos valores que están demasiado alejados del centro de la distribución. Para ello hay que detectarlos primero, lo cual se hace adaptando el siguiente programa que viene con *R* y que requiere el paquete *mvoutlier*

```
#==============================
# DETECCION DE OUTLIERS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Paquete necesario
library(mvoutlier)
#Dibujo divariado
outliers <- aq.plot(mtcars[c("mpg","disp","hp","drat","wt","qsec")])
#Muestre la lista de outliers
outliers # show list of outliers
```

Para poner a prueba la normalidad de una única variable, se usa el test de Kolmogorov-Smirnov o el criterio de quantil-quantil o el test de Shapiro-Wilk.

El criterio Q-Q se lee en un dibujo apropiado y se interpreta igual que el test K-S: lo que no de una línea recta es un índice de anormalidad que puede ser o no estadísticamente significativo:

```
#==============================
#GRAFICA  Q-Q
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
attach(mtcars)
qqnorm(mpg)
qqline(mpg)
```

El test de Shapiro-Wilk de normalidad unidimensional:

```
#==============================
# TST DE SHAPIRO PARA NORMALIDAD
#Limpia la memoria
rm(list = ls())
#Modo gráfico
```

```
par(mfrow=c(3,2), pch=16)
#x= vector numérico
shapiro.test(x)
```

Para las manovas, usamos la correspondiente generalzación multivariada que da el prefijo m y que corre sobre matrices:

```
#==============================
#TEST MULTIVARIADO DE NORMALIDAD
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Paquete necesario
library(mvnormtest)
#test Shapiro para multinormalidad
mshapiro.test(M)
```

Acá hay otra opción:

```
#==============================
# ESTUDIO GRAFICO DE NORMALIDAD
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#datos = matriz
x <- as.matrix(datos)
#Se calcula el centrodie
center <- colMeans(x)
n <- nrow(x); p <- ncol(x); cov <- cov(x);
#Matriz de distancias
d <- mahalanobis(x,center,cov)
qqplot(qchisq(ppoints(n),df=p),d,
  main="Dibujo QQ de normalidad multivariada",
  ylab="Mahalanobis D2")
abline(a=0,b=1)
```

Para medir la homogeneidad de la varianzas usamos el terst de Bartlett que es no paramétrico.

```
# TEST DE BARTLETT PARA HOMOGENEIDAD DE VARIANZAS
bartlett.test(y~G, data=mydata)
```

También se puede usar el test de Figner-Killen

```
# # TEST DE FIGNER-KILLEN  PARA HOMOGENEIDAD DE VARIANZAS
#y es numérico
#G= factor de agrupamiento
fligner.test(y~G, data=mydata)
```

Hay untest gráfico para la homogeneidad de varianzas debido a Brown-Forsyth, necesita el paquete *HH*:

```
#==============================
# TEST GRAFICO DE HOMOGENEIDAD DE VARIANZAS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
```

```
par(mfrow=c(3,2), pch=16)
library(HH)
#y es numérico
#G= factor de agrupamiento
hov(y~G, data=mydata)
plot.hov(y~G,data=mydata)
```

Para estudiar la homogeneidad de las covarianzas puede usarse una propuesta debida a Andy Liaw quien adaptó a R un código para MatLab y que aparece en:

`http://finzi.psych.upenn.edu/R/Rhelp02a/archive/33330.html`

Capítulo 5

Regresión lineal de mínimos cuadrados

```
Los efectos son proporcionales a la causa.
```

52 ***Objetivo: Tenemos un caso de regresión cuando esperamos una relación causa-efecto entre una variable estímulo y una variable de respuesta de un sistema. Pero además, si la magnitud de la respuesta es proporcional a la del estímulo, tenemos una situación de regresión lineal. Nuestro objetivo es estudiar cómo se hace regresión lineal en presencia de ruido aleatorio cuando se aplica el método conocido como mínimos cuadrados.***

53 *Nuestro camino comienza desde la teoría básica, para agarrar los conceptos que dan lugar a las aplicaciones y a la discusión, y después iremos a grandes pasos.*

5.1. Regresión lineal

54 ***Ejemplo*** *El caminante.*

Uno espera que el espacio recorrido por un caminante sea aproximadamente proporcional al tiempo que el lleva caminando sin importar que a veces camine un poco más rápido o un poco más despacio. Sea x el tiempo que lleva caminando y y el espacio recorrido. Si el caminante fuese a una velocidad uniforme β, el espacio recorrido y se relaciona con el tiempo que lleva caminando x por

$$y = \alpha + \beta x$$

La constante α es el espacio inicial, es decir, la distancia entre el punto donde el caminante inicia su recorrido y un punto de referencia elegido.

55 ***Precaución: Nosotros usamos α para indicar dos cosas muy diferentes: es, por un lado, el nivel de significancia de una prueba y por otra es el parámetro libre en la regresión lineal, pero ninguna confusión será posible si uno presta atención al contexto.***

Pero como a veces se camina a una velocidad un poquito mayor que β y a veces un poco menor, podemos proponer que las variables estímulo-respuesta se relacionan mejor por:

$$y = \alpha + \beta x + \epsilon$$

donde ϵ es una v.a., pues uno nunca sabe qué podrá distraer o emocionar al caminante causando una desacelere o acelere en su caminar. Como de costumbre, el ruido ϵ se supone que tiene una distribución normal con media $\mu = 0$ y desviación σ. Una desviación grande indica que el caminante es muy variable en su velocidad.

Este modelo se denomina modelo de **regresión lineal** y se aplica cuando hay una clara relación de proporcionalidad causa-efecto entre la variable x y la variable y. Uno habla de **regresión no lineal** cuando no hay proporcionalidad entre la causa y el efecto.

Cuando nosotros no tenemos ninguna relación de causa-efecto pero observamos una relación lineal entre dos variables, nosotros usamos **correlación**.

Aprendamos el despliegue operacional de la regresión lineal.

56 ***Ejemplo*** *En muchas profesiones se observa que el ingreso mensual es proporcional al tiempo de experiencia. En la vida real se nota que el profesional logra una época de maduración en las cuales sus ingresos aumentan de forma inusual, lo cual demanda unos 20 a 25 años de sufrimiento. Veamos si los siguientes datos, para pocos años de experiencia, se ajustan a un modelo lineal. Los datos los reportamos por pares de la forma (tiempo en años, sueldo en unidades arbitrarias): (3,5), (2,5), (2,4), (4,9) (3,7), (5,11).*

Solución:

Paso 1: Nosotros dibujamos los puntos sobre un plano cartesiano, lo cual se denomina un **diagrama de dispersión**.

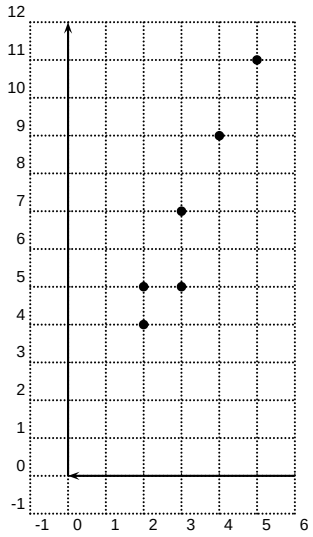

Figure 5.0. Los años de experiencia en el eje horizontal, el sueldo en el eje vertical.

Paso 2: Mirando el diagrama de dispersión, uno se forja una idea sobre aquella línea recta que mejor se ajusta a los datos, es decir, aquella que mejor parece representar los datos.

Cuando la línea viene de unos datos particulares, de una muestra, nosotros usamos para la línea la ecuación con letras latinas:

$y = a + bx$.

Pero cuando nosotros formulamos un modelo, el cual debe corresponder a un muestreo con un número infinito de datos, usamos letras griegas

$y = \alpha + \beta x$

En nuestro caso, la línea que mejor se ajusta a nuestros datos parece pasar por el punto más cercano al origen y por el más lejano. Esos puntos son: (2,4) y (5,11), los cuales nos generan una línea (en letras latinas):

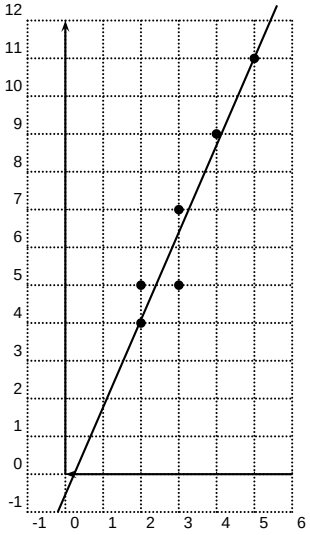

Figure 5.1. Estimación visual de la línea que mejor representa los datos, aquella que causa un mínimo de descontento global.

La pendiente de dicha línea es
$b = \frac{y_2 - y_1}{x_2 - x_1} = \frac{11-4}{5-2} = \frac{7}{3} = 2{,}33$
Por lo tanto, la ecuación de la línea es
$y = a + bx = a + 2{,}33x$
Para hallar a estudiamos un punto cualesquiera, por ejemplo, (2,4):
$4 = a + 2{,}33(2) = a + 4{,}66$
por lo que $a = 4 - 4{,}66 = -0{,}66$
Por consiguiente, nuestra **estimación visual de la línea de regresión** es
$y = -0{,}66 + 2{,}33x$
Mirando la nueva gráfica, vemos que es razonable creer que el ingreso mensual es proporcional a la experiencia, es decir que se justifica un modelo lineal, pues los puntos se ajsustan bien a la línea.

Ahora pasamos de lo intuitivo a los métodos rigurosos para sacar la línea de regresión de mínimos cuadrados. La metodología es la siguiente:

Paso 3: Construímos la siguiente tabla:

Regression de ingresos mensuales (y) vs tiempo de experiencia x					
	x	y	xy	x^2	y^2
	3	5	15	9	25
	2	5	10	4	25
	2	4	8	4	16
	4	9	36	16	81
	3	7	21	9	49
	5	11	55	25	121
Sumas	$\sum x$ =19	$\sum y$ =41	$\sum xy$ =145	$\sum x^2$ = 67	$\sum y^2$ =317

Paso 4: Calculamos los valores medios tanto de y como de x. Como en este caso tenemos 6 pares de datos:
$\bar{x} = \sum x/6 = 19/6 = 3{,}17$
$\bar{y} = \sum y/6 = 41/6 = 6{,}83$

Paso 5: Ahora nosotros podemos calcular la línea recta, $y = a + bx$, que mejor se ajusta a los datos, la cual minimiza el descontento cuadrático global de todos y cada uno de los puntos al ser representados por una línea.

57 ***Notación*** *La siguiente notación ayuda a hacer muchos cálculos relacionados con regresión:*

$SS_{xx} = \sum x^2 - n\bar{x}^2$;
$SS_{xy} = \sum xy - n\bar{x}\bar{y}$;

Con esta notación, la línea de regresión (min cuadrados): $y = a + bx$ se determina por
$b = \frac{SS_{xy}}{SS_{xx}} = \frac{\sum xy - n\bar{x}\bar{y}}{\sum x^2 - n\bar{x}^2}$;
$a = \bar{y} - b\bar{x}$.

Para nuestro ejemplo, esto se convierte en
$b = \frac{145 - 6(3{,}17)(6{,}83)}{67 - 6(3{,}17)^2} = \frac{15{,}09}{6{,}71} = 2{,}24$
$a = \bar{y} - b\bar{x}$
$a = 6{,}83 - (2{,}24)(3{,}17) = 6{,}83 - 7{,}10 = -0{,}27$

Así, la línea de regresión de mínimos cuadrados es
$y = -0{,}27 + 2{,}24x$

Comparamos esta respuesta con la que habíamos sacado visualmente:
$y = -0{,}66 + 2{,}33x$
Comparando las dos ecuaciones, la sacada por cálculo gráfico y la sacada por fórmulas, vemos que hay concordancia y por tanto podemos confiar en que hemos hecho bien las cosas.

5.2. Método de mínimos cuadrados

Hemos visto el cómo de la línea de regresión pero no hemos visto el por qué. Es conveniente saber el por qué para poder entender de qué forma los resultados dependen de un método y así poder dejar la puerta abierta para la investigación de otras metodologías.

Una noción intuitiva de la línea de regresión se puede lograr con el siguiente experimento mental: uno ata con un resorte cada punto a una varilla pero asegurándose de que todos los resortes queden bien tensionados. Después uno libera el sistema a su propio destino hasta que la tensión blobal se minimice. La posición final de la varilla estará globalmente lo más cerca posible de todos los puntos y nos dará una idea de la línea de regresión.

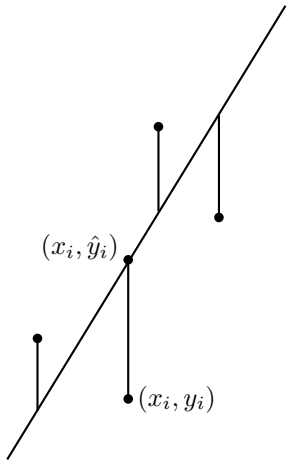

Figure 5.2. Para hallar la línea de regresión por el método de los resortes que mejor se ajuste a un conjunto de puntos dado se ata un resorte desde cada punto a una varilla recta. Se suelta el sistema para que la tensión global se minimice. Se tiene cuidado de que los resortes queden verticales en su posición definitiva. La posición final de la varilla representará la línea buscada.

En vez de resortes, podemos hacer una variante que consiste en reemplazar los resortes por hilo y templarlos a mano con el único requisito de que la cantidad total de hilo sea la mínima. Esto nos producirá la línea de regresión por el método del hilo templado. Y como ya tenemos dos metodologías podemos preguntarnos: ¿darán los dos métodos la misma línea en todos los casos? Si uno comete errores pequeños en la metodología de alguno de ellos, ¿existirán grandes cambios en la respuesta?

Hemos, pues, formulado dos preguntas importantes y para poder estudiarlas necesitamos formalizar las metodologías. El método de los resortes tiene como objetivo minimizar la tensión global mientras que el método del hilo templado tiene como objetivo minimizar el hilo total gastado. Sea (x_i, y_i) el punto número i, donde hay n puntos. Sea $y = a + bx$ la ecuación de una línea cualquiera que no sea vertical. El punto sobre la línea que queda en la dirección vertical de (x_i, y_i) es $(x_i, \hat{y}_i) = (x_i, a + bx_i)$. La cantidad de hilo entre (x_i, y_i) y $(x_i, \hat{y}_i)$ es $\mid y_i - \hat{y}_i \mid = \mid y_i - (a + bx_i) \mid = \mid y_i - a - bx_i \mid$.

El hilo total gastado es entonces

$$H = \sum_i^n \mid y_i - a - bx_i \mid$$

La línea de regresión por el método del hilo tiene que minimizar la función H sobre todos las líneas, es decir, sobre todos las dupletas de la forma (a, b). Hay muchos métodos para resolver este problema. Uno de ellos es por ensayo y error: se toma una pareja (a, b), se calcula H. Se toma otra pareja y se vuelve a calcular H. Uno compara cual H es el menor y se queda con el (a, b) correspondiente. Después una prueba otra parjea y así hasta que uno se canse. O quizá uno quiera ayudarse de un programa en Java.

Veamos ahora qué sucede con el método de los resortes. El objetivo en este caso es minimiar la tensión global. La tensión de cada cuerda es la fuerza generada por su elongación que es proporcional a la distancia entre el punto y la línea, pues entre más se elonge un resorte, más fuerza hace. Es decir, la tensión asociada al punto (x_i, y_i) es proporcional a la distancia vertical entre dicho punto y la línea, lo cual da $k \mid y_i - \hat{y}_i \mid = k \mid y_i - (a + bx_i) \mid = k \mid y_i - a - bx_i \mid$. Por ende, la tension global es

$$T = k \sum_i^n (signo) \mid y_i - a - bx_i \mid$$

El problema con el signo es el siguiente: la fuerza es un vector con una dirección y la suponemos que siempre se dirige hacia la línea. Así, si un punto está encima de la línea, la tensión va hacia abajo y se está abajo de ella, la tensión va hacia arriba. El objetivo de la regresión lineal pr el método de los resortes es minimizar la tensión global T. Aunque este problema sea más laborioso que el de los hilos, también puede resolverse por ensayo y error o uno podría ayudarse con un algoritmo genético, que simula la evolución biológica con el ánimo de resolver problemas de matemáticas.

El problema con los métodos del hilo y de los resortes es que involucran al valor absoluto y signos que indican dirección. Para hacer una especificación del valor absoluto hay que ver si se trata de un número positivo y dejarlo igual o si de uno negativo y cambiarle el signo. Todo eso es muy engorroso. Existe un truco que siendo simple permite liberarse de los inconvenientes del valor absoluto y es tomar los cuadrados en vez del valor absoluto. Físicamente, éso corresponde a guiarse por la energía potencial en el sistema de resortes y no por la fuerza o por la cantidad de hilo gastada. Formalzando:

El método de regresión de los mínimos cuadrados asociado a un conjunto de n puntos de la forma(x_i, y_i) produce una función E que mide el error cuadrático dado por

$$E = \sum_i^n (y_i - a - bx_i)^2$$

y su objetivo es hallar la pareja (a, b) que minimice dicha función. A la línea correspondiente $y = ax+b$ se la llama **línea de regresión por el método de mínimos cuadrados** . A los matemáticos les gusta el método del error cuadrático más que el de los resortes y el del hilo porque la función E puede minimizarse por derivadas. Para hallar un máximo o mínimo local de una función derivable definida en un intervalo abierto, sin los extremos, lo primero que hay que hacer es derivar e igualar a cero, pues la derivada da la pendiente de la línea tangente al punto. Cuando uno tiene un máximo o un mínimo, la línea tangente es horizontal:

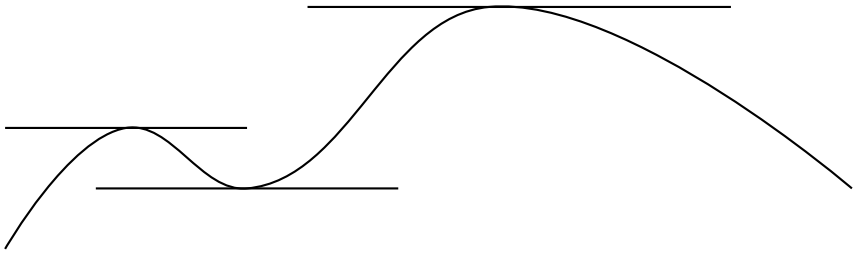

Figura 5.3. Sobre una cruva suave, los máximos y mínimos que no estén en los extremos de la curva, se hallan derivando e igualando a cero, pues la derivada da la pendiente de la línea tangente y sobre un máximo o mínimo dicha línea es horizontal, con pendiente cero.

Cuando uno tiene una función de una variable, la derivada es una derivada ordinaria:

$f(x) = 3x^2 + 5x + 7$
$f'(x) = 6x + 5$

Pero cuando uno tiene una función de dos variables, la derivada es una derivada parcial que significa que estamos no en una curva sino en una montaña en un mundo de tres dimensiones y ya no se toman línea tangetes horizontales sino planos tangentes horizontales.

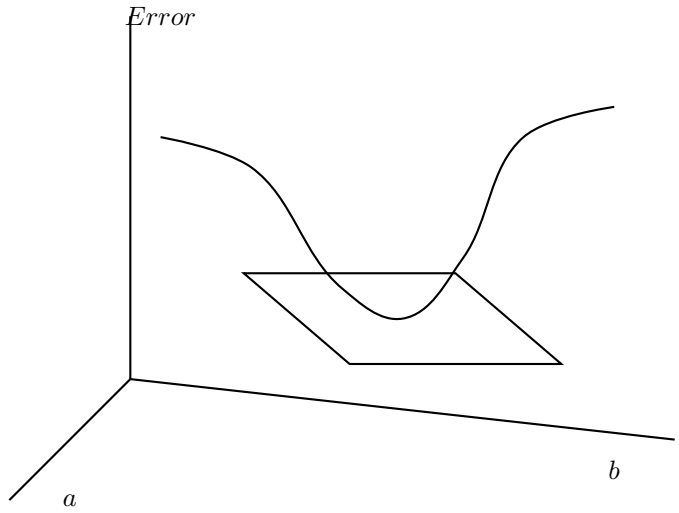

Figura 5.4. La función Error depende de dos variables (a, b). Su gráfica es como un valle en un espacio tridimensional. Para hallar los mínimos hallamos los puntos donde el plano tangente sea horizontal, lo cual implica que las derivadas parciales sean iguales a cero.

La derivada parcial de una función de varias variables con respecto a una de ella se calcula como si fuese una derivada ordinaria pero asumiendo que todas las demás variables tienen un valor fijo o constante. Ejemplo: si $f(x, y) = 3x^2y + 2xy - x^2 + y^3 - 8$ entonces la derivada parcial de f con respecto a x se nota $\partial f/\partial x$ y es

$\partial f/\partial x = 6xy + 2y - 2x + 0 + 0$

En efecto, la derivada parcial con respecto a x de $3x^2y$ es $3y(2x) = 6yx = 6xy$ pues $3y$ se toma como constante. De igual modo, la derivada parcial con respecto a x de y^3 da cero, lo mismo que la de 8, pues la derivada de una constante es cero.

Similarmente, si $f(a, b) = (3a + 4b)^2$

$\partial f/\partial a = 2(3a + 4b)(3)$

pues hay que tener en cuenta la derivada interna que con respecto a a es 3.

Minimicemos ahora la función E por derivación parcial igualada a cero.

$E = \sum_i^n (y_i - a - bx_i)^2$

Derivemos con respecto a a:

$\partial E/\partial a = \sum_i^n 2(y_i - a - bx_i)(-1) = 2\sum_i^n (y_i - a - bx_i)(-1) = 2\sum_i^n (-y_i + a + bx_i) = 0$

dividiendo por dos y reorganizando:

$\sum_i^n -y_i + \sum_i^n a + b\sum_i^n x_i = 0$

$a\sum_i^n 1 + b\sum_i^n x_i = \sum_i^n y_i$

$na + b\sum_i^n x_i = \sum_i^n y_i$

Como $\bar{x} = (\sum_i^n x_i)/n$, y también $\bar{y} = (\sum_i^n y_i)/n$, entonces $(\sum_i^n x_i) = n\bar{x}$ y $(\sum_i^n y_i) = n\bar{y}$, y por tanto:

$na + nb\bar{x} = n\bar{y}$

Dividiendo por n

$a + b\bar{x} + = \bar{y}$

y despejando a obtenemos:

$a = \bar{y} - b\bar{x}$

Derivemos con respecto a b:

$\partial E/\partial b = \sum_i^n 2(y_i - a - bx_i)(-x_i) = 2\sum_i^n (y_i - a - bx_i)(-x_i) = 2\sum_i^n (-x_iy_i + ax_i + b(x_i)^2) = 0$

dividiendo por dos y reorganizando:

$\sum_i^n -x_iy_i + a\sum_i^n x_i + b\sum_i^n (x_i)^2 = 0$

$a\sum_i^n x_i + b\sum_i^n (x_i)^2 = \sum_i^n x_iy_i$

$an\bar{x} + b\sum_i^n (x_i)^2 = \sum_i^n x_iy_i$

$(\bar{y} - b\bar{x})n\bar{x} + b\sum_i^n (x_i)^2 = \sum_i^n x_iy_i$

$n\bar{x}\bar{y} - nb(\bar{x})^2 + b\sum_i^n (x_i)^2 = \sum_i^n x_iy_i$

$b[-n(\bar{x})^2 + \sum_i^n (x_i)^2] = \sum_i^n x_i y_i - n\bar{x}\bar{y}$

$b = \frac{\sum_i^n x_i y_i - n\bar{x}\bar{y}}{-n(\bar{x})^2 + \sum_i^n (x_i)^2} = \frac{\sum_i^n x_i y_i - n\bar{x}\bar{y}}{\sum_i^n (x_i)^2 - n(\bar{x})^2} = \frac{SS_{xy}}{SS_{xx}}$

donde

$SS_{xy} = \sum_i^n x_i y_i - n\bar{x}\bar{y}$
$SS_{xx} = \sum_i^n (x_i)^2 - n(\bar{x})^2$

Y esta es la historia de la línea de regresión por el método de mínimos cuadrados. Todo lo demás de este capítulo se edifica sobre esta misma metodología, la cua se aplica para una variable lo mismo que para muchas.

5.3. Test para la relación funcional

El modelo de regresión lineal $y = \alpha + \beta x + \epsilon$ asume una dependencia funcional de y con respecto a x. Por consiguiente, la hipótesis nula natural en un estudio de regresión es que y de hecho no depende de x. Eso quiere decir que cuando x cambia, la y parece no oír, es decir, cuando la x cambia, la y permanece como estaba, excepto los cambios debidos al ruido aleatorio. Eso implica que la y permanece constante, horizontal. Pero como hay ruido, la y irá una vez arriba y otra vez abajo de su promedio. Todo esto es equivalente a decir que la pendiente del modelo teórico es $\beta \neq 0$.

$H_o : \beta = 0$, (x no afecta linealmente a y. Aunque x cambie, y no cambia de forma consistentemente lineal).
$H_a : \beta \neq 0$, (y cambia proporcionalmente a los cambios en x)

Hay varios procedimientos para probar estas hipótesis. La necesidad de hacer la prueba se justifica porque el ruido puede causar que aparezcan relaciones espurias, que no vuelven a aparecer cuando se repite el experimento. ¿Pero si el ruido puede crear relaciones espurias, cómo podemos asegurarnos que una relación funcional existe? Lo podemos hacer porque el ruido tiende a autoaniquilarse: la tendencia sistemática tiende a verse más nítida entre más datos haya, y para un cierto volumen de datos, el ruido puede ser tolerable y no impedir que se vea el efecto de la **variable dependiente o experimental** sobre la **variable dependiente o respuesta**.

Aprendamos a usar una F, la que nos sirvió para comparar varianzas, para dilucidar si $\beta = 0$ o no.
La forma como nosotros sabremos que y depende de x es moviendo el valor de x y observando el efecto sufrido por y. Si y de forma consistente varía al variar la x, nosotros podremos concluir que una relación funcional existe, pero si la variación observada de y pudiese atribuírse al efecto del ruido, no habría necesidad de suponer entonces que una relación funcional entre x y y existe. La variación de y se da por:

La suma total de cuadrados (total Sum of Squares) = SS Total = $\sum (y_i - \bar{y})^2 = \sum y_i^2 - \frac{(\sum y_i)^2}{n}$

Ahora denotamos por $\widehat{y}$ el valor $\alpha + \beta x_o$ que es el valor de la línea en x_o. Observando que

$0 = -\widehat{y} + \widehat{y}$

podemos insertar ese cero en el SS total como sigue:

SS total= $\sum (y_i + 0 - \bar{y})^2 = \sum (y_i - \widehat{y} + \widehat{y} - \bar{y})^2 == \sum ((y_i - \widehat{y}) + (\widehat{y} - \bar{y}))^2$
$= \sum (y_i - \widehat{y})^2 + 2\sum (y_i - \widehat{y})(\widehat{y} - \bar{y}) + \sum (\widehat{y} - \bar{y})^2$

58 ***Ejercicio*** *Pruebe que siempre se tiene que:* $2\sum (y_i - \widehat{y})(\widehat{y} - \bar{y}) = 0$

Aplicando este resultado, obtenemos:

59 ◇ ***Teorema:*** *SS Total* $= \sum(y_i - \bar{y})^2 = \sum(y_i - \widehat{y})^2 + \sum(\widehat{y} - \bar{y})^2$.

Esta ecuación nos permite desentrañar lo que es información de lo que es ruido. Pero cada modelo tiene su definición específica de ruido. En nuestro caso, ruido es cualquier cosa que cause que un evento se desvíe del modelo de regresión lineal, es decir, el efecto del ruido coincide con el término $(y_i - \widehat{y})^2$. Por otra parte, $(\widehat{y} - \bar{y})^2$ representa la información pues dice que el modelo lineal se aparta de ser constante e igual al promedio de y. Recordemos que el promedio de y es el bastión de la H_o: la y no escucha los cambios de x, sino que permanece constante e igual a su promedio, aparte de los cambios debidos al ruido.

60 ◇ ***Teorema:*** *En nuestro modelo, la información y el ruido son ortogonales, i.e., el ruido no interfiere con la información y, por tanto, ambos conceptos están nítidamente definidos.*

Vemos que el término $\sum(\widehat{y} - \bar{y})^2$ es la variación atribuible a la dependencia lineal. Así que, la idea es que esta variación sea lo suficientemente grande como para ser considerada como relevante. ¿Pero, grande con respecto a que? Pues a la variabilidad causada por el ruido que se da por $\sum(y_i - \widehat{y})^2$. Para facilitar los cálculos, nosotros podemos usar el próximo resultado.

61 ***Definición y teorema*** *Si definimos SS de regresión mediante la identidad*

SS de regresión$= \sum(\widehat{y} - \bar{y})^2$*, lo que se aparta consistentemente del promedio.*

y si definimos el SS del ruido como

SS del ruido $= \sum(y_i - \widehat{y})^2$*, lo que se aparta de la línea de regresión,*

entonces

a) SS Total$= \sum y_i^2 - \frac{(\sum y_i)^2}{n}$

b) SS de regresión $= \frac{(\sum xy - \frac{\sum x \sum y}{n})^2}{\sum x^2 - \frac{(\sum x)^2}{n}}$

c) SS del ruido= SS total - SS de regresión.

Bajo la hipótesis nula, la variable independiente x no influye para nada sobre la variable dependiente y, sino que toda posible variación es nada más que un efecto de los factores no controlados, que genéricamente se denomina azar o ruido y que se modelan por la v.a. ϵ que tiene media 0 (pues es azar y por tanto no es ni fu ni fa) pero varianza σ^2. Por consiguiente, bajo la hipótesis nula, la única fuente de variación con respecto a la media global es el azar. Esto nos permite decir que tenemos dos tipos de variación, una registrada por *SS de regresión* y la otra por *SS de ruido*. Pero ambas, bajo la H_o, tienen una única fuente el ruido, σ^2. La gran idea es ahora transformar estas dos variaciones en varianzas. Curiosamente, todo lo que se necesita para obtener varianzas es dividir dichas variaciones por números adecuados, llamados grados de libertad.

Ahora, podemos definir un estadígrafo para comparar la información contra el ruido. Definimos por tanto:

62 ♣ ***Definición.*** $R_{exp} = \frac{\frac{SS\ de\ Regresión}{1}}{\frac{SS\ de\ ruido}{n-2}}$

Debemos ahora predecir lo que se espera que valga R_{exp} bajo la H_o. Cuando no hay más que azar, no hay otras fuentes de variación. Por tanto, las varianzas que entran en la R_{exp} deben estar ambas relacionadas con el azar. Se puede demostrar que dichas varianzas son ambas estimadores (insesgados) de la varianza del azar, σ^2. Por tanto, bajo la H_o, y si hubiese un número infinito de datos, el valor esperado o promedio de R_{exp} sería 1. Podemos ahora comparar entonces lo que se ve, R_{exp} con lo que se cree, que $R = 1$.

63 ◇ ***Teorema:*** *Cuando la hipótesis nula, $\beta = 0$, es correcta el estadígrafo*

$$F_{exp} = \frac{R_{exp}}{R} = \frac{R_{exp}}{1} = \frac{\frac{SS\ de\ Regresión}{1}}{\frac{SS\ de\ ruido}{n-2}}$$

que relaciona lo que se ve con lo que se cree bajo la Ho, se distribuye como una F con 1 g.l. para el numerador y $n-2$ g.l. para el denominador. Si los grados de libertad son infinitos en número, el estadígrafo F_{exp} toma bajo la H_o el valor uno. Por lo tanto, la región de aceptación de la H_o contendrá el UNO. Por otro lado, cuando la variable independiene x afecta la variable dependiente y, su efecto se mide a través de SS de regresión: entre más pronunciado sea el efecto de x sobre y, más grande será este término y la F_{exp} será significativamente más grande que UNO. Por ésa razón, la H_o se rechaza con una cola, la cola superior.

64 ***Ejemplo*** *Imaginemos que dos pueblos hermanos forman colonias separadas. Sus formas de hablar divergirán e iran formando lenguas diferentes. La diferencia se cuantifica por una distancia: se le da un texto de un lengua a un representante de la otra población y se le examina sobre el grado de comprensión obtenido. Entre menos comprensión, más distancia. Los siguientes pares de datos (1,1)(2,2)(3,1)(4,2)(5,3)(6,2) (7,2) (8,2) indican el tiempo, en centurias, y la distancia entre dos lenguas. Si las dos lenguas van cada una por su lado, la distancia entre ellas crecería con el tiempo pero si las dos lenguas están ligadas debido a una estrecha relación entre sus hablantes, es posible que ellas evolucionen pero de forma sincronizada y la distancia no aumentará, sino que cambiará erráticamente alrededor de un promedio. Nuestra H_o es que con el tiempo no habrá separación: $\beta = 0$. Decidamos si esto es verdad o no.*

Solución: Hagamos primero el diagrama de dispersión:

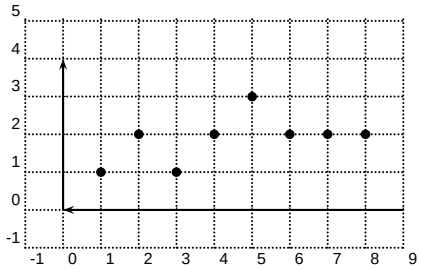

Figure 5.5. La línea recta horizontal que mejor representa la H_o es, estimada visualmente, $y = 2$. Como esa línea es horizontal, la H_o se acepta, la pendiente β es cero.

Ahora construyamos la tabla de regresión:

Regresión de distancia de separación y vs tiempo x					
	x	y	xy	x^2	y^2
	1	1	1	1	1
	2	2	4	4	4
	3	1	3	9	1
	4	2	8	16	4
	5	3	15	25	9
	6	2	12	36	4
	7	2	14	49	4
	8	2	16	64	4
Sumas	$\sum x$ =36	$\sum y$ =15	$\sum xy$ =73	$\sum x^2$ = 204	$\sum y^2$ =31

Las sumas de cuadrados son:
SS Total= $\sum y^2 - \frac{(\sum y_i)^2}{n} = 31 - \frac{(15)^2}{8} = 2{,}875$
SS de regresión= $\frac{(\sum xy - \frac{\sum x \sum y}{n})^2}{\sum x^2 - \frac{(\sum x)^2}{n}} = \frac{73 - \frac{(36)(15)}{8}}{204 - \frac{(36)^2}{8}} = \frac{30{,}25}{42} = 0{,}7202$
SS de ruido= SS Total - SS de regresión= 2.875-0.7202= 2.1548.
Los grados de libertad: para regresión, 1; para ruido, 8-2=6.

$F_{exp} = \frac{\frac{SS\ de\ regresión}{1}}{\frac{SS\ de\ ruido}{n-2}} = \frac{\frac{0,7202}{1}}{\frac{2,1548}{6}} = \frac{0,7202}{0,3591} = 2,005$

Para $\alpha = 0,02$ y dos colas, el valor crítico de F con 1 y 6 g.l. es 13.7. Nuestro estadígrafo dio 2.005, lo cual es pequeño. Nosotros concluimos que aquí no tenemos datos suficientes para pretender que hay una regresión lineal justificada.

Veredicto: Las lenguas estudiadas co-evolucionan, van como dos hermanitas cogidas de la mano a pesar de que son un poco distintas, lo cual se sabe pues la línea de regreión no es el eje X sino que queda un poco por encima.

Para entender mejor el veredicto, calculemos la línea de regresión de mínimos cuadrados
Las medias son: $\bar{x} = 4,5$ y $\bar{y} = 1,875$ con $n = 8$. Por consiguiente, la pendiente asociada a la muestra es

$b = \frac{\sum xy - n\bar{x}\bar{y}}{\sum x^2 - n\bar{x}^2}$
$b = \frac{73-8(4,5)(1,875)}{204-8(4,5)^2} = 0,044$
$a = \bar{y} - b\bar{x}$
$a = 1,875 - (0,044)(4,5) = 1,677$

Así, la línea de regresión de mínimos cuadrados es
$y = 1,677 + 0,044x$
Por otra parte, el valor promedio de y es 1.875. Nuestro veredicto dice que la mejor manera de explicar los datos de la muestra es diciendo que la distancia entre las lenguas es de 1.875, la cual es constante a través del tiempo, pero debido a que el análisis literario de la muestra no cubrió todos las facetas del lenguaje a toda hora, por puro azar se generó la impresión de que había una pequeña divergencia dada por $b = 0,044$. También hubiese sido posible que una muestra hubiese generado la impresión de que las lenguas se acercan en lugar de divergir. Una pendiente negativa hubiese resultado si quitásemos los puntos (1,1) y (5,3). O sin quitar nada pero aumentando el punto (9,1).

65 ***Interpretación general del veredicto***

Tenemos en general que cuando se acepta la hipótesis nula de que la pendiente de la línea de regresión es horizontal, lo que estamos diciendo es que al repetir muchas veces el experimento podemos esperar que la nueva pendiente sea unas veces positiva y otras negativa. Es decir, no tenemos poder predictivo en cuanto al signo de la pendiente. Pero cuando se rechaza la hipótesis nula con una cola y se infiere que, por ejemplo, la pendiente es positiva, entonces tenemos base para decir que al repetir muchas veces el experimento, la pendiente será una muy buena proporción de veces positiva. Acá tenemos poder predictivo.

5.4. Predicciones

La primera utilidad de la línea de la regresión está en la predicción del valor que tomará la función para un valor determinado. Pero atención, como los datos se basan en una muestra que tiene varianza de ruido no nula, las predicciones no podrán ser exactas sino que tendrán un margen de error dado por un intervalo de confianza.

66 ***Ejemplo*** *Si un modelo de mínimos cuadrados predice una relación causa (x) -efecto (y) dada por y:*
$y = -0,27 + 2,24x$
y deseamos saber el valor de la respuesta para $x = 2,5$*, entonces nuestra predicción dice que:*
$y(2,5) = -0,27 + 2,24(2,5) = -0,27 + 5,6 = 5,33$

67 ***El error estándar***

Predecir con base en un modelo lineal es algo que debe hacerse con cuidado: si el modelo del ejemplo anterior cuantifica la relación entre la cantidad de vitaminas ingeridas y la fortaleza del sistema inmunológico, entonces hay que tener presente que algunas vitaminas pueden ser tóxicas si se ingieren en gran cantidad. Eso quiere decir que las extrapolaciones carecen de autoridad entre más lejos estén del rango de datos.

El sentimiento de confiabilidad de una predicción se cuantifica mediante el **error estándar** de la predicción, el cual se nota s_y, el cual se define mediante el siguiente grupo de fórmulas:

$\bar{x} = \sum x/n.$
$\bar{y} = \sum y/n.$
$SS_{xx} = \sum x^2 - n\bar{x}^2$
$SS_{xy} = \sum xy - n\bar{x}\bar{y};$
$SS_{yy} = \sum y^2 - n\bar{y}^2;$
$b = \frac{SS_{xy}}{SS_{xx}};$
$SSE = SS_{yy} - bSS_{xy}$
$s = \sqrt{SSE/(n-2)}$
$c = \sqrt{1 + \frac{1}{n} + \frac{(x_o - \bar{x})^2}{SS_{xx}}}$
$s_y = sc$

Supongamos ahora que nosotros hemos escogido una significancia α. Entonces, el valor predicho de y para un x_o dado se da por el intervalo de confianza:

$$[(a + bx_o) - t_{\alpha/2}s_y, (a + bx_o) + t_{\alpha/2}s_y]$$

donde t se toma con n-2 grados de libertad. Nosotros perdemos 2 grados de libertad porque nosotros calculamos a y b del conjunto de datos.

68 ***Ejemplo*** *Calculemos el valor predicho de y para* $x_o = 10$ *y* $\alpha = 0{,}05$ *si hay* $n = 6$ *pares de datos con*

$\sum x$ =*19,* $\sum y$ =*41,* $\sum xy$ =*145,* $\sum x^2$ = *67,* $\sum y^2$ =*317*

Solución: ejecutamos las fórmulas:

$\bar{x} = \sum x/6 = 19/6 = 3{,}16.$
$\bar{y} = \sum y/6 = 41/6 = 6{,}83.$
$SS_{xx} = \sum x^2 - n\bar{x}^2 = 67 - 6(3{,}16^2) = 6{,}7$
$SS_{xy} = \sum xy - n\bar{x}\bar{y} = 145 - 6(3{,}16)(6{,}83) = 15{,}50;$
$SS_{yy} = \sum y^2 - n\bar{y}^2 = 317 - 6(6{,}83)^2 = 37{,}10;$
$b = \frac{SS_{xy}}{SS_{xx}} = 15{,}50/6{,}7 = 2{,}31;$
$a = \bar{y} - b\bar{x} = 6{,}83 - (2{,}31)(3{,}16) = -0{,}47$
$SSE = SS_{yy} - bSS_{xy} = 37{,}10 - 2{,}31(15{,}50) = 1{,}295$
$s = \sqrt{SSE/(n-2)} = \sqrt{1{,}295/4} = 0{,}57$
$c = \sqrt{1 + \frac{1}{n} + \frac{(x_o - \bar{x})^2}{SS_{xx}}}$
$c = \sqrt{1 + \frac{1}{6} + \frac{(10-3{,}16)^2}{6{,}7}} = \sqrt{1 + 0{,}167 + \frac{(6{,}83)^2}{6{,}7}}$
$= \sqrt{1 + 0{,}167 + \frac{46{,}6}{6{,}7}} = = \sqrt{1{,}167 + 6{,}96} = 2{,}85$
$s_y = sc$
$s_y = 0{,}57(2{,}85) = 1{,}62$

En consecuencia, para 4 g.l., $\alpha = 0{,}05$ y dos colas, $t = 2{,}77$, por lo que el intervalo de confianza para la predicción de y para $x_o = 10$ es:

$[(a + bx_o) - t_{\alpha/2}s_y, (a + bx_o) + t_{\alpha/2}s_y]$
(-0.47+2.31(10) - 2.77(1.62), -0.47+2.31(10)+ 2.77(1.62)
(-0.47+23.1 -4.48, -0.47+23.1 +4.48) = (18.15, 27.11)

Observemos que los intervalos son muy grandes, lo cual denota que hacer en regresión predicciones precisas es algo muy costoso en términos del número de datos a reunir. Conviene tener en cuenta que

la respuesta definitiva es sensible a las aproximaciones que uno haya utilizado a lo largo de los cálculos. Que sea ese otro motivo para recordar que *Excel*, *Gnumeric* y *R* pueden hacer cálculos con una precisión del orden de 10 cifras decimales.

El intervalo de confianza también es útil para probar una hipótesis nula sobre una predicción. Digamos, nosotros no podemos exigir, con un significancia de 0.05, que para $x_o = 10$ el valor de y pueda ser 35 o 10.

69 ***Ejercicio*** *Invente, invente, invente muchas telenovelas y deles un buen final.*

5.5. Correlación y covarianza

La regresión asume que hay una relación causa-efecto entre dos variables, a la causa se llama indepediente y al efecto dependiente. Con todo, también puede encontrarse que dos variables varíen de manera conjunta, que ambas aumentan, que ambas disminuyan o que mientras la una aumente, la otra disminuya y sin embargo uno no pueda decir que hay una relación causa-efecto. Estos casos se estudian mejor por la correlación y la covarianza.

70 ***Definición*** *Sean X, Y dos variables aleatorias que se registran conjuntamente como valores (x_i, y_i). La covarianza entre X y Y se define como:*

$$COV(X;Y) = \sum_i^n = 1(x_i - \bar{x})(y_i - \bar{y})$$

¿Qué mide la covarianza? Ella mide para cada (x_i, y_i) las desviaciones de cada variable con respecto a su media y si dichas desviaciones tienen ambas el mismo signo, produce un aporte positivo a la covarianza total, pero si tienen signo diferente, produce un aporte negativo. Si hay consistencia entre todos los valores, entonces, una covarianza positiva dice que ambas variables varían con igual signo con respecto a su media, es decir, ambas se desvían hacia arriba de sus medias o ambas hacia abajos. Por tanto, la gráfica de la relación (x_i, y_i) tenderá a aglomerarse sobre una función creciente.

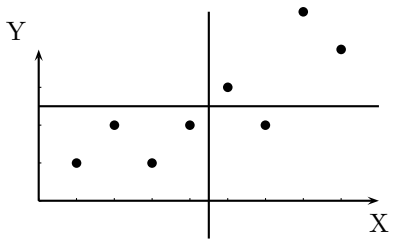

Figure 5.6. La media de X está en 4.5, la de Y en 2.5. Se observa la siguiente tendencia: cuando X sube de su media, Y también lo hace. Si X baja de su media, Y hace lo mismo con la suya: la covarianza es positiva y los datos se ajustan a una función creciente.

Pero si la covarianza tiene signo negativo, éso significa que hay una dominancia de los puntos en que las desviaciones de sus variables con respecto a sus medias tienen signo contrario: la una crece pero la otra decrece. En ése caso la gráfica de la relación será una función decreciente:

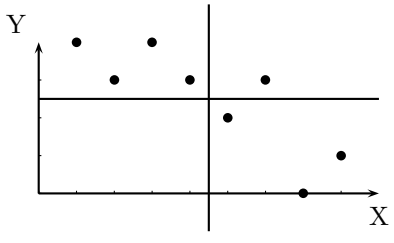

Figure 5.7. La media de X está en 4.5, la de Y en 2.5. Se observa la siguiente tendencia: cuando X sube de su media, Y baja de la suya. Si X baja de su media, Y sube de la suya: la covarianza es negativa y los datos se ajustan a una función decreciente.

Una covarianza cercana a cero indica que no hay una tendencia definida y se tienen abundantes puntos en todos los cuatro cuadrantes generados por las medias ocmo ejes.

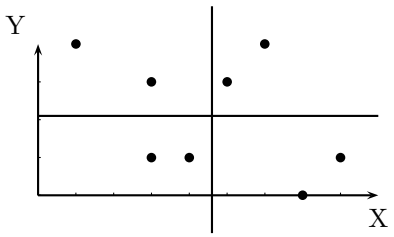

Figure 5.8. La media de X está en 4.6, la de Y en 2.1. No se observa una tendencia dominante, cada quien se mueve por su lado. La covarianza es cercana a cero.

Ejemplos: si uno mide el peso y la altura de los niños, uno encontrará una covarianza positiva. Si uno mide la altura de los niños menores de cinco años y el tiempo que pasan durmiendo, uno encuentra una covariana negativa. Si uno mide la altura de los niños de la misma edad y su destreza para las matemáticas, uno encontrará una covarianza cercana a cero.

El siguiente es un truco que nos permite comparar covarianzas de diversos grupos de datos:

71 ***Definición y teorema.*** *Sean X, Y dos variables aleatorias que se registran conjuntamente como valores (x_i, y_i). La correlación r entre X y Y se define como:*

$r = \frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum(x_i-\bar{x})^2 \sum(y_i-\bar{y})^2}} = \sqrt{\frac{a\sum y+b\sum xy-n\bar{y}^2}{\sum y^2-n\bar{y}^2}}$*, pero con su signo.*

A r^2 se le llama coeficiente de determinación. La correlación toma valores entre -1 y 1. Es -1 cuando la covarianza es negativa y los puntos de la gráfica se ajusta perfectamente a una línea decreciente. Es +1 cuando la correlación es positiva y los puntos de la gráfica se ajustan perfectamente a una línea creciente. Una correlación cercana a cero indica que los puntos están desperdigados sin un desequilibrio consistente.

El azar puede crear correlaciones que no existen, es decir, que cuando se repite el experimento o el muestro, ya no aparecen. Para filtrar el efecto del azar formulamos

$H_o : \rho = 0$ donde ρ es la correlación poblacional en tanto que r es la correlación de la muestra.

La H_o puede rechazarse con una o dos colas y para ello se usa una t:

$s_r = \sqrt{\frac{1-r^2}{n-2}}$; $\quad t = \frac{r}{s_r}$, con n-2 g.l.

Dado un r, para probar $H_o : \rho = \rho_o$ se usa una z como sigue:

$\omega = \omega(r) = 0{,}5ln(\frac{1+r}{1-r})$; $\quad z = \frac{\omega(r)-\omega(\rho_o)}{\sigma_\rho}$

donde $\sigma_\rho = \sqrt{\frac{1}{n-3}}$

El intervalo de confianza para $\omega(\rho)$ es $\omega(r) \pm 1{,}96\sigma_\rho$. El IC de r se halla a partir del de ω con $r = \frac{e^{2\omega}-1}{e^{2\omega+1}}$.

5.6. Resumen de fórmulas

Regresión (causa-efecto) y correlación (no necesariamente causa y efecto) con n pares de datos.

I. Regresión. Causa $=x$, efecto $= y$. Modelo: $y = \alpha + \beta x + \epsilon$; ϵ es el ruido.

1. Hacer dibujo y tabla con x, y, xy, x^2, y^2. Calcular $\bar{x}$ y $\bar{y}$; $SS_{xx} = \sum x^2 - n\bar{x}^2$; $SS_{xy} = \sum xy - n\bar{x}\bar{y}$; $SS_{yy} = \sum y^2 - n\bar{y}^2$; $SSE = SS_{yy} - bSS_{xy}$

2. Línea de regresión (min cuadrados): $y = a + bx$; $b = \frac{SS_{xy}}{SS_{xx}}$; $a = \bar{y} - b\bar{x}$.

3. Test t para $\beta = 0$. Ponga $\beta_o = 0$ en 4. Se toman $n-2$ g.l.

4. Test t para $\beta = \beta_o$: $s = \sqrt{SSE/(n-2)}$; $s_b = s/\sqrt{SS_{xx}}$; $t = \frac{b-\beta_o}{s_b}$

5. Intervalo de confianza para β: $(b - ts_\beta, b + ts_\beta)$ con $n-2$ g.l.

6. Anova para $\beta = 0$: $TotalSS = \sum y_i^2 - \frac{(\sum y_i)^2}{n}$; $RegreSS = \frac{(S_{xy})^2}{S_{xx}}$; $RuidoSS = totalSS - regreSS$; $Fisher = \frac{\frac{RegreSS}{1}}{\frac{RuidoSS}{n-2}}$ con 1 y $n-2$ g.l.

7. Prediccion de y dado x_o: $\widehat{y} = a + bx_o$.

8. IC para la predicción de y dado x_o: $\widehat{y} \pm t_{\alpha/2} s_y$ con $s_y = s\sqrt{1 + \frac{1}{n} + \frac{(x_o - \bar{x})^2}{SS_{xx}}}$

9. IC para la predicción del valor medio de y dado x_o: $\widehat{y} \pm t_{\alpha/2} s\sqrt{\frac{1}{n} + \frac{(x_o - \bar{x})^2}{SS_{xx}}}$

10. Dado x_o, $H_o : y = y_o$. Se usa $t = \frac{a + bx_o - y_o}{s_y}$ con n-2 g.l con s_y de 8.

11. IC para α dado a: se pone $x_o = 0$ en 8.

12. Dado a, test $H_o : \alpha = \alpha_o$: $t = \frac{a - \alpha_o}{s_y}$ con $n-2$ g.l. con s_y de 8.

13. Para comparar dos pendientes: $ruido = \sum y^2 - \frac{(\sum xy)^2}{\sum x^2}$

$s_p^2 = \frac{ruido_1 + ruido_2}{n_1 + n_2 - 4}$; $s_{(b_1 - b_2)} = \sqrt{\frac{s_p^2}{(\sum x^2)_1} + \frac{s_p^2}{(\sum x^2)_2}}$

$t = \frac{b_1 - b_2}{s_{(b_1 - b_2)}}$ con $n_1 + n_2 - 4$ g.l.

14.Si se sabe que $b_1 = b_2$, todos los datos se reunen y dan una sola $b = b_c$:

$b_c = \frac{(\sum x^2)_1 b_1 + (\sum x^2)_2 b_2}{(\sum x^2)_1 + (\sum x^2)_2}$

15. Para comparar 2 predicciones de 2 modelos se usa el s_p^2 de 13:

$y_1 = a_1 + b_1 x_0$, $y_2 = a_2 + b_2 x_0$,

$s_{(y_1 - y_2)} = \sqrt{s_p^2 [\frac{1}{n_1} + \frac{1}{n_2} + \frac{(x_o - \bar{X}_1)^2}{(\sum x^2)_1} + \frac{(x_o - \bar{X}_2)^2}{(\sum x^2)_2}]}$

$t = \frac{y_1 - y_2}{s_{(y_1 - y_2)}}$; g.l$= (n_1 - 2) + (n_2 - 2) = n_1 + n_2 - 4$.

II. Correlación. Se mide con r, coeficiente de correlación, y con r^2, coeficiente de determinación.

14. $r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} = \sqrt{\frac{a \sum y + b \sum xy - n\bar{y}^2}{\sum y^2 - n\bar{y}^2}}$, pero con su signo.

16. Dado un r, para probar $H_o : \rho =$ correlación poblacional $= 0$:

$s_r = \sqrt{\frac{1 - r^2}{n-2}}$; $t = \frac{r}{s_r}$, con n-2 g.l.

17. Dado un r, para probar $H_o : \rho =$ correlación poblacional $= \rho_o$:

$\omega = \omega(r) = 0{,}5 ln(\frac{1+r}{1-r})$; $z = \frac{\omega(r) - \omega(\rho_o)}{\sigma_\rho}$ donde $\sigma_\rho = \sqrt{\frac{1}{n-3}}$

18. El IC para $\omega(\rho)$ es $\omega(r) \pm 1{,}96\sigma_\rho$. El IC de r se halla a partir del de ω con $r = \frac{e^{2\omega} - 1}{e^{2\omega + 1}}$.

72 *Verificación de supuestos*

El modelo de regresión lineal univariada que hemos visto tiene los siguientes supuestos:

1. Existe una clara división entre la variable que se controla o independiente x, y las demás variables que pueden influir sobre la variable respuesta o dependiente.

2. No hay error en el proceso de control.

3. De no ser por las variables no controladas, el sistema siempre respondería lo mismo ante los mismos valores del estímulo x.

4. Toda fuente de incertidumbre se debe a las variables no controladas, cuyo efecto se ve como una v.a. normalmente distribuida con media cero y varianza σ^2 que se superpone al valor promedio de la variable respuesta. Se asume que dicha varianza no depende del valor de x ni tampoco de ningún otro factor externo.

¿Qué tanto peso debe concedérsele a cada uno de estos requisitos? Eso puede depender del experimento en particular. En general, cuando se habla de rigor se tiene en mente la naturaleza del ruido, si se distribuye normalmente y si su varianza es independiente de x. Y como puede pasar que el rigor exiga abandonar el modelo, entonces uno quisiera saber qué remedio existe que esté por ahí a la mano.

Los estudiosos han observado que lo que más problemas da es el conjunto de outliers que corresponden a puntos que quedan demasiado lejos de la línea de regresión. Esos outliers hacen que las colas de la distribución del ruido o azar queden mas gordas de lo que debieran. La sugerencia es quitarlos, primero los más alejados e ir cogiendo los más cercanos hasta que ya se pueda aceptar la normalidad. Los outliers más peligrosos son los que están al mismo tiempo alejados de la línea y en un extremo de ella pues un dato así tira el resto de la distribución hacia su lado. Este proceso de maquillaje se justifica alegando, por ejemplo, que hay variables no controldas que inciden demasiado sobre la variable respuesta que se mide. Por supuesto, el objetivo siguiente debería ser tratar de dilucidar dichas variables.

R tiene muchas facilidades para tratar con outliers, las más corrientes queda bien verlas en el taller que sigue a continuación. A las desviaciones de la línea $y_i - \hat{y}_i$, se llaman oficialmente **residuos**: los residuos deben estar normalmente distribuidos, con media cero (no se chequea porque es cierto por construcción) y varianza que no depende de x. Para poder chequear la homogeniedad de varianzas, uno debe tener por lo menos dos datos de la **variable respuesta o dependiente** por cada dato de la **variable independiente o estímulo o experimental**.

5.7. Trabajando con Gnumeric

Debido a la variada naturaleza de las preguntas que uno se puede hacer en relación a la regresión y a la correlación, *Gnumeric* es un ayudante muy apropiado. Para ello se teclean los datos en sus dos columnas respectivas y a su lado se insertan columnas con los valores xy, x^2 , y^2. A partir de la sumas uno puede calcular las sumas de cuadrados que miden las variaciones y aplicarlas en las fórmulas adecuadas.

5.8. Taller en R sobre regresión lineal

La regresión lineal univariada se estudia cómodamente desde la GUI. Para ello se editan los datos, se activan, y sobre ellos se llama la siguiente secuencia de menús: *Statistics + Fit models + Linear regresion*. En respuesta se despliegará un diálogo en el cual uno puede seleccionar cuál vector de datos irá como variable dependiente y cuál como independiente. Para hacer gráficas uno mira aquí y allá hasta encontrar un menu que esté activado y que tenga que ver con ellas. Es inteligente mirar las gráficas y las tablas de salida a ver si es correcta la designación de cuál variable es independiente o explicativa y cuál es dependiente o de respuesta. Si la asignación hecha es incorrecta, es suficiente escribir, por ejemplo, *sueldoB sueldoA* en vez de *sueldoA sueldoB*.

Para estudiar la hipótesis de dependencia lineal entre dos vectores de datos usando la consola:

```
#==============================
#REGRESION LINEAL UNIVARIADA ENTRE DOS VECTORES
#Correr por partes
#
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(5,5), pch=16)
#
#Parte 1
semana <- c(1,2,3,4,5,6,7)
```

```
precio <- c(2,4,5,6,7,8,9)
#El símbolo ~ indica que se estudia una relación entre
#la variable dependiente a la izquierda y las independientes
#a la derecha.
#La regresión se estudia por medio de
#los modelos lineales (linear models).
#y es el nombre del modelo a estudiar.
y<-lm(precio ~ semana)
#Liste los coeficientes de regresión
y$coefficients
#Publica todos los resultados
summary(y)
#Dibujar modelo)
library(car) # cargar paquete que dibuja
#Meter los dos vectores en una tabla
carestia <- data.frame(semana, precio)
#cargar paquete
library(car)
#Dibujar el modelo
scatterplot(precio ~ semana, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = carestia)
#
#Parte 2
# ESTUDIO DE SUPUESTOS
#Paquete diagnóstico de residuos de regresión
#La distancia de Cook mide la peligrosidad de los
#residuos: entre más grande sea, más peligrosos:
plot(y)
#
#Parte 3
#Listado de residuos
y$residuals
y$fitted.values
#Dibujar residuos
y1 = min(y$residuals - 3)
y2 =max(y$residuals + 3)
boxplot(y$residuals, ylim = c(y1,y2), ylab="residuals", pch=19)
#
#Parte 4
scatterplot(y$residuals ~ precio, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = carestia)
#Chequear normalidad de los residuos
#Pegar los valores esperados y los residuos a la tabla
carestia <- cbind(carestia, y$fitted.values, y$residuals)
library(car)
#
#Parte 5
qq.plot(y$residuals, dist= "norm", col=palette()[1],
         ylab="Residual Quantiles", main="Normal Probability Plot", pch=19)
#Las varianzas deben ser independientes de x:
bartlett.test(y$residuals ~ semana)
```

5.9. Regresión doble

Hemos estudiado un modelo de causa-efecto en el cual la causa puede modelarse por una variable unidimensional. La generalización inmediata es pasar de líneas de regresión a planos de regresión. Una vez hecha esta generalización, el paso a cualquier número de dimensiones es más de lo mismo.

Nuestro modelo es:

$y = output = a + b_1 x_1 + b_2 x_2 + \epsilon$

donde los x_i son las dos variables cuantitativas que inciden sobre el resultado. El ruido se describe por ϵ, que se aume como v.a. normalmente distribuida con media cero y desviación σ, la misma en cualquier parte del input.

Tenemos 3 variables a, b_1, b_2, por lo que necestamos 3 ecuaciones independientes. Supongamos que los datos vienen en forma de tabla con n renglones. Después de formular la función error y minimizarla con derivada parciales, terminamos con las 3 ecuaciones siguientes:

$\sum y = na + b_1 \sum x_1 + b_2 \sum x_2$
$\sum x_1 y = a \sum x_1 + b_1 \sum x_1^2 + b_2 \sum x_1 x_2$
$\sum x_2 y = a \sum x_2 + b_1 \sum x_1 x_2 + b_2 \sum x_2^2$

De estas ecuaciones, uno resuelve las incógnitas e inmediatamente uno puede hacer predicciones:

$Valor\ predicho\ de\ y\ para\ valores\ de\ x_1\ y\ x_2 = \widehat{y} = a + b_1 x1 + b_2 x_2$

Nuestras predicciones tienen un error debido a σ. El estadígrafo que nos ayuda a cuantificar dicha incertidumbre es s_y, el error estándard de la estimación o predicción, donde el 2 es el número de variables independientes y n es el número de puntos de la forma (x_1, x_2, y):

$s_y = \sqrt{\frac{\sum (y_i - \widehat{y_i})^2}{n-2-1}}$

Esto nos sirve para calcular el intervalo de confianza de la predicción para un par de x_1 y x_2:

$((a + b_1 x_1 + b_2 x_2) - t_{\alpha/2} s_y, (a + b_1 x_1 + b_2 x_2) + t_{\alpha/2} s_y)$, con $n - k - 1$ g.l.

Decimos que un modelo lineal se justifica cuando podemos alegar que el modelo explica una proporción adecuada de la variabilidad de la variable respuesta. Equivalentemente, podemos estudiar la hipótesis nula:

$H_o : b_1 = b_2 = ... = b_n = 0$
H_a : al menos uno de los $b_i \neq 0$.

Para estudiar estas hipótesis, el análisis de varianza nos ilumina: la variación total se divide en dos partes: una explicada por el modelo lineal y la otra, la variabilidad residual, identificada con el ruido. El quebrado o razón entre estas dos variaciones, apropiadamente normalizadas por sus grados de libertad siguen una distribución F.

La variación total es

$TSS = \sum (y_i - \bar{y})^2 = \sum (y_i - \widehat{y_i})^2 + \sum (\widehat{y_i} - \bar{y})^2$

Definiendo $RSS = \sum (\widehat{y_i} - \bar{y})^2 =$ (suma de cuadrados explicada por le modelo de regresión (sum of squares explained by the regression model) y

$NSS = \sum(y_i - \widehat{y}_i)^2$ suma de cuadrados atribuible al ruido (sum of squares defined as noise), obtenemos:

$TSS = RSS + NSS.$

Ahora nosotros podemos calcular el estadígrafo de Fisher:

$F = \frac{RSS/2}{NSS/(n-2-1)}$

el cual sigue una distribución F con 2 g.l. en el numerador y $n-2-1$ g.l. en el denominador.

5.10. Regresión doble en Excel o CALC o Gnumeric

Veámos como se procede a usar una hoja de cálculo para calular una regresión doble.

Se comienza con una tabla con tres columas de la forma x_1, x_2, y. Dicha tabla se extiende con las siguientes columnas: $x_1y, x_2, y, x_1^2, x_2^2,\ y^2$ y se calculan las sumas sobre cada columna. Veámos como usamos esta tabla para calcular el modelo

$y = a + b_1x_1 + b_2x_2$

Este modelo tiene 3 incógnitas, así que necesitamos 3 ecuaciones independientes, las cuales se formulan así: sumando el modelo sobre todos los valores (x_1, x_2, y) obtenemos

$\sum y = \sum a + \sum b_1x_1 + \sum b_2x_2$

esta es nuestra primera ecuación:

Si multiplicamos la ecuación del modelo por x_1 a cada lado y después sumamos, obtenemos

$\sum x_1y = \sum ax_1 + \sum b_1x_1x_1 + \sum b_2x_1x_2$

Si hacemos lo mismo con x_2:

$\sum x_2y = \sum ax_2 + \sum b_1x_1x_2 + \sum b_2x_2x_2$

Ahora tenemos 3 ecuaciones de las cuales uno puede despejar las 3 incógnitas:

$$\begin{array}{l}\sum y = na + b_1\sum x_1 + b_2\sum x_2\\ \sum x_1y = a\sum x_1 + b_1\sum x_1^2 + b_2\sum x_1x_2\\ \sum x_2y = a\sum x_2 + b_1\sum x_1x_2 + b_2\sum x_2^2\end{array}$$

Notemos que cada coeficiente puede leerse directamente de las sumas de la tabla extendida. Después de ésto, uno pasa a la consulta de tablas.

El procedimiento puede extenderse a cualquier número de variables, situación que se denomina regresión múltiple y que discutiremos un poco a continuación.

5.11. Regresión multiple

En la regresión lineal simple uno tiene una variable independiente x, y una variable dependiente y y uno aspira a que y tenga una dependecia lineal de x: $y = ax + b + error$. En la regresión lineal múltiple

uno tiene varias variables explicativas o independientes y una variable dependiente o respuesta. El nuevo objetivo es explicar y linealmente por medio de las variables independientes:

$y = \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + error$

Si este modelo se ajusta a los datos, entonces uno evalúa cada coeficiente y también el error que se supone que es una v.a. aleatoria de distribución normal con media cero y varianza σ^2 que es independiente de todos los x_i.

Mientras que la gráfica de una regresión sencilla es una línea, la de la regresión múltiple es un plano si hay 2 variables independientes y es un hiperplano si dicho número es mayor que 2.

La H_o en una regresión múltiple es que todos los coeficientes valen cero, es decir que las variables independientes no influyen de manera sistemática sobre la variable respuesta:

$H_o : \beta_1 = \beta_2 = ... = \beta_n = 0$
H_a : at least one $\beta_i \neq 0$.

La discrepancia entre lo que se ve y lo que se cree en el modelo se mide por una F que divide la variación total en los aportes dados por cada una de las variables y el aporte del ruido. Estas variaciones se dividen por grados de libertad adecuados y se obtienen varianzas. Con más claridad, tenemos que la variación total es:

$TSS = \sum(y_i - \bar{y})^2 = \sum(y_i - \widehat{y}_i)^2 + \sum(\widehat{y}_i - \bar{y})^2$

Definiendo $RSS = \sum(\widehat{y}_i - \bar{y})^2$ = suma de cuadrados explicada por el modelo de regresión y

$NSS = \sum(y_i - \widehat{y}_i)^2$ suma de cuadrados atribuible al ruido. Se obtiene:

$TSS = RSS + NSS.$

Calculamos el estadígrafo de Fisher

$F = \frac{RSS/k}{NSS/(n-k-1)}$

que tiene una distribución F con k g.l. en el numerador y $n - k - 1$ en el denominador.

Las predicciones vienen con un valor promedio que está en el centro de un intervalo de confianza:
Si definimos s_y como el error estándard de la estimación de la predicción

$s_y = \sqrt{\frac{\sum(y_i - \widehat{y}_i)^2}{n-k-1}}$

donde k es el número de variables independientes y n es el número de puntos. El intervalo de confianza para la predicción es:
$((a + b_1 x_1 + b_2 x_2 + ...) - t_{\alpha/2} s_y, (a + b_1 x_1 + b_2 x_2 + ...) + t_{\alpha/2} s_y)$, with d.f=n-k-1.

La regresión múltiple se estudia cómodamente desde la *GUI* de *R*. De hecho, no hay diferencia conceptual importante entre una regresión lineal y múltiple o multivariada (muchas variables independientes y muchas variables dependientes) y por tanto *R* y su *GUI* evolucionarán hacia un único caso general cuya especialización sale del trabajo que se esté ejecutando. Para hacer regresión múltiple desde la consola uno puede adaptar el siguiente programa que muestra una regresión múltiple ideal basada en una simulación:

```
#==============================
#REGRESION MULTIPLE: SIMULACION
#Limpia la memoria
rm(list = ls())
```

```
#Modo gráfico
par(mfrow=c(3,2), pch=16)
x <- rep (c(1:5), each = 5)
y <- rep( c(1:5), times = 5)
z <- c(1:25)
#Generamos 25 observaciones al azar de la normal
w<-rnorm(25, mean = 0, sd = 0.1)
for(i in 1:25)
{ z[i] = 2*x[i] + 3* y[i] + w[i]}
tabla <- data.frame(x,y,z)
tabla
multipleReg  <- lm(z ~ x + y, data = tabla)
summary(multipleReg)
#Hacer dibujo
library(scatterplot3d)
barras <- scatterplot3d(x,y, z,  highlight.3d=TRUE,
type="h", main="3D Scatterplot")
barras$plane(multipleReg)
```

Con datos reales, una regresión lineal múltiple luce como sigue:

```
#==============================
#REGRESION MULTIPLE
#Correr por partes
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#
#Parte uno
#Cargar paquete
data(Angell, package = "car")
#Publicar  información sobre el paquete
#help("Angell")
#parte dos
#Listar datos
Angell
multReg  <- lm(moral~hetero+mobility, data=Angell)
summary(multReg)
#Hacer dibujo
library(scatterplot3d)
attach(Angell)
barras <- scatterplot3d(moral,hetero,mobility,   highlight.3d=TRUE,
type="h", main="3D Scatterplot")
barras$plane(multReg)
#
#Parte 2
#Diagramas de dispersión
pairs(Angell)
#Covarianzas
#Quitamos lo que no sea numérico
Angell2 <- data.frame(Angell$moral, Angell$hetero, Angell$mobility)
#Matriz de correlaciones
cor(Angell2)
anova(multReg)
```

En regresión múltiple uno parte de un modelo exhaustivo en el sentido en que todas las variables

explicativas aparecen en el modelo. Pero sin duda que habrá unas variables más importantes que otras. El peso de cada variable se averigua con un análisis de varianza, lo cual es ejecutado por la última instrucción del programa anterior.

El poder explicativo de las variables se mide por el *p-value*, entre más grande, el poder es menor. Entre más pequeño sea el *p-value* de una variable estímulo, más poder explicativo tiene, más clara es la dependencia de la variable de salida con respecto a dicha variable. Puede suceder que el papel de determinada variable no sea mayor cosa. Conviene quitar dicha variable del modelo. Eso puede hacerse cuando su *p-value* es mayor que la significancia escogida, que puede ser el 0.05.

Existe un paquete para *R* que ayuda en la simplificación de modelos: se llama *leaps* y hay que bajarlo desde la *cran*.

73 *Caso de estudio*

En el paquete *LifeCycleSavings* hay unos datos que relacionan la tasa de ahorro por país con algunas variables explicativas, cuya lista con sus indicadores en *R* es la siguiente:

```
[,1]  sr     numeric  aggregate personal savings: tasa de ahorro
[,2]  pop15  numeric  % of population under 15 : porcentaje de niños
[,3]  pop75  numeric  % of population over 75  : porcentaje de ancianos
[,4]  dpi    numeric  real per-capita disposable  income: salario real
[,5]  ddpi   numeric  % growth rate of dpi : crecimiento de la economía
```

El siguiente programa que debe ejecutarse en 13 pasos ilustra cómo se procede para analizar unos datos de regresión.

```
#==============================
#CASO DE ESTUDIO: AHORRO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#
#Parte 1: preliminares
#cargar paquete
data(LifeCycleSavings, package="datasets")
#Listar datos
LifeCycleSavings
#Literatura: teclar q para salir de la ayuda
help(LifeCycleSavings)
#
#Parte 2: Histogramas
hist(LifeCycleSavings$sr,  plot = TRUE, breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$ddpi,  plot = TRUE,  breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$pop15,  plot = TRUE, breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$pop75,  plot = TRUE,  breaks="Sturges",
  col="darkgray")
hist(LifeCycleSavings$dpi,  plot = TRUE,  breaks="Sturges",
  col="darkgray")
#Retorne a modo gráfico 1 x 1
par(mfrow=c(1,1), pch=16)
#
#Parte 3: Regresión.
#Regresión univariada todo contra todo.
#Para saber cuál es la variable independiente y
```

```
#cuál la dependiente, guíese por los rangos
#que están en los márgenes.
#La dependencia de sr de las demás variables
#se ve en la primera fila de gráficas.
#No parece haber algo interesante a la vista.
pairs(LifeCycleSavings, panel = panel.smooth,
          main = "LifeCycleSavings data")
pairs(LifeCycleSavings,
          main = "LifeCycleSavings data")
#Regresión multiple para explicar sr
fm1 <- lm(sr ~ pop15 + pop75 + dpi + ddpi, data = LifeCycleSavings)
summary(fm1)
#
#Parte 4: simplificar
fm2 <- lm(sr ~ pop15 + pop75 +  ddpi, data = LifeCycleSavings)
summary(fm2)
fm3 <- lm(sr ~ pop15 +  ddpi, data = LifeCycleSavings)
summary(fm3)
#Gran conclusión: hay un ahorro basal siempre funcionando.
#Lo más malo para el ahorro
# es tener niños.
#Lo mejor es tener una economía en crecimiento.
#
#Parte 5: Análisis de supuestos
#Detección de outliers dentro del modelo
library(car)
outlier.test(fm3)
#Los outliers están por fuera de la región acorralada.
#Cuando R se quede pensando, uno puede
#señalar en la gráfica un dato y aparecerá el nombre.
#Así se detectan los outliers por nombre propio.
#Para terminar: click derecho en la gráfica.
qq.plot(fm3, main="QQ Plot")
#
#Parte 6: importancia del sesgo causado por los outliers.
#Para encontrar  el nombre de un punto, click izquierdo
#Para pasar a la próxima gráfica: click derecho.
#todos en la misma página, 3 por fila, 1 fila.
par(mfrow=c(1,3), pch=16)
leverage.plots(fm3, ask=FALSE) # leverage plots
#
#Parte 7
#Diagrama D de Cook que da la importancia del sesgo
#como outlier causado por cada dato.
#Identificar  D values > 4/(n-k-1)
cutoff <- 4/((nrow(LifeCycleSavings)-length(fm3$coefficients)-2))
plot(fm3, which=4, cook.levels=cutoff)
#
#Parte 8:  normalidad de los residuos
#Distribución de los residuos estandarizados
library(MASS)
sresid <- studres(fm3)
hist(sresid, freq=FALSE, plot = TRUE,
   main="Distribution of Studentized Residuals")
xfit<-seq(min(sresid),max(sresid),length=40)
```

```
yfit<-dnorm(xfit)
lines(xfit, yfit)
#
#Parte 9:  varianzas iguales
# non-constant error variance test
ncv.test(fm3)
# Distribución de residuos a lo largo de
#la variable ajustada por el modelo.
#Si hay problemas, quite los outliers
#o haga regresión robusta.
spread.level.plot(fm3)
#
#Parte 10: las variables explicativas
#deben ser independientes o si no
#las varianzas se inflan.
#Si hay problemas, quite variables explicativas,
#estudie pairs en parte 3 y  matriz de covarianzas.
# variance inflation factors
vif(fm3)
#Hay problemas si vif > 4
#sqrt = square root: raíz cuadrada
sqrt(vif(fm3)) > 2 #
#
#Parte 11
#Test de no-linealidad
#Si falla, proponga un modelo curvo,
#por ejemplo polinómico.
# component + residual plot
cr.plots(fm3, one.page=TRUE, ask=FALSE)
#
#Parte 12
# Otro test de no linealidad
ceres.plots(fm3, one.page=TRUE, ask=FALSE)
#
#Parte 13
# Test de autocorrelación de los residuos,
#deben venir al azar. Se decide por el p-value
#lag = 1 : se compara un dato con el siguiente
durbin.watson(fm3)
```

5.12. Regresión polinómica

Si se acepta la H_o del modelo de la regresión lineal, los datos se explican bien por una línea o por un plano o hiperplano. Si se rechaza, existe la opción de acudir a toda la infinidad de modelos alternativos. Los polinomios pueden ser una buena opción. El siguiente programa presenta una regresión univariada de unos datos que son generados para que se ajusten a una parábola y después se les hace la regresión por un polinomio de grado 2:

```
#==============================
#REGRESION POLINOMICA DE ORDEN 2: SIMULACION
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#SIMULACION
```

```
#Números de 1 a 20
x <- c(1:20)
x
#Generamos 20 observaciones al azar de la normal
w<-rnorm(20, mean = 0, sd = 0.001)
#En parab vamos a meter la parábola
parab <- c(1:20)
#Parábola + error
for (i in 1:20) {parab[i] = 7*i*i -100*i  -20 + w[i]}
parab
plot(parab)
tablaxd <- data.frame(x,parab)
#Regresión polinómica de grado 2
poli2 <- lm(parab ~ poly(x, 2, raw = TRUE), data = tablaxd)
summary(poli2)
#Dibujar el modelo
#En qué puntos se evaluará el modelo:
#200 puntos entre 0 y 20
xseq <- seq(0, 20, length.out = 200)
#Se unen los puntos con una curva suave:
lines(xseq, predict(poli2, data.frame(x = xseq)), col = 2)
#Sofisticaciones
#plot(poli2)
```

Para pensar en regresión polinómica, uno haría bien en visualizar los datos y percibir que no se ajustan a una línea recta. Después, uno puede invocar la regresión polinómica a ver qué logra, como lo ilustra el ejemplo siguiente que relaciona el movimiento de 4 bolsas europeas: DAX, SMI, CAC, FTSE

```
#==============================
#REGRESION POLINOMICA DE ORDEN 2
#BOLSAS EUROPEAS: DAX, SMI, CAC, FTSE
#Correr por partes
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1
data(EuStockMarkets, package="datasets")
Eu <- as.data.frame(EuStockMarkets)
#help(EuStockMarkets)
matplot(Eu,axes=F,frame=T,type='b',ylab="")
#Parte 2
#Datos de DAX
#Números de 1 a 1860
tiempo  <- c(1:1860)
# cargar paquete que dibuja
library(car)
#Dibujar el modelo
scatterplot(Eu$DAX ~ tiempo, reg.line=lm, smooth=FALSE, labels=FALSE,
  boxplots='xy', span=0.5)
#
#Regresión polinómica de grado = grado
grado = 2
poli2 <- lm(DAX ~ poly(tiempo, grado, raw = TRUE), data = Eu)
summary(poli2)
#Dibujar el modelo
```

```
#En qué puntos se evaluará el modelo: tiempo
#Se unen los puntos con una curva suave:
lines(tiempo, predict(poli2, data.frame(x = tiempo)), col = 2)
#Sofisticaciones
#plot(poli2)
#
#Parte 3
#Procedimiento sinónimo para grado = 2:
tiempoala2 <- tiempo*tiempo
poli2s <- lm(DAX ~ tiempo + tiempoala2, data = Eu)
summary(poli2s)
a = poli2s$coefficients[1]
b = poli2s$coefficients[2]
c = poli2s$coefficients[3]
#Inicialización
f  <- c(1:1860)
for (t in 1:1860) {f[t] = a + b * t + c * t*t}
graf <- data.frame(Eu$DAX, f)
#Esta gráfica es más fiel que la de lines.
#La de lines tiene el intersecto mal.
matplot(graf)
```

Cuando uno juega con este programa cambiando el grado del polinomio, uno se da cuenta que, en general, entre más alto sea el grado, hay un mejor ajuste y la curva se parece más a los datos. Entonces uno podría pensar que sería mejor tomar un polinomio de grado tan grande como sea posible, digamos 1500. Esta consideración nos lleva a una pregunta muy importante: ¿qué criterios nos dicen que un modelo es mejor que otro?

Una regla sencilla es buscar un mejor ajuste, medido por el error estándard residual, pero evitando la vanidad, la cual se refleja en involucar coeficientes demasiado pequeños. Naturalmente que ser pequeño o grande es algo que depende del contexto. Cuando se trata de bolsas, el contexto está dado por la situación internacional que por momentos es muy estable pero a veces se pone un poquito caótica. Dependiendo del poder que uno tenga para predecir los momentos críticos, asi mismo uno puede involucrar muchos coeficientes. Para el ejemplo presente, un polinomio de grado 2 parece necesario, pero involucrar un polinomio de mayor grado exige una justificación.

Este tema es toda una disciplina que se llama **Selección de modelos** (Model Selection) y ahí se ventilan criterios de muy diversa naturaleza para escoger un modelo adecuado (Sewell, 2008):

```
Sewell M (2008)
Model Selection
www.modelselection.org/model-selection.pdf
```

Capítulo 6

Regresión multivariada

```
Muchas variables afectan a un sistema complejo
```

74 ***Objetivo.*** *En regresión multivariada se estudian las relaciones de causa-efecto sobre sistemas complejos, con muchas variables estímulo y muchas variables respuesta. El estudio no es por partes sino como un todo: los datos pesan más juntos que separados. La regresión es lineal cuando se asume que cambios en algún estímulo generan cambios proporcionales en alguna respuesta.*

75 ***Ejemplo en pedagogía***

Se cree que los que son buenos para las materias muy técnicas son buenos para todo. Para estudiar si eso es cierto, se tabularon las calificaciones de dos materias técnicas, matemáticas y física, y también las de dos materias no técnicas, literatura y fútbol. Queremos saber que tan buen predictor es el conjunto de notas en materias técnicas de las notas en las no técnicas.

El análisis univariado puede estudiar la capacidad de predicción de matemáticas sobre literatura, el análisi múltiple la de matemáticas y física sobre literatura, pero el análisis multivariado estudia como un todo la influencia de las notas en materias técnicas sobre materias no técnicas. Por consiguiente, el multivariado es el que responde de manera natural la inquietud presentada. En el taller siguiente hacemos todos los estudios.

Atención: el análisis multivariado se edifica enteramente sobre el álgebra lineal y por consiguiente no trabaja con tablas sino con matrices. La diferencia entre tablas y matrices es que, con condiciones apropiadas, las matrices pueden sumarse, restarse, multiplicarse, pero las tablas no. Para convertir dos vectores en las columnas de una matriz se usa el comando *cbind(vect1, vect2)*. Para convertir dos vectores en las filas o renglones de una matriz se usa el comando *rbind(vect1,vect2)*.

```
#==============================
#REGRESION MULTIVARIADA
#Limpiar memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Materias TEcincas y NoTEcnicas
fIsica <-      c(1.1, 2.5, 3.3, 2.4, 1.2, 2.3, 4.4, 2.5, 1.2, 3.3)
matemAtica <-  c(3.1, 1.2, 2.3, 4.2, 2.4, 3.2, 4.1, 1.3, 2.2, 4.1)
literatura <-  c(2.1, 1.4, 3.2, 4.3, 2.3, 1.2, 3.2, 1.2, 3.4, 4.1)
futbol <-      c(2.3, 4.5, 3.4, 2.3, 4.3, 2.3, 3.4, 3.8, 3.9, 4.1)
tablaTodo <- data.frame(fIsica,matemAtica,literatura,futbol)
#Parte 1: correlación univariada
#Estudiamos todas las correlaciones entre pares de variables.
#MatemAticas parece buen predictor:
pairs(tablaTodo, panel = panel.smooth,
          main = "TEcnicas y NoTEcnicas")
```

```
#MatemAticas predice literatura:
matLit <- lm(literatura  ~ matemAtica, data = tablaTodo)
summary(matLit)
#Diagrama de dispersión: cargamos paquete necesario
library(car)
scatterplot(literatura  ~ matemAtica, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = tablaTodo)
#
#Parte 2: Refinamos un caso
#Hay un outlier que podemos quitar: 3.2 contra 1.2
matemAtica9 <-  c(3.1, 1.2, 2.3, 4.2, 2.4,  4.1, 1.3, 2.2, 4.1)
literatura9 <-  c(2.1, 1.4, 3.2, 4.3, 2.3,  3.2, 1.2, 3.4, 4.1)
tablaml9 <- data.frame(matemAtica9,literatura9)
#MatemAticas predice literatura:
matLit <- lm(literatura9  ~ matemAtica9, data = tablaml9)
summary(matLit)
#Diagrama de dispersión:
scatterplot(literatura9  ~ matemAtica9, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = tablaml9)
#
#Parte 3: correlación múltiple
#MatemAticas y fIsica predicen futbol
fut <- lm(futbol  ~ matemAtica + fIsica, data = tablaml9)
summary(fut)
#Hacer dibujo
library(scatterplot3d)
barras <- scatterplot3d(matemAtica,fIsica, futbol,  highlight.3d=TRUE,
type="h", main="3D Scatterplot", data = tablaTodo)
barras$plane(fut)
#
#Parte 4: análisis multivariado
tEcnicas <- cbind(fIsica, matemAtica)
noTEcnicas <- cbind(literatura, futbol)
anova(Predictor1 <- lm (noTEcnicas ~ tEcnicas) )
anova(Predictor2 <- lm (noTEcnicas ~ tEcnicas -1) )
#- residuals almost identical
summary(resid(Predictor1) - resid(Predictor2))
opar <- par(mfrow = c(2,2), oma = c(0, 0, 1.1, 0))
plot(Predictor1, las = 1)      # Residuals, Fitted, ...
par(opar)
```

Lo que nosotros hemos tratado de hacer puede perfeccionarse como sigue: uno comienza con una variable que prometa buen poder explicativo (medido por un bajo p-value en su coeficiente respectivo) y añade otra variable sólo cuando el poder explicativo aumenta significativamente, por ejemplo, si el p-value de la F de toda la prueba disminuye considerablemente. Y también se puede hacer el proceso a la inversa: comenzar con un modelo totalmente lleno e ir quitando variables, una por una y desde la que tenga un p-value más alto hasta que queden únicamente variables con un p-value por debajo del crítico. Como todo esto es rutina, hay un paquete que automatiza el proceso. Se llama *leaps* y hay que bajarlo de la red.

Cuando uno tiene un proyecto con muchas variables, se considera que el proceso de simplificación de lo complejo a lo simple se puede guiar gráficamente, para lo cual el paquete *lattice* podría ser de gran ayuda pues ofrece una buena gama de gráficos tipo Trellis, las cuales presentan trazas en bajas dimensionales de problemas multidimensionales. Cuando la variables explicativas no son todas cuantitativas, el estudio puede incluir manovas, que es el análisis de varianza multivariado. Las gráficas de Trellis son perfectas para complementar dicho estudio.

```
Robert Kabacoff (2008)
Trellis graphs
http://www.statmethods.net/advgraphs/trellis.html
```

Los supuestos para el análisis de varianza multivariado son muy pesados: se necesita una distribución normal en todo el espacio de variables explicativas, lo mismo que en el espacio de variables respuesta y el espacio conjunto. Se requiere que las variables explicativas sean independientes. Por ello, el análisis multivariado es un arte complicado, para el cual R promete tener muchas ayudas:

```
Hewson P (2010) Multivariate Statistics
http://cran.r-project.org/web/views/Multivariate.html
```

Para saber más sobre regresión desde la univariada hasta el proceso de automatización de simplificación de modelos multivariados, el siguiente link lleva a un curso muy bien desarrollado, el cual incluye discusiones muy enriquecedoras con código y muchos consejos casi maternales.

```
Utts Jessica (2007) Regression analysis,
http://www.ics.uci.edu/~jutts/st108/
```

6.1. Conclusión

Decimos que tenemos un modelo de regresión lineal cuando la variable de salida cambia proporcionalmente a los cambios de la variable de entrada. Discernir cuándo se justifica un modelo de regresión lineal es algo que puede hacerse en presencia de ruido, lo cual se hace con una F, la cual compara la varianza debida al supuesto modelo con la varianza causada por el ruido. Si la F resultante es grande, comparada con un valor crítico correspondiente, se rechaza la H_o y uno tiene permiso para creer que un cambio en la variable independiente es seguido por un cambio en la variable dependiente. Hemos visto diversos grados de complejidad de este esquema: lineal univariado (una variable estímulo y otr de respuesta) , lineal múltiple (varias variables estímulo y otra de respuesta), multilineal (varias variables estímulo y varias de respuesta).

Capítulo 7

Descriptores sintéticos

```
Decir lo máximo con lo mínimo.
```

76 ***Introducción y objetivo.*** *Sintetizar la información contenida en los datos es algo que nos permite avanzar más rápidamente en la toma de decisiones. El nombre técnico de la disciplina correspondiente es* ***reducción dimensional*** *y cuyo objetivo básico es simplificar los datos, quitando variables o factores redundandes o proponiendo un nuevo punto de vista que simplifique la descripción de los datos. Dentro de estas metodologías vamos a ver en primer lugar la llamada* ***teoría de componentes principales****, cuyo objetivo es proponer un conjunto de macrovariables o combinaciones lineales de las variables originales con el objetivo de explicar con el mínimo de ellas lo máximo de la variación de los datos. En segundo término veremos la teoría de análisis factorial (factor analysis).*

7.1. Análisis de componentes principales

El objetivo de la teoría de componentes principales es presentar una metodología para sintetizar datos cuantitativos pagando el mínimo costo posible en la pérdida de información.

El principio que anima la metodología es el mismo que se usa en el lenguaje común: en vez de decir que Fulano es muy despierto para percibir y utilizar las circunstancias, capacitado para tomar decisiones, bien entrenado en lo operacional, dispuesto a esforzarse cuando toca, decimos simplemente que Fulano es un ejecutivo. La idea es pues, combinar una serie de variables en una sola **macrovariable** o en unas cuantas o en cualquier caso en el mínimo número posible procurando que el poder descriptivo, medido por la varianza, permanezca tan grande como se pueda. Por lo tanto, el análisis de componentes principales lo capacita uno a decir lo máximo de lo que falta por decir usando lo mínimo de palabrería.

77 ***Ejemplo: aprender a dibujar***

Las profesoras de niños pequeños saben que aprender a dibujar es algo muy necesario para adquirir un fino control motriz en tanto que uno se divierte. En un colegio dan dos horas de dibujo con colores, pero cada hora tiene su propia nota, a la cual la profesora les pone igual calificación: la correlación entre las dos notas es perfecta. Lo que veremos es la percepción del mismo fenómeno desde la teoría de componentes principales.

```
#==============================
#ANALISIS DE COMPONENTES PRINCIPALES: COLORES
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Colores1 y colores2
#Parte 1: regresión
#Notas de niños de  preescolar en dos materias.
colores1 <- c(1,2,3,2,5,4,6,3,7,4,6,5,7,4,5,8)
```

```
colores2 <- colores1
tabla <- data.frame(colores1,colores2)
#Esperamos una alta correlación:
library(car)
scatterplot(colores1  ~ colores2, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = tabla)
#Análisis de regresión
dibujo <- lm(colores1 ~ colores2)
summary(dibujo)
#
#Parte 2: análisis de componentes principales
comp <- princomp(tabla, cor=FALSE)
#Ordene las componentes de acuerdo a su importancia
#para explicar la varianza
summary(comp)
#Diagrama de importancias relativas
plot(comp)
#
#Conclusión: las dos materias son en un 100%  más de lo mismo.
#¿Qué es exactamente más de lo mismo?
#Es la componente principal uno:
#una aptitud hacia el dibujo con colores.
```

En otro colegio también hay dos horas para dibujo, pero en la primera hora hay una actividad en la cual los niños aprenden a dibujar con colores y en la otra con acuarelas. Por regla general, el que es bueno para lo uno, lo es para lo otro. Así que esperamos una alta correlación entre las dos variables, pero la correlación no será perfecta pues dibujar con acuarelas requiere aprender a manejar el tiempo de secado, lo cual es algo bastante difícil.

```
#==============================
#ANALISIS DE COMPONENTES PRINCIPALES: ACUARELA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Colores y acuarela
#Parte 1: regresión
#Notas de niños de  preescolar en dos materias.
colores <-  c(1,7,3,2,5,4,6,3,7,4,6,5,4,5,8,2)
acuarela <- c(2,6,1,3,4,5,5,4,5,3,4,5,3,6,7,2)
tabla2 <- data.frame(colores,acuarela)
#Esperamos una alta correlación:
library(car)
scatterplot(colores  ~ acuarela, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = tabla2)
#Análisis de regresión
dibujo <- lm(acuarela ~ colores, data = tabla2)
summary(dibujo)
#
#Parte 2: análisis de componentes principales
comp2 <- princomp(tabla2, cor=FALSE)
#Ordene las componentes de acuerdo a su importancia
#para explicar la varianza
summary(comp2)
#Diagrama de importancias relativas
```

```
plot(comp2)
#
#Parte 3
#Descomponga cada variable como
#combinación de las componentes principales:
#hay dos componentes, la primera es aptitud para dibujar,
#la segunda es aptitud para manejar procesos con interferencia
#que tiene que ver con los tiempos de secado.
loadings(comp2)
biplot(comp2)
#
#Parte 4
#Explique cada individuo como una
#combinación de las componentes principales:
comp2$scores # the principal components
#
#Conclusión: las dos materias son en un 91%  más de lo mismo.
# ¿Qué es exactamente más de lo mismo?
#Es una aptitud hacia el arte del dibujo
#que permitirá aconsejar a quienes deseen
#aprender a dibujar con óleo.
#Por ejemplo: los estudiantes que aparecen en la lista
#en los lugares 1, 3 y 16 no sirven para el óleo
#a no ser que tomen consejería especial.
```

Cuando los niños ya están más grandes, no se les da colores y acuarela sino que se les da un única materia de dibujo, pues hay muchas cosas diversas que aprender. En compensación, se les da una nueva materia que signifique un desafío para los niños. Tenemos un doble trabajo: garantizar a los padres que no se pierde nada al sustituír colores y acuarela por dibujo y que la nueva materia representa un desafío alcanzable para los niños.

```
#==============================
#ANALISIS DE COMPONENTES PRINCIPALES: DIBUJO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Colores y escultura
#Parte 1: correlación
#Notas de niños de  primero en dos materias.
dibujo <-    c(1,2,3,2,5,4,6,3,7,4,6,5,7,4,5,8)
escultura <- c(0,4,1,1,1,7,4,4,4,1,3,6,9,7,8,5)
tabla3 <- data.frame(dibujo,escultura)
library(car)
scatterplot(escultura  ~ dibujo, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = tabla3)
arte <- lm(escultura ~ dibujo)
summary(arte)
#Conclusión: la correlación el alta pero no es significativa,
#las dos materias no son más de lo mismo.
#Pero, ¿será la escultura un desafío alcanzable?
#¿Qué tanta novedad hay en la escultura con respecto
#a la materia de dibujo?
#
#Parte 2: preparar datos
#Centrar los datos:
```

```
dibujoc = dibujo - mean(dibujo)
esculturac <- escultura - mean(escultura)
tablac <- data.frame(dibujoc, esculturac)
scatterplot(esculturac  ~ dibujoc, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data = tablac)
#
#Parte 3:
#análisis de componentes principales
comp3 <- princomp(tablac, cor=FALSE)
#Ordene las componentes de acuerdo a su importancia
#para explicar la varianza
summary(comp3)
#Diagrama de importancias relativas
plot(comp3)
#Conclusión: las dos materias tienen un 77% en común
#representada en la primera componente principal,
#que es una aptitud para modelar,
#pero hay una divergencia del 22% que corresponde
#a una capacidad para diversificarse, pues se relaciona
#con cosas específicas que se aprenden en una materia
#o en la otra.
#Parte 4
#Descomponga cada variable como
#combinación de las componentes principales
loadings(comp3)
biplot(comp3)
#Explique cada individuo como una
#combinación de las componentes principales:
comp3$scores # the principal components
#Los alumnos que parecen en la lista en los lugares
#1, 2 y 3 necesitan consejería urgente.
```

7.2. Análisis factorial

El análisis de componentes principales tiene como objetivo proponer macrovariables, combinaciones de las variables originales, que sean nocorrelacionadas y que maximicen su capacidad sintetizadora al máximo: que con lo mínimo cubran lo máximo de varianza. El análisis factorial es un complemento que también propone macrovariables pero pone su punto de vista no sobre la varianza sino sobre las covarianzas: encontrar la máxima comunalidad entre las variables minimizando la covarianza con lo ya dicho.

```
#==============================
#ANALISIS FACTORIAL
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: datos de un colegio
dibujo <-    c(1,2,3,2,5,4,6,3,7,4,6,5,7,4,5,8)
escultura <- c(0,4,1,1,1,7,4,4,4,1,3,6,9,7,8,5)
lenguaje <-  c(6,5,4,6,7,4,5,8,7,6,5,6,7,6,7,5)
matemAtica<- c(1,3,5,4,3,6,7,5,6,3,4,5,2,3,4,6)
ciencias <-  c(3,6,4,5,6,5,6,6,5,4,4,2,4,5,5,5)
canto <-     c(3,2,4,5,7,5,3,8,7,5,4,7,2,5,4,6)
tabla <- data.frame(dibujo,escultura, lenguaje,matemAtica,ciencias,canto)
```

```
#Análisis factorial: se empieza con un factor o macrovariable.
fact <- factanal(tabla, factors=1, rotation="varimax")
print(fact, digits=2, cutoff=.3, sort=TRUE)
descomponer <- fact$loadings
descomponer
update(fact, factors=2, rotation="promax")
update(fact, factors=3, rotation="promax")
#
#Comparar con análisis de componentes principales
comp <- princomp(tabla, cor=FALSE)
#Ordene las componentes de acuerdo a su importancia
#para explicar la varianza
summary(comp)
#Diagrama de importancias relativas
plot(comp)
#Parte 4
#Descomponga cada variable como
#combinación de las componentes principales
loadings(comp)
```

Capítulo 8

Regresión robusta

Apaciguar los problemas con los outliers

78 ***Motivación y bjetivo.*** *La regresión que se basa en la metodología de los mínimos cuadrados es muy sensible a los outliers. La regresión robusta ofrece otra metodología que es más estable, robusta, ante ellos.*

8.1. Solución primitiva: detecte y quite los outliers

Toda la estadística que hemos visto es muy sensible a los outliers, a tal grado que pequeños desajustes en los supuestos pueden causar fuertes sesgos. Por ejemplo: queremos calcular la media poblacional y tenemos los datos -1, 0, 1. Su media es cero y no hay razón para designar a alguno de los datos como outlier. Pero si añadimos el dato 40, que sería un dato extremo, un **outlier**, los datos quedan -1, 0, 1, 40. La nueva media es 10 y nos damos cuenta que un outlier causó un corrimiento tremendo de la estimación de la media. Y lo dicho de la media también es válido para todo lo que uno pueda imaginarse. La **estadística robusta** es una reacción que persigue producir prodecimientos de análisis de datos que sean más estables ante el influjo de los outliers.

Pues si los outliers molestan, lo más fácil es quitarlos. Esto nos dice que la estadística robusta debe empezar por elaborar una metodología para la detección de outliers. Debido a que el poder de análisis visual que tenemos los humanos es impresionante, lo más fácil y directo es usar gráficas.

79 ***Ejemplo***

Si tenemos una lista de datos, los outliers se detectan por medio de un histograma. Si los datos vienen de una poblacion distribuída centralmente, la media y varianza de la población quedaría mejor estimada por los datos sin los outliers que con ellos.

```
#==============================
#DETECTAR y QUITAR OUTLIERS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Todas las gráficas en la misma página
par(mfrow=c(2,1), pch=16)
 x <- c(3,4,5,4,6,5,4,6,5,4,7,4,5,4,6,5, 20)
hist(x, seq(0, 22, 1), plot = TRUE, prob=T)
mean(x)
var(x)
 y <- c(3,4,5,4,6,5,4,6,5,4,7,4,5,4,6,5)
hist(y, seq(0, 22, 1), plot = TRUE, prob=T)
```

```
mean(y)
var(y)
```

Nos damos cuenta de que la varianza puede ser mucho más sensible a los outliers que la media.

80 ***Ejemplo***

Estamos en una regresión: al hacer un text de outliers, detectamos uno en la variable de salida, lo quitamos y hacemos la regresión de nuevo:

```
#==============================
#DETECTAR y QUITAR OUTLIERS EN UNA REGRESION
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: Detectar outliers
x <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
y <- c(0,1,2,3,4,5,6,9,9,10,13,13,15,16,17,35)
r <- data.frame(y,x)
library(car)
scatterplot(y~x, reg.line=lm, smooth=FALSE, labels=FALSE,
  boxplots='xy', span=0.5, data=r)
z<-lm(y ~ x, data = r)
z
plot(z)
#Conclusión:
#El dato número 16 es un  outlier intolerable: quitémoslo
#
#Parte 2: quitar outliers
x <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)
y <- c(0,1,2,3,4,5,6,9,9,10,13,13,15,16,17)
rClean <- data.frame(y,x)
scatterplot(y~x, reg.line=lm, smooth=TRUE, labels=FALSE,
  boxplots='xy', span=0.5, data=rClean)
w<-lm(y ~ x, data = rClean)
w
#Línea de regresión sin outliers en negro
a = w$coefficients[1]
b = w$coefficients[2]
abline(a,b, col = "black")
#Línea de regresión con outliers en azul
a2 = z$coefficients[1]
b2 = z$coefficients[2]
abline(a2,b2, col = "blue")
#Parte 3: revisar los nuevos residuos
#Sin outliers, todo está más tranquilo:
plot(w)
```

Debido a que uno puede quitar un outlier y generar otros, esta metodología para detectarlos es muy engorrosa. Se espera que funcione bien cuando los outliers sean muy poquitos. Pero como la estadística tiene que ver caa vez más con grandes masas de datos, toda esta tecnología puede llegar a considerarse como primitiva. Aunque uno puede pensar en automatizarla, la idea difícil, tal vez genial, es atacar el problema desde su base y ver a qué se debe que los outliers molesten tanto. Diversas ideas han sido propuestas y estudiadas (Roncheti, 2006)

Ronchetti E (2006) The historical development of robust statistics

www.stat.auckland.ac.nz/˜iase/publications/17/3B1_RONC.pdf

Este tema está bajo investigación y en cierta forma no es muy popular. En realidad, detectar y quitar los outliers es algo que puede resultar mucho más práctico en muchos casos. Sin embargo, hay algunas ideas muy cómodas de entender como la siguiente:

8.2. Regresión a la Huber

Vamos a ver una idea debida a Huber quien la publicó en 1964 para estimar el centro de una distribución pero fue después generalizada a la regresión. Explicaremos sobre análisis univariado la introducción especialemente preparada para R por John Fox (2002) y que desarrolla el caso de la regresión múltiple.

Fox J (2002) Robust regression
cran.r-project.org/doc/.../appendix-robust-regression.pdf

81 ***La idea es generalizar***

El método de los mínimos cuadrados fue escogido simplemente por la facilidad con que se estudia. Se ha observado que este modelo de regresión es muy sensible a los outliers. Ahora bien: ¿qué de raro puede tener que esta metodología se enloquesca con los outliers si es que ellos se ven a través de un cuadrado que magnifica todo lo que se aleja de cero? En efecto, en este método se minimiza la función

$E = \sum_i^n (y_i - a - bx_i)^2$

si definimos el error e_i en el punto x_i como la diferencia entre la salida observada y la salida esperada según el modelo lineal

$e_i = y_i - a - bx_i$

entonces E toma la siguiente expresión

$E = \sum_i^n (e_i)^2$

en donde queda claro que los errores se ven a través de una parábola la cual hace ver lo pequeño aún más pequeño pero lo grande mucho más grande. (Por la misma razón, el mismo problema adolece a la varianza de una muestra como estimador de la varibilidad poblacional.)

¿Cuál puede ser entonces el remedio?

Todo se trata de minimizar una medida de la discrepancia entre lo que se ve y lo que se espera por un modelo lineal, pero no es obligatorio tomar la metodología de los mínimos cuadrados. ¿Cuáles son las buenas cosas de esta metodología y cómo se generalizan?

Hay un universo de generalizaciones de la función E de mínimos cuadrados de la forma:
$E(\rho) = \sum_i^n \rho(e_i)$
donde las propiedades de ρ deberían ser muy sencillas, muy similares a la de los mínimos cuadrados en cuyo caso $\rho(e) = e^2$:

- $\rho(e) \geq 0$, la medida del error debe ser positiva o cero.
- $\rho(0) = 0$, la medida del error debe ser fidedigna.
- $\rho(e) = \rho(-e)$, da lo mismo un error por arriba que por abajo.
- $\rho(e_i) \geq \rho(e_{i'})$ si $\mid e_i \mid > \mid e_{i'} \mid$, errores mayores deben dar medidas mayores.

El objetivo es minimizar E, para lo cual se deriva con respecto a y con respecto a b. Estudiémos la derivada con respecto a b en el caso de los mínimos cuadrados:

$\partial E/\partial b = \sum_i^n 2(y_i - a - bx_i)(-x_i) = 2\sum_i^n (y_i - a - bx_i)(-x_i) = 2\sum_i^n (e_i)(x_i) = 0$

la cual se puede generalizar en

$\sum_i^n x_i w(e) = 0$

donde w es una función que da peso (weight) a los residuos.

En el procedimiento de los mínimos cuadrados, la elección de ρ y w están ligados, pero también se puede considerar opciones separadas, por ejemplo, en el método de los cuadrados sirve $\rho(e) = e^2$ con $w(e) = 1$ al igual que $w(e) = 2$ y multitud de otras opciones.

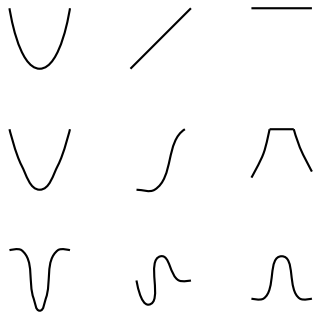

Figure 8.0. Algunas metodologías famosas para cuantificar el error global a minimizar en regresión. En la primer fila está la elección de los mínimos cuadrados, en la segunda la de Huber y en la tercera la de Tukey. En la primer columna tenemos ρ, de la cual se computa la derivada ψ, en la segunda columna, y en la tercera columna aparece w.

Una vez que se han hecho las elecciones de ρ, de la cual se computa la derivada ψ, y de w, sigue un trabajo numérico delicado para despejar las ecuaciones y R está programando para hacerlo por nosotros. En el siguiente programa lm indica la regresión ordinaria y rlm la regresión robusta. La línea de regresión ordinaria va en negro, la robusta en rojo.

```
#============================
#REGRESION ROBUSTA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: datos con un outlier
x <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
y <- c(0,1,2,3,4,5,6,9,9,10,13,13,15,16,17,35)
r <- data.frame(y,x)
library(car)
scatterplot(r$y~r$x, smooth = FALSE)
#
#Mientras que R piensa, señale puntos con click izquierdo
#y termine con click derecho:
identify(x,y)
#
#Parte 2
```

```
#Regresión a la Huber
library(MASS)
#rlm = robust linear model = regresión robusta
Huber <- rlm(y ~ x, data = r)
summary(Huber)
#Huber2 = sinónimo de Huber
Huber2 <- rlm(y ~ x, method = "M", data = r)
summary(Huber2)
plot(Huber)
#
#Parte 3
#Peso de cada dato:
#El dato 16 tiene muy poco peso:
#Huber$w = weights = peso de los datos
plot(Huber$w, ylab="Huber Weight", col = "blue")
#Mientras que R piensa, señale puntos con click izquierdo
#y termine con click derecho:
identify(1:16, Huber$w)
#
#Parte 4:
#Comparemos con la regresión odinaria
#lm =liner model = regresión ordinaria
ro<-lm(y ~ x, data = r)
ro
library(car)
scatterplot(r$y~r$x, smooth = FALSE)
#Regresión ordinaria en negro
a = ro$coefficient[1]
b = ro$coefficient[2]
abline(a,b,col = "black")
#Regresión de Huber en azul
aH = Huber$coefficients[1]
bH = Huber$coefficients[2]
abline(aH,bH, col = "blue")
#Regresion a la Tukey en verde
Tukey <- rlm(y ~ x,  psi = psi.bisquare,data = r)
summary(Tukey)
aT = Tukey$coefficients[1]
bT = Tukey$coefficients[2]
abline(aT,bT, col = "green")
#Regresion a la Hampel en amarillo
Hampel <- rlm(y ~ x, psi = psi.hampel,init = "lts", data = r)
summary(Hampel)
aP = Hampel$coefficients[1]
bP = Hampel$coefficients[2]
abline(aP,bP, col = "yellow")
#Otra variante
Tukey2 <- rlm(y ~ x, method = "MM",  psi = psi.bisquare,data = r)
summary(Tukey2)
```

Conclusión válida para R versión 2.11.05.1: a no ser que haya un bug en la programación, el método de quitar outliers parece mejor que la regresión robusta.

Capítulo 9

Modelos lineales generalizados

`Usar métodos lineales para estudios no lineales`

82 ***Objetivo.*** *Con ligeras modificaciones, los modelos lineales se pueden aplicar a problemas de modelamiento no lineal. Tenemos ahora una buena batería de modelos sin los cuales es inimaginable la estadística moderna.*

9.1. Rregresión exponencial

Si uno tiene unos datos de los cuales uno supone que deben seguir la ley

$y = ae^{kt}$

uno está en el mundo de la regresión exponencial. Este tipo de regresión se reduce a una regresión ordinaria si uno toma logaritmo natural:

$lg(y) = log(a) + ktlog(e)$
$lg(y) = log(a) + kt$

que es de la forma
$w = c + kt$

donde $w = log(y)$ y $c = log(a)$ y por lo tanto $a = e^c$

83 ***Example*** *Ajustemos datos de la forma (x, y) a un modelo de regresión exponencial. Podemos interpretar x como el tiempo, y_o representa una población inicial, c es la tasa reproductiva, si es positva, o la de decaimiento si es negativa. El modelo es:*

$y = y_o e^{cx} + ruido,$

donde $y_o = 100$ y $c = -0,3$. Tomamos logaritmo natural y después ajustamos un modelo lineal sobre (x, lny).

Ejecutamos:

El modelo exponencial. La última fila tiene las sumas.							
x	$s = y_o e^{cx}$	$ruido$	$y = s + rudio$	lny	$xlny$	x^2	$(lny)^2$
0	100	0,2	100,2	4,607168189	0	0	21,22599872
1	74,08182207	-0,1	73,98182207	4,303819415	4,303819415	1	18,52286156
2	54,88116361	0,3	55,18116361	4,010621656	8,021243312	4	16,08508607
3	40,65696597	-0,3	40,35696597	3,697764019	11,09329206	9	13,67345874
4	30,11942119	0,2	30,31942119	3,411788471	13,64715388	16	11,64030057
5	22,31301601	-0,1	22,21301601	3,100678424	15,50339212	25	9,614206689
6	16,52988882	-0,2	16,32988882	2,792997099	16,75798259	36	7,800832794
7	12,24564283	0,3	12,54564283	2,52937342	17,70561394	49	6,397729898
8	9,071795329	0,1	9,171795329	2,21613305	17,7290644	64	4,911245695
9	6,720551274	0,2	6,920551274	1,93449543	17,41045887	81	3,74227257
10	4,978706837	0,3	5,278706837	1,66368115	16,6368115	100	2,76783497
55	371,5989739			34,26852032	138,8088321	385	116,3818283

Las medias son: $\bar{x} = 55/11 = 5$.

$\bar{y} \to \bar{lny} = 34,26852032/11 = 3,115320029$

Ahora calculamos el coeficiente de regresión lineal:

$b = \frac{\sum xy - n\bar{x}\bar{y}}{\sum x^2 - n\bar{x}^2}$

con lny en lugar de y:

$b = \frac{\sum xlny - n\bar{x}\bar{lny}}{\sum x^2 - n\bar{x}^2}$
$b = \frac{138,8088321 - 11(5)(3,115320029)}{385 - n(5)^2}$
$b = \frac{-32,53377}{110} = -0,2957615$
$a = \bar{y} - b\bar{x} \to \bar{lny} - b\bar{x} = 3,115320029 - (-0,2957615)(5)$
$a = 3,115320029 - (-0,2957615)(5) = 4,594127$

por tanto: el modelo lineal produce $lny = 4{,}59412 - 0{,}2957x$, con un coeficiente de regresión de 0.99. Por ende, el modelo exponencial es:

$y = e^a e - bx = e^4{,}59412e^{-0,2957x} = 98{,}90e^{-0,2957x}$.

El siguiente programa ilustra la forma de ejecutar un regresión exponencial en R:

```
#==============================
#REGRESION EXPONENCIAL: SIMULACION
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Fabricamos los datos:
#x,y,z son variables explicativas
#w es variable respuesta, tiene ruido
t <- c(1:10)
y <- t
#Ruido: generamos 10 observaciones al azar de la normal
n <- c(1:10)
n<-rnorm(10, mean = 0, sd = 0.1)
#Datos con dependencia exponencial
for(i in 1:10)
{ y[i] <-  6* exp(2*t[i])  + n[i]}
```

```
tabla <- data.frame(t,y)
tabla

#Regresión exponencial
w = log(y)
tablaGrande<-cbind(tabla, w)
modelo <- lm(w ~ t)
a<- exp(modelo$coefficient[1])
a
k<-modelo$coefficient[2]
k
#Modelo interpolado:
library(car)
#100 puntos entre 0 y 10
tseq <- seq(0, 10, length.out = 100)
yModelo <- seq(0, 10, length.out = 100)
for(i in 1:100)
{ yModelo[i] <-  a* exp(k*tseq[i]) }
matplot(tseq, yModelo, col = "blue",  pch = 16)
```

9.2. Regresión polinomial múltiple

Un modelo lineal múltiple propone una relación lineal entre un conjunto con más de una variable explicativa y una variable respuesta. Tiene el siguiente aspecto

$$w = 3x + 2y - z + 8$$

Si uno tiene unos datos que no aceptan un modelo lineal múltiple, uno puede optar por un modelo polinomial múltiple que puede lucir como el siguiente:

$$p = x^3 + 2xy + 3x + 2y - z + 8$$

Para llegar a un modelo como el anterior, hay que partir de un modelo completo con todas las potencias y todas las combinaciones e ir quitando términos, uno por uno, desde aquellos con p-value más grande, hasta que queden los que tengan un p-value por debajo de la significancia crítica.

Para calcular un modelo polinomial múltiple, se adapta el siguiente programa que estudia el modelo sobre unos datos de simulación:

```
#==============================
#REGRESION POLINOMIAL MULTIPLE: SIMULACION
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Fabricamos los datos:
#x,y,z son variables explicativas

x <- rep (c(1:5), each = 25)
y <- rep( c(1:5), each = 5, times = 5)
z <- rep( c(1:5), times = 25)
#Ruido: generamos 125 observaciones al azar de la normal
#Inicializamos n simbólicamente
n <- c(1:125)
#Inicialización verdadera
```

```
n<-rnorm(125, mean = 0, sd = 0.1)
 #w es variable respuesta, tiene ruido
#Inicialización simbólica
w <- n
#Inicialización verdadera
for(i in 1:125)
{ w[i] <-  8 + 3*x[i] + 2* z[i] + 2*x[i] * y[i] +  x[i]*x[i]*x[i]  + n[i]}
tabla <- data.frame(x,y,z,w)
tabla
#
#Regresión polinomial múltiple
#Modo uno: normal
xy = x*y
xcubo <- x*x*x
#tabla ampliada 1
#Paso 2: regresión
tablaGrande1 <- cbind(tabla,  xy,xcubo)
multiplePoliReg  <- lm(w ~ x  + z + xy+ xcubo , data = tablaGrande1)
summary(multiplePoliReg)
#
#Regresión polinomial múltiple
#Modo dos: variables centradas, se resta la media
xm <- x -mean(x)
ym <- y - mean(y)
zm <- z - mean(z)
xmcubo <- xm*xm*xm
xmym = xm*ym
#tabla ampliada 2
#Paso 2: regresión
tablaGrande2 <- cbind(tabla, xm, ym,zm,xcubo,xy)
multiplePoliReg2  <- lm(w ~ xm  + zm + xmym+ xmcubo , data = tablaGrande2)
summary(multiplePoliReg2)
```

9.3. Regresión de Lowess

Un modelo muy práctico de regresión es la regresión de Lowess que hace regresión pero por pedazos. Se decide automáticamente si la regresión es lineal o polinómica. El parámetro f indica el tamaño de los entorno a tener en cuenta, entre más grande, más suave será la curva quebrada de Lowess:

```
#=============================
#REGRESION  DE LOWESS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
x <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
y <- c(0,1,2,3,4,5,6,9,9,10,13,13,15,16,17,35)
r <- data.frame(x,y)
#main: título principal
plot(r, main = "lowess(y vs x)")
#f = factor de suavización, entre más grande, más suave
lines(lowess(r, f=.2), col = 2)
lines(lowess(r, f=.4), col = 3)
lines(lowess(r, f = 2/3), col = 4)
#Caja de letreros: posición, contenido, lty = line types,, colores
```

```
legend(2, 30, c(paste("f = ", c(".2",".4", "2/3" ))), lty = 1, col = 2:3:4)
```

9.4. El modelo logístico

En un modelo logístico tenemos la siguiente dependencia no lineal:

$y(t) = \frac{K}{1+C_o e^{-rt}}$

y su gráfica es del siguiente estilo:

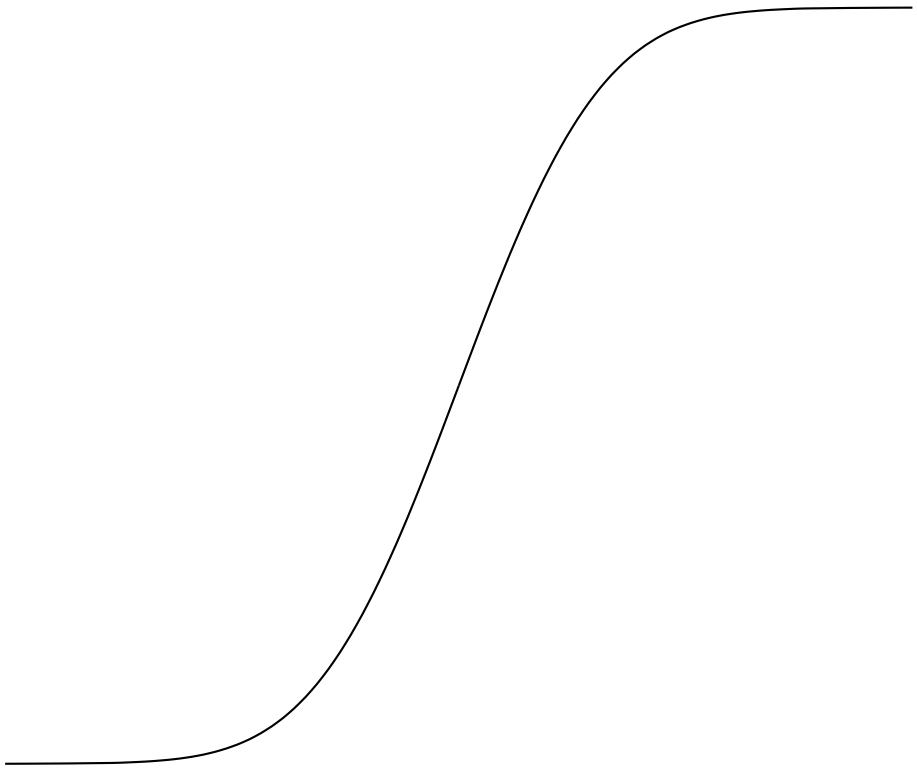

Figura 9.0. Un modelo logísitico representa algunos tipos de crecimiento

Este modelo es una propuesta usual para ajustra el crecimeinto de una población o el de una planta. Se sabe que muchos tipos de crecimiento no se puede describir por este modelo, por ejemplo, el de humanos.

Nuestro análisis combina sencillez y efectividad (no así la literatura). Usamos el hecho de que un modelo logístico es la consecuencia natural de una consideración muy simple: el crecimiento de una población es el resultado de la interacción de dos factores principales: reproducción y ecología. La reprodrucción nos dice que el número de bebés es proporcional al número de padres. La ecología dice que una población grande daña el ambiente de tal manera que éste se torna enemigo del crecimiento poblacional. Pero, ¿cuándo decimos que una población es grande? Eso depende de la relación entre ella y su ecosistema. La ecuación más simple que contiene estas ideas es la siguiente:

Supongamos que el ecosistema tiene capacidad para alvergar K individuos. Entonces una población es grande o pequeña en relación a K, a la cual se la llama la capacidad de carga.

Consideremos una población pequeña, que no crea problemas ecológicos y que simplemente se reproduce. Sea t el tiempo, sea y el tamaño de la población. Denotamos por dy/dt el incremento o decremento

de la población por unidad de tiempo. Si esta fracción es positiva, la población crece. Si es negativa, decrece. Si es cero, la población permanece constante. Tenemos:

$\frac{dy}{dt} = ry$

Estamos diciendo que la reproducción es proporcional al número de individuos. Tomemos ahora a la ecología en consideración, cuando la población es grande. En dicho caso, el alimento puede escasear o las interacciones entre individuos pueden tornarse antireproductivas. Modelamos esta situación por

$\frac{dy}{dt} = ryF$

donde F ies un coeficiente de depresión que depende de la relación entre y y K. A F se le puede definir mediante la siguiente ecuación:

$\frac{dy}{dt} = ry(\frac{K-y}{K})$

Vemos que cuando la población es pequeña, $F = \frac{K-y}{K}$ es casi 1 y en dicho caso la ecología no afecta el crecimiento. Pero cuando la población crece y llega a ser grande, y crece hacia K, F es casi cero y la población crece despacio o se estabiliza. Tenemos un crecimiento suave con una estabilización no catastrófica. Incluso, cuando F es negativa, como por una inmigración, la población declina.

Puede probarse que la solución a la ecuación diferencial anterior es

$y(t) = \frac{K}{1+C_o e^{-rt}}$ donde $C_o = \frac{K-y(0)}{y(0)}$.

Usaremos ahora esas ideas para poner a prueba que unos datos se ajustan al modelo logístico: en vez de mirar a la solución de la ecuación, podemos estudiar la ecuación diferencial directamente:

84 ***Ejemplo*** *Tabulemos la solución $y(t)$ del modelo logístico conjunatmente con el valor de dy/dt dado por*

$dy/dt = \frac{dy}{dt} = ry(\frac{K-y}{K})$

Sea

$r = 0,2$

$K = 100$

$N_o = 1$

$C_o = 99$

El modelo logístico		
t	$y(t)$	dy/dt
1	1,218704513	0,240770421
2	1,484523445	0,292497069
3	1,807260968	0,354919809
4	2,198596232	0,430051596
5	2,672363099	0,520189571
6	3,244833003	0,627908718
7	3,934978505	0,756027589
8	4,764682962	0,907532185
9	5,758846041	1,085440593
10	6,945315966	1,292588365
11	8,354559011	1,53131449
12	10,0189585	1,803032641
13	11,97162594	2,107685533
14	14,24461613	2,443105048
15	16,86647887	2,804339555
16	19,85916597	3,183060248
17	23,23445055	3,567210725
18	26,99019983	3,941098192
19	31,10703969	4,286112101
20	35,54609871	4,582169475
21	40,2485411	4,809818099
22	45,13742121	4,952710655
23	50,1220035	4,99997023
24	55,10416192	4,947895062
25	59,98596018	4,800561199
26	64,67720123	4,569159528
27	69,10174235	4,270246878
28	73,20169917	3,923362311
29	76,93917971	3,548561193
30	80,29571528	3,164339272
31	83,26993487	2,786222868
32	85,87419549	2,426084197
33	88,1308511	2,092076389
34	90,06868947	1,789000248
35	91,71986831	1,518905176
36	93,11749982	1,281762419
37	94,29389692	1,076101391
38	95,27941094	0,89954989
39	96,10175134	0,749257046
40	96,78567044	0,622202083
41	97,35290601	0,515404586
42	97,8222933	0,426056526
43	98,20997842	0,351595961
44	98,52968346	0,289739646
45	98,79298967	0,238488317
46	99,00961663	0,196114956
47	99,18768436	0,161143415
48	99,33395146	0,132322467
49	99,4540264	0,108598545
50	99,55255179	0,089089222

Si dibujamos y contra t, observamos una curva en forma sigmoidea, pero si graficamos dy/dt contra

$y(t)$ obtenemos una parábola (ejercicio sobre *Excel* o *Gnumeric*). Esto ilustra un resultado general

85 $\diamond$ ***Teorema.*** *Para poner a prueba la hipótesis de que unos datos $(t, y(t))$ se ajustan a un modelo logístico es equivalente a poner a prueba la hipótesis de que los datos $(y, dy/dt)$ se ajustan a una parábola.*

En el modelo de regresión múltiple con dos variables, nosotros ponemos a prueba la hipótesis de que $y = a + b_1x_1 + b_2x_2$ y necesitamos una tabla con los valores $x_1y,\ x_2y,\ x_1x_2,\ x_1^2,\ x_2^2, y^2$. Una regresión parabólica es de la forma $y = d + ex + fx^2$.

Un modelo de regresión parabólica $y = d + ex + fx^2$ se reduce a otro de regresión lineal doble $y = a + b_1x_1 + b_2x_2$ done la x va en lugar de x_1 y x^2 en lugar de x_2.

Para estudiar el modelo logísitico, se estudia la regresión parabólica sobre los datos $(y, dy/dt)$. Esto genera una seria de sustituciones como sigue:

$x_1 \to y,\ x_2 \to y^2,\ y \to \Delta y,\ x_1y \to y\Delta y,\ x_2y \to y^2\Delta y,\ x_1x_2 \to yy^2 = y^3,\ x_1^2 \to y^2,\ x_2^2 \to y^4$ and $y^2 \to (\Delta y)^2$.

El modelo logístico								
x_1	x_2	y	x_1y	x_2y	x_1x_2	x_1^2	x_2^2	y^2
y	y^2	Δy	$y\Delta y$	$y^2\Delta y$	y^3	y^2	y^4	$(\Delta y)^2$
1,22	1,49	0,24	0,29	0,36	1,81	1,49	2,21	0,06
1,48	2,20	0,29	0,43	0,64	3,27	2,20	4,86	0,09
1,81	3,27	0,35	0,64	1,16	5,90	3,27	10,67	0,13
2,20	4,83	0,43	0,95	2,08	10,63	4,83	23,37	0,18
2,67	7,14	0,52	1,39	3,71	19,08	7,14	51,00	0,27
3,24	10,53	0,63	2,04	6,61	34,16	10,53	110,86	0,39
3,93	15,48	0,76	2,97	11,71	60,93	15,48	239,76	0,57
4,76	22,70	0,91	4,32	20,60	108,17	22,70	515,39	0,82
5,76	33,16	1,09	6,25	36,00	190,99	33,16	1099,87	1,18
6,95	48,24	1,29	8,98	62,35	335,02	48,24	2326,85	1,67
8,35	69,80	1,53	12,79	106,88	583,14	69,80	4871,85	2,34
10,02	100,38	1,80	18,06	180,99	1005,70	100,38	10076,05	3,25
11,97	143,32	2,11	25,23	302,07	1715,77	143,32	20540,57	4,44
14,24	202,91	2,44	34,80	495,73	2890,36	202,91	41172,10	5,97
16,87	284,48	2,80	47,30	797,77	4798,14	284,48	80927,79	7,86
19,86	394,39	3,18	63,21	1255,36	7832,19	394,39	155540,69	10,13
23,23	539,84	3,57	82,88	1925,72	12542,88	539,84	291426,89	12,72
26,99	728,47	3,94	106,37	2870,98	19661,57	728,47	530669,83	15,53
31,11	967,65	4,29	133,33	4147,45	30100,66	967,65	936342,49	18,37
35,55	1263,53	4,58	162,88	5789,69	44913,39	1263,53	1596495,76	21,00
40,25	1619,95	4,81	193,59	7791,64	65200,43	1619,95	2624222,00	23,13
45,14	2037,39	4,95	223,55	10090,59	91962,39	2037,39	4150944,95	24,53
50,12	2512,22	5,00	250,61	12561,00	125917,26	2512,22	6311225,39	25,00
55,10	3036,47	4,95	272,65	15024,13	167322,06	3036,47	9220141,93	24,48
59,99	3598,32	4,80	287,97	17273,93	215848,41	3598,32	12947873,85	23,05
64,68	4183,14	4,57	295,52	19113,44	270553,81	4183,14	17498663,26	20,88
69,10	4775,05	4,27	295,08	20390,65	329964,33	4775,05	22801110,10	18,24
73,20	5358,49	3,92	287,20	21023,29	392250,48	5358,49	28713401,81	15,39
76,94	5919,64	3,55	273,02	21006,20	455452,04	5919,64	35042106,65	12,59
80,30	6447,40	3,16	254,08	20401,77	517698,75	6447,40	41568991,16	10,01
83,27	6933,88	2,79	232,01	19319,34	577383,91	6933,88	48078720,32	7,76
85,87	7374,38	2,43	208,34	17890,86	633268,73	7374,38	54381442,78	5,89
88,13	7767,05	2,09	184,38	16249,26	684516,46	7767,05	60327017,78	4,38
90,07	8112,37	1,79	161,13	14513,03	730670,43	8112,37	65810527,93	3,20
91,72	8412,53	1,52	139,31	12777,84	771596,53	8412,53	70770732,40	2,31
93,12	8670,87	1,28	119,35	11113,99	807409,62	8670,87	75183965,26	1,64
94,29	8891,34	1,08	101,47	9567,98	838399,00	8891,34	79055909,15	1,16
95,28	9078,17	0,90	85,71	8166,26	864962,32	9078,17	82413100,63	0,81
96,10	9235,55	0,75	72,00	6919,80	887552,20	9235,55	85295321,21	0,56
96,79	9367,47	0,62	60,22	5828,46	906636,48	9367,47	87749419,31	0,39
97,35	9477,59	0,52	50,18	4884,79	922670,76	9477,59	89824680,13	0,27
97,82	9569,20	0,43	41,68	4077,02	936081,19	9569,20	91569609,06	0,18
98,21	9645,20	0,35	34,53	3391,21	947254,87	9645,20	93029880,38	0,12
98,53	9708,10	0,29	28,55	2812,82	956535,87	9708,10	94247176,94	0,08
98,79	9760,05	0,24	23,56	2327,66	964224,99	9760,05	95258669,87	0,06
99,01	9802,90	0,20	19,42	1922,50	970581,79	9802,90	96096930,46	0,04
99,19	9838,20	0,16	15,98	1585,36	975827,95	9838,20	96790114,87	0,03
99,33	9867,23	0,13	13,14	1305,66	980151,33	9867,23	97362305,09	0,02
99,45	9891,10	0,11	10,80	1074,16	983710,06	9891,10	97833925,83	0,01
99,55	9910,71	0,09	8,87	882,94	986636,53	9910,71	98222183,97	0,01
2748,92	225645,74	98,49	4959,04	329305,43	20085054,76	225645,74	1843852763,32	333,20
$\sum x_1$	$\sum x_2$	$\sum y$	$\sum x_1y$	$\sum x_2y$	$\sum x_1x_2$	$\sum x_1^2$	$\sum x_2^2$	$\sum y^2$

Estamos ajustando el modelo $\widehat{y} = a + b_1x1 + b_2x_2$, por lo que tenemos 3 incógnitas, que deben salir de 3 ecuaciones:

$\sum y = na + b_1 \sum x_1 + b_2 \sum x_2$
$\sum x_1 y = a \sum x_1 + b_1 \sum x_1^2 + b_2 \sum x_1 x_2$
$\sum x_2 y = a \sum x_2 + b_1 \sum x_1 x_2 + b_2 \sum x_2^2$

Reemplazando:

$98,49 = 50a + 2748,92b_1 + 225645,74b_2$
$4959,04 = 2748,92a + 225645,74b_1 + 20085054,76b_2$
$329305,43 = 225645,74a + 20085054,76b_1 + 1843852763,32b_2$

Podemos resolver el sistema para encontrar: $a = 0$, $b_1 = 0{,}2$, $b_2 = 0{,}002$. Hemos demostrado entonces que el mejor modelo logísitco que se ajsuta a nuestros datos es

$\frac{dy}{dt} = 0{,}2y + 0{,}002y^2$

de acá obtenemos los parámetros del modelo:

$\frac{dy}{dt} = ry\frac{(K-y)}{K} = ry - \frac{ry^2}{K} = 0{,}2y + 0{,}002y^2$

por tanto $r = 0{,}2$ y $r/K = 0{,}002$ o sea $K = 100$.

Usemos ahora las anovas para hacer un análisis de significancia del modelo de regresión hallado:

La significancia se estudiará para el modelo de regresión múltiple que predice que

$dy/dt = a + b_1 x_1 + b_2 x_2 = 0{,}2y + 0{,}002y^2$

la variación total es

$TSS = \sum (y_i - \bar{y})^2 = \sum (y_i - \widehat{y}_i)^2 + \sum (\widehat{y}_i - \bar{y})^2$

Podemos calcular con la ayuda de nuestra tablas

$NSS = \sum (y_i - \widehat{y}_i)^2$

Lo cual da cero como se esperaba.

La variabilidad explicada por el modelo es pues

$RSS = \sum (\widehat{y}_i - \bar{y})^2 = TSS$

El estadígrafo de Fisher es

$F = \frac{RSS/k}{NSS/(n-k-1)} = TSS/0 = \infty$

Esto dice que el ajuste es perfecto, tal como debe ser pues inventamos los datos para un ajuste perfecto sin haber introducido ruido alguno.

9.5. Predicción de eventos binomiales

Un evento binomial es aquel que puede venir en una de dos formas, cara o sello, si o no, desertó o nó desertó, hubo desastre o nó. A pesar de que uno no pueda predecir si un evento binomial aleatorio ocurrirá o nó, se considera relevante la posibilidad de decir con qué probabilidad puede ocurrir el evento bajo circunstancias dadas y referidas por variables explicativas cuantitativas y/o categóricas.

Hay tres modelos básicos que ligan las variables explicativas con la probabilidad de ocurrencia del evento: el primero es el lineal (la probabilidad de desastre es proporcional a ciertos factores), el segundo se relaciona con la distribución normal (Probit analysis) y el tercero con la distribución logit (Logistic regression). Se ilustra la parte interpretativa del modelo lineal y de la regresión logísitca.

Para calcular un modelo lineal, uno usa la opción GENERAL LINEAL MODELS que está ligada al análisis de varianza. En cambio, los otros dos modelos (Probit analysis, Logistic regression) son parte del análisis de regresión sobre variables categóricas o de atributos.

Se considera que el modelo lineal es machetero, pues la probabilidad está entre cero y uno y en cambio los modelos lineales usan líneas, planos y superfcies, todos ilimitados. Como consecuencia, hay que hacerse el de la vista gorda cuando se ve que los modelos predicen probabilidades negativas o mayores que uno. Con todo, los modelos lineales son muy sencillos, muy intuitivos y sin embargo muy poderosos. La diferencia con los otros dos modelos (Probit analysis, Logistic regression) es que ellos en vez de planos usan toboganes (funciones de forma sigmoidea) que encajan suavemente con el cero y con el uno, valores límites de la probabilidad.

Tratemos de entender lo básico sobre cómo se interpreta la tecnología de los modelos lineales generalizados. Para ello, estudiamos algunos ejemplos ficticios basados en el desempeño de los alumnos de unas clases de taller de escritura. Empezamos con un interés por saber qué impacto tienen el número de problemas resueltos por el estudiante y el número de horas asistidas a clase sobre la probabilidad de pasar un examen.

86 ***Ejemplo*** *Descubriendo las tendencias*

Lo que uno espera es que hay más probabilidad de que se pase el examen entre más problemas resueltos haya y más alta sea la asistencia a clase. Eso es precisamente lo que cuenta el siguiente análisis que concluye con una gráfica que indica la probabilidad de pasar como función de la dos variables explicativas *NumResueltos, HorasClase.*

```
#=============================
#TENDENCIAS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
NumResueltos <- c(1,2,3,1,5,3,2,5,7,8,12,13,15,8,10,5,10,15)
HorasClase <- c(1,3,5,1,3,5,1,3,5,1,3,5,1,3,5,1,3,5)
PasoSiNo <- c(0,0,0,0,0,1,0,1,0,0,1,0,0,1,1,0,1,1)
tabla <- data.frame(NumResueltos, HorasClase,PasoSiNo)
tabla
reg <- lm(PasoSiNo ~ NumResueltos + HorasClase)
library(scatterplot3d)
barras <- scatterplot3d(NumResueltos, HorasClase,PasoSiNo,  highlight.3d=TRUE,
type="h", main="3D Scatterplot")
barras$plane(reg)
summary(reg)
```

La gráfica describe lo que pasó y denota una tendencia: entre más se haga problemas o más se vaya a clase, mejor, es decir, mayor la probabilidad de pasar. Podemos aventurarnos a aconsejarles a los muchachos que vayan todos los días a clase y que se hagan unos 20 problemas.

¿Podemos pasar de una descripción de lo que pasó a una predicción que diga que nuestro consejo es válido para otras circunstancias semejantes?

Lamentablemente no podemos: eso nos lo da la columna de las probabilidades en la tabla de coeficientes producidas por el método *summary(reg)* en la que vemos que los coeficientes asociados a las dos variables explicativas pueden ser producidos por azar con elevadas probabilidades, 0.283 y 0.163, ambas mayores que 0.05. Eso se corrobora con el p-value de la F para la anova que pone a prueba la hipótesis nula de que no hay dependencia entre las entradas y las salidas: dicho valor es 0.1298, mayor que 0.05, lo que nos dice que esta situación es cómodamente explicada por el azar.

¿Por qué no podemos pasar de una descripción a una predicción? Hay dos causas posibles a la vista: no hay suficientes datos en relación con la dispersión de los datos y las variables explicativas son insuficientes.

87 ***Ejemplo*** *Tendencia con poder predictivo*

Para aumentar el poder predictivo, usando pocos datos, modificamos la tabla del ejemplo anterior con la idea de que se ajuste mejor a la tendencia esperada. La nueva tabla es:

```
#==============================
#TEST DE TENDENCIAS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
NumResueltos <- c(0,1,2,1,5,3,2,5,7,8,12,13,12,11,10,12,10,15 )
HorasClase <- c(1,3,4,1,3,5,1,3,5,2,3,5,0,3,5,4,3,5)
PasoSiNo <- c(0,0,0,0,1,1,0,1,1,1,1,1,0,1,1,1,1,1)
tabla <- data.frame(NumResueltos, HorasClase,PasoSiNo)
tabla
reg <- lm(PasoSiNo ~ NumResueltos + HorasClase)
library(scatterplot3d)
barras <- scatterplot3d(NumResueltos, HorasClase,PasoSiNo,  highlight.3d=TRUE,
type="h", main="3D Scatterplot")
barras$plane(reg)
summary(reg)
```

El resultado puede darse con gráficas como la mostrada en el ejemplo anterior o con un nuevo tipo de gráfica:

Diagrama de densidad que muestra la incidencia de la asistencia a clase y del trabajo personal sobre la probabilidad de pasar: hacia el rojo es mayor la probabilidad.

Acá vemos que hacer problemas o ir a clase son actividades muy rentables para esta clase. El poder predictivo de nuestra conclusión se justifica con las tablas producidas por el método *summary(reg)*:

El valor de la F nos dice que la tendencia descrita por el modelo no se da por azar, pues la probabilidad es menor al 5 %. Los valores de la t, del orden de 0.01, nos dicen que las dos variables propuestas son necesarias para tratar de explicar los resultados: lo mejor es hacer los ejercicios e ir a clase, pues uno hace muchos ejercicios para tener la oportunidad de cometer errores y va a clase para corregidos a tiempo y no después del examen.

88 ***Ejemplo*** *Añadiendo otra variable.*

Vimos que los datos de la tabla del primer ejemplo no nos permiten decir que tenemos una explicación o teoría sobre qué hay que hacer para pasar el examen. Ya vimos que una posible causa de este fracaso es la pobreza de datos en relación con la variabilidad. Otra posible causa es que falta al menos una variable a considerar. Añadimos a la tabla del primer ejemplo una tercera variable que se llama *expresión verbal* y que se mide en un test aparte produciendo notas de 0 a 5, entre más alta, mejor. La tabla queda:

```
#==============================
#AÑADIENDO OTRA VARIABLE
#Limpiar la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
NumResueltos <- c(1,2,3,1,5,3,2,5,7,8,12,13,15,8,10,5,10,15)
HorasClase <- c(1,3,5,1,3,5,1,3,5,1,3,5,1,3,5,1,3,5)
ExpresiOnVerbal <- c(3,2,2,3,3,4,3,4,4,2,3,2,2,3,5,2,3,4)
PasoSiNo <- c(0,0,0,0,0,1,0,1,0,0,1,0,0,1,1,0,1,1)
```

```
tabla <- data.frame(NumResueltos, HorasClase, ExpresiOnVerbal,PasoSiNo)
tabla
reg <- lm(PasoSiNo ~ NumResueltos + HorasClase + ExpresiOnVerbal)
library(scatterplot3d)
barras <- scatterplot3d(NumResueltos, HorasClase,
                        PasoSiNo,  highlight.3d=TRUE,
                        type="h", main="3D Scatterplot")
barras$plane(reg)
summary(reg)
```

La F, de valor 0.01, nos dice que el modelo tiene poder predictivo. La probabilidad de los coeficientes nos dice que el modelo es redundante y que podemos quitar la variable que mide la asistencia a clase:

```
#==============================
#SIMPLIFCAR MODELO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
NumResueltos <- c(1,2,3,1,5,3,2,5,7,8,12,13,15,8,10,5,10,15)
HorasClase <- c(1,3,5,1,3,5,1,3,5,1,3,5,1,3,5,1,3,5)
ExpresiOnVerbal <- c(3,2,2,3,3,4,3,4,4,2,3,2,2,3,5,2,3,4)
PasoSiNo <- c(0,0,0,0,0,1,0,1,0,0,1,0,0,1,1,0,1,1)

tabla <- data.frame(NumResueltos, HorasClase, ExpresiOnVerbal,PasoSiNo)
tabla
reg <- lm(PasoSiNo ~ NumResueltos +  ExpresiOnVerbal)
library(scatterplot3d)
barras <- scatterplot3d(NumResueltos, ExpresiOnVerbal,
                        PasoSiNo,  highlight.3d=TRUE,
                        type="h", main="3D Scatterplot")
barras$plane(reg)
summary(reg)
```

Podemos quitar también la variable que mide el trabajo en casa:

```
#==============================
#OTRA SIMPLIFICACION
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
NumResueltos <- c(1,2,3,1,5,3,2,5,7,8,12,13,15,8,10,5,10,15)
HorasClase <- c(1,3,5,1,3,5,1,3,5,1,3,5,1,3,5,1,3,5)
ExpresiOnVerbal <- c(3,2,2,3,3,4,3,4,4,2,3,2,2,3,5,2,3,4)
PasoSiNo <- c(0,0,0,0,0,1,0,1,0,0,1,0,0,1,1,0,1,1)
tabla <- data.frame(NumResueltos, HorasClase, ExpresiOnVerbal,PasoSiNo)
tabla
reg <- lm(PasoSiNo ~    ExpresiOnVerbal)
library(car)
scatterplot(PasoSiNo ~    ExpresiOnVerbal, smooth = FALSE, data = tabla)
summary(reg)
```

¿Se termina acá la investigación?
Esa pregunta se responde por el dato siguiente:
R-Squared = 0.4174

Este estadígrafo significa que el modelo explica menos de la mitad de la variabilidad de los datos. Eso implica que queda casi un 60 % por explicar. La solución es: o agregar más datos o más variables o cambiar de teoría (con muchos datos se pueden estudiar varias teorías, por ejemplo considerando interacciones y variables anidadas).

9.6. Regresión logística

Aplicar los modelos lineales a la predicción de eventos binomiales a partir de variables cuantitativas es algo que es muy sencillo y directo y si uno puede digerir resultados con probabilidades negativas o mayores que uno, entonces uno no necesita preocuparse de más. Con todo, se han producido modelos especialmente diseñados para la tarea y el más famoso de ellos es la **regresión logística**. Veamos cómo se utiliza aplicado a los datos de unas de las clases discutidas en la sección aterior:

```
#============================
#REGRESION LOGISTICA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: Modelo completo
NumResueltos <- c(0,1,2,1,5,3,2,5,7,8,12,13,12,11,10,12,10,15 )
HorasClase <- c(1,3,4,1,3,5,1,3,5,2,3,5,0,3,5,4,3,5)
PasoSiNo <- c(0,0,0,0,1,1,0,1,1,1,1,1,0,1,1,1,1,1)
Pasar <- as.factor(PasoSiNo)
tabla <- data.frame(NumResueltos, HorasClase,PasoSiNo)
tabla
library(scatterplot3d)
barras <- scatterplot3d(NumResueltos, HorasClase,PasoSiNo,  highlight.3d=TRUE,
type="h", main="3D Scatterplot")
#La regresión logística usa un modelo lineal generalizado:
#una transformación  + modelos lineales
reglog <- glm(Pasar ~ NumResueltos + HorasClase,family=binomial(), data = tabla)
summary(reglog)
#Intervalos de confianza del 95% para los coeficientes
confint(reglog)
#Coeficientes exponenciales
exp(coef(reglog))
#Intervalos de confianza del 95% para los coeficientes exponenciales
exp(confint(reglog))
#Valores predichos para el modelo sobre los datos de la tabla, 18 renglones
predict(reglog, type="response")
# Residuos: diferencias entre los predicho por el modelo y los datos
residuals(reglog, type="deviance")
#
#Parte 2:
#Modelos univariados:
reglog2 <- glm(Pasar ~ NumResueltos,family=binomial(), data = tabla)
plot(reglog2)
#La altura sobre la región negra es la probabilidad de 0
#El segmento sobre la región clara indica la prob de 1.
cdplot(Pasar ~ NumResueltos, data=tabla)
#
#Parte 3: otro univariado
reglog3 <- glm(Pasar ~ HorasClase,family=binomial(), data = tabla)
plot(reglog3)
```

```
#La altura sobre la región negra es la probabilidad de 0
#El segmento sobre la región clara indica la prob de 1.
cdplot(Pasar ~ HorasClase, data=tabla)
#Parte 4: comparar modelos
#Un vacío indica un número muy pequeño
anova(reglog,reglog2, test = "Chisq")
```

Sobre los datos dados el modelo hace lo mejor que puede pero sufre.

9.7. Regresión de Poisson

En una regresión logística se predice una respuesta que viene en una de dos formas, cara o sello, varón o niña, hubo terremoto o no hubo. La generalización a un tipo de respuesta que viene en varios eventos discretos que pueden ser más de dos se llama la regresión de Poisson.

Analicemos los datos de una clase, dado que nuestra tarea es predecir la nota, que puede tomar los valores 1,2,3,4,5 a partir de dos descriptores, el primero mide el trabajo en casa y el segundo la asistencia a clase:

```
#==============================
# REGRESION DE POISSON
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: Modelo lineal para comparar
NumResueltos <- c(0,1,2,1,5,3,2,5,7,8,12,13,12,11,10,12,10,15 )
HorasClase <-   c(1,3,4,1,3,5,1,3,5,2, 3, 5, 0, 3, 5, 4, 3, 5)
Nota <-     c(0,2,3,0,4,3,1,3,4,3, 4, 5, 4, 5, 4, 5, 5,5)
tabla <- data.frame(NumResueltos, HorasClase, Nota)
#Regresión lineal
library(scatterplot3d)
barras <- scatterplot3d(NumResueltos, HorasClase,Nota,  highlight.3d=TRUE,
type="h", main="3D Scatterplot")
regPlana <- lm(Nota ~ NumResueltos + HorasClase,family=binomial(), data = tabla)
summary(regPlana)
barras$plane(regPlana)
summary(regPlana)
#
#Parte 2: modelo de Poisson
regPoisson <- glm(Nota ~ NumResueltos + HorasClase,
                  family = poisson(), data = tabla)
summary(regPoisson)
#Valores predichos para el modelo sobre los datos de la tabla, 18 renglones
predict(regPoisson, type="response")
# Residuos: diferencias entre los predicho por el modelo y los datos
residuals(regPoisson, type="deviance")
```

9.8. ¿Cuánto viviré?

Todos creemos que por poco somos eternos hasta que los compañeros con los que uno ha compartido la vida comienzan a a morirse. Y entonces uno comienza a hacer cuentas un poco atormentadas sobre cuánto tiempo le queda a uno de vida.

Este tipo de preguntas son importantes para las compañías de seguros que deben aceptar o negar la posibilidad de adquirir pólizas. También es muy seria la situación de los médicos que con frecuencia son tomados por profetas que pueden decidir el tiempo de vida de la gente.

Examinemos un caso de la vida real, en el cual se estudia la tasa de sobrevivencia ante cancer en soldados retirados en dependencia de si ha tenido tratamiento previo o no, lo cual se codifica como variable *prior*:

```
#==============================
#TIEMPO DE VIDA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(survival)
data(veteran)
#help(veteran)
veteran
plot(survfit(Surv(time,status)~prior,data=veteran),
xlab="Años después", main = "Sobrevivencia/Tratamiento previo",
xscale=365, ylab="% sobrevivencia", yscale=100,
col=c(2,3))
#Caja de letreros: posición, contenido, lty = line types, colores
legend(1.3, 0.8, c(paste(" = ", c("No","Si"))), lty = 1, col = 2:3)
survdiff(Surv(time,status)~prior, data=veteran)
```

Acá hay otro caso relacionado con un cancer de pulmón:

```
#==============================
#TASAS DE SOBREVIVENCIA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: cargar paquete
library(survival)
#Información
#Para salir de la ayuda, teclee q de quit
#help(lung)
#
#Parte 2:
#Diagrama de dispersión
library(car)
scatterplot(lung$time~lung$age, smooth = FALSE)
#
#Análisis de sobrevivencia
# Empaquetar datos
TasaSobrev <- with(lung, Surv(time,status))

#Analizar modelos: efecto de la edad
edad <- survfit(TasaSobrev ~ age,data=lung)
summary(edad)
edad
#Efecto de la edad y de la pérdida de peso
edadPeso <- survfit(TasaSobrev ~ age + wt.loss,data=lung)
summary(edadPeso)
# Predicción de la tasa de sobrevivencia para hombres
tasaSH <- coxph(TasaSobrev ~ age+ wt.loss,
  data=lung, subset=sex==1)
summary(tasaSH)
```

Para saber más:

Lumley T (2004) The survivial package, R-news, Vol 4/1, June, pag 26.
http://cran.r-project.org/doc/Rnews/Rnews_2004-1.pdf

Mai Zhou M (2010) Use Software R to do Survival Analysis and Simulation
www.ms.uky.edu/~mai/Rsurv.pdf

Crawley M J (2010) Statistics: an introduction using R
http://www3.imperial.ac.uk/portal/pls/portallive/docs/1/1171928.PDF

Capítulo 10

Tablas de contingencia

`Relación estadística entre dos o más factores`

89 ***Objetivo.*** *Estudiar la relación de dependencia entre dos o más factores de tipo categórico.*

10.1. Ejemplos básicos

Las tablas de contingencia pueden salir de varios lados. En primera instancia, vienen de experimentos o muestreos a propósito que se registran en tablas naturales como la dd siguiente, en la cual reunimos datos con el ánimo de corroborar si efectivamente las mujeres tienen ojos más grandes que los hombres (todos adultos). Una mujer se codifica como m, un hombre como h, ojos grandes como g, ojos pequeños como p. La clasificación del tamaño de los ojos se hace por instinto.

```
#=============================
#TABLA DE CONTINGENCIA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
GEnero <- c("h","m","h","h","m","h","m","m","h","m","h","m")
Ojos <-   c("p","g","p","g","g","p","g","g","p","p","p","g")
dd <-data.frame(GEnero,Ojos)
dd
#Con dd compute la tabla de contingencia
tablaCont <- with(dd, table(GEnero,Ojos))
#Hacemos un test chi-cuadrado de independencia
chisq.test(tablaCont)
#Formas sinónimas
chisq.test(dd$GEnero, dd$Ojos)
chisq.test(GEnero, Ojos)
#Test exacto de independencia de Fihser
fisher.test(tablaCont)
```

La hipótesis nula de una tabla de contingencia es que los factores son independientes. En este caso, *GEnero* y *Ojos* son independientes. La discrepancia entre lo observado y lo predicho por la H_o se mide con un estadígrafo que se distribuye aproximadamente como una chi-cuadrado. Cuando dicha discrepancia es grande, el *p-value* es pequeño y se rechaza la H_o. Cuando la discrepancia es pequeña, el *p-value* es grande pues por azar en muchas ocasiones se consiguen eventos más extremos, y se acepta la H_o. En el presente caso, miramos la tabla de contingencia y notamos que la gran mayoría de hombres tienen ojos pequeños en tanto que la gran mayoría de mujeres los tienen grandes. Estamos diciendo que los factores *GEnero* y *Ojos* son dependientes y por tanto predecimos que debe haber una gran discrepancia entre lo

predicho por la H_o y lo observado, lo cual debe reflejarse en un *p-value* más pequeño que el crítico, que por defecto es 0.05. Lamentablemente, el p-value asociado a nuestros datos es 0.0836 que es pequeño pero no lo suficiente: si recolectmos más datos, lo más seguro es que logremos probar lo que queremos.

Para tablas 2×2 se puede usar el test de Fisher que es exacto. Su H_o es que las proporciones son iguales y que por tanto la relación entre ellas es uno.

Conviene recordar que uno puede caer en un problema si codifica categorías con números, como en el caso en el cual una mujer se codifica como 1 y un hombre como 0. En un caso así, se le dice a R que los números son realmente códigos de categorías:

```
#==============================
#CODIFICACION Y  TABLAS DE CONTINGENCIA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Codificación numérica
GEnero <- c(0,1,0,0,1,0,1,1,0,1,0,1,0,1,0)
#Los números no son números sino códigos de un factor
GEneroF <- as.factor(GEnero)
#Longitud horizontal de los ojos:
Ojos <-   c(4,7,4,5,5,4,7,8,4,7,7,7,4,8,5 )
#Partimos Ojos en dos categorías
#que ofrezcan el máximo contraste
Ojos.cat<-cut(Ojos,breaks =2)
#Calculamos la tabla de contingencia
d<-table(Ojos.cat, GEneroF)
#Publicamos la tabla
d
chisq.test(Ojos.cat, GEneroF)
```

Podemos usar procedimiento sinónimos, como por ejemplo:

```
#==============================
#TABLA DE CONTINGENCIA: VARIANTE
#Datos numéricos
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
GEnero <- c(0,1,0,0,1,0,1,1,0,1,0,1,0,1,0)
Ojos <-   c(4,7,4,5,5,4,7,8,4,7,7,7,4,8,5 )
#Partimos GEnero en dos categorías
#que ofrezcan el máximo contraste
GEnero.cat<-cut(GEnero,breaks =2)
#Partimos Ojos en dos categorías
#que ofrezcan el máximo contraste
Ojos.cat<-cut(Ojos,breaks =2)
#Calculamos la tabla de contingencia
d<-table(Ojos.cat, GEnero.cat)
#Publicamos la tabla
d
chisq.test(Ojos.cat, GEnero.cat)
```

¿De qué sirve obtener resultados si la audiencia no los puede digerir? De pura vanagloria. Lo que sucede es que ninguna audiencia resiste una lluvia de datos e indicadores numéricos. En cambio, todas las audiencias celebran informes gráficos bien logrados y estratégicos. La solución que *R* propone es tener los indicadores numéricos a mano pero ofrecer todas las opciones gráficas que se requieran. Para el caso de las tablas de contingencia, tenemos la siguiente opción:

```
#==============================
#GRAFICAS DE TABLAS DE CONTINGENCIA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
GEnero <- c(0,1,0,0,1,0,1,1,0,1,0,1,0,1,0)
#Los números no son números sino códigos de un factor
GEneroF <- as.factor(GEnero)
Ojos <-   c(4,7,4,5,5,4,7,8,4,7,7,7,4,8,5 )
#Partimos Ojos en dos categorías
#que ofrezcan el máximo contraste
Ojos.cat<-cut(Ojos,breaks =2)
#Calculamos la tabla de contingencia
d<-table(Ojos.cat, GEneroF)
#Publicamos la tabla
d
#Representación gráfica de la tabla de contingencia:
#la mayoría de mujeres tienen ojos grandes,
#la mayoría d ehombres ojos pequeños
library(vcd)
mosaic(d, main = "0=mujer, 1 = hombre,
       \n (4,6] =ojos pequeños, (6,8] = ojos grandes" )
chisq.test(Ojos.cat, GEneroF)
```

La gráfica nos muestra que los de un tipo de género se asocian con un tipo de tamaño de ojos, y que por lo tanto, los factores son independientes. Mirando la tabla y su codificación, uno puede entonces concluir: las mujeres tienen lo ojos grandes y los hombres los tienen pequeños.

Una telenovela más realista nos invita a imaginar que ojos grandes o chicos es una clasificación que realmente se hace en relación a la altura de la persona. Una situación así podría arrojar datos como los siguientes:

```
#==============================
#TABLA TRIDIMENSIONAL
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Partición automática
GEnero <- c(0,1,0,0,1,
            0,1,1,0,1,
            0,1,0,1,0)
#Los números no son números sino códigos de un factor
GEneroF <- as.factor(GEnero)
Ojos <-   c(4,7,4,5,5,
            4,4,8,4,7,
            7,7,4,8,5 )
Altura <- c(170, 165, 157, 171, 170,
            170, 160, 154, 158, 148,
```

```
            158, 158, 180, 180, 182)
tablaDatos <-data.frame(GEnero,Ojos,Altura)
tablaDatos
#Tabla de contingencia multidimensional
#muchas categorías, difícil de digerir.
#ftable = flat table = representación plana
#de una tabla multidimensional:
ftable(tablaDatos)
#Para hacer algo más digestible:
#Partimos Ojos en dos categorías
#que ofrezcan el máximo contraste
Ojos.cat<-cut(Ojos,breaks =2)
#Partimos Altura en dos categorías
#que ofrezcan el máximo contraste
Altura.cat<-cut(Altura,breaks =2)
#Calculamos la tabla de contingencia
d<-table(Ojos.cat, GEneroF, Altura.cat)
#Publicamos la tabla multidimensional digerible:
d
ftable(d)
#Hacemos un test chi-cuadrado de independencia
chisq.test(d)
#Representación gráfica de la tabla de contingencia,
#Es una variación gráfica de la ftable.
library(vcd)
mosaic(d )
```

El p-value asociado a los datos es de 0.13, demasiado grande, lo cual dice que hay una discrepancia pequeña entre lo predicho por la H_o y lo observado: no tenemos datos suficientes para probar con este test que los factores Ojos, GEnero y Altura puedan tener alguna dependencia estadísticamente significativa.

Por otra parte, la gráfica contiene en dos dimensiones información sobre tres variables. Para lograrlo, se vale de una representación de árbol: el factor ojos tiene dos categorías, ojos pequeños y ojos grandes. la categoría de ojos pequeños viene subdividida en dos opciones: bajitos y altos, y lo mismo la categoría de ojos grandes. A su vez, los ojos pequeños y bajitos pueden ser hombre (0) o mujer (1). Utilicemos los códigos siguientes:

Ojos: (4,6] =pequeños = p, (6,8] = grandes = g

GEnero: 0=mujer = m, 1 = hombre = h,

Altura: (148,165] = bajitos = b, (165,182] = altos = a

Vista con esa codificación, la gráfica tiene 8 rectángulos que reflejan las frecuencias absolutas de cada una de las 8 posibles combinaciones, a saber:

(Ojo,GEnero,Altura)	
(p,h,b)	(p,m,b)
(p,h,a)	(p,m,a)
(g,h,b)	(g,m,b)
(g,h,a)	(g,m,a)

Esta gráfica se lee conjuntamente con el output de la instrucción *ftable(d)*: la mayoría de hombres tiene ojos pequeños y son altos. Y en cambio, la mayoría de mujeres son bajitas y tienen los ojos grandes. Pero nos faltan datos para probar estas asociaciones.

Dejamos a R la tarea de particionar los rangos como a él le pareció conveniente. Si uno prefiere hacerlo a su propio estilo, se puede hacer siguiendo el ejemplo siguiente:

```
#============================
#TABLA TRIDIMENSINAL CUANTITATIVA
#Datos numéricos, partición personalizada
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
GEnero <- c(0,1,0,0,1,
            0,1,1,0,1,
            0,1,0,1,0)
#Los números no son números sino códigos de un factor
GEneroF <- as.factor(GEnero)
Ojos <-   c(4,7,4,5,5,
            4,4,8,4,7,
            7,7,4,8,5 )
Altura <- c(170, 165, 157, 171, 170,
            170, 160, 154, 158, 148,
            158, 158, 180, 180, 182)
tablaDatos <-data.frame(GEnero,Ojos,Altura)
tablaDatos
#Tabla de contingencia multidimensional
#en clases apropiadas
dd <- with(tablaDatos,ftable(GEnero, OjosGrandes= Ojos > 6,
                                      Altos  = Altura > 175))
dd
#Hacemos un test chi-cuadrado de independencia
chisq.test(dd)
```

Con la nueva forma de definir las categorías, uno se queda con la debil impresión de que los hombres tienen ojos pequeños y son bajitos, y que las mujeres tienen ojos grandes y son bajitas.

90 *Relaciones funcionales*

Debido a que tenemos datos sobre la altura y el tamaño de los ojos, tenemos una estructura de datos adecuada para alegar que tener ojos grandes o chicos es algo que se decide en términos relativos a la altura: ¿Nos permitirá este nuevo punto de vista probar que las mujeres tienen ojos grandes en comparación con los hombres?

```
#============================
#TABLAS Y RELACIONES FUNCIONALES
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
GEnero <- c(0,1,0,0,1,
            0,1,1,0,1,
            0,1,0,1,0)
#Los números no son números sino códigos de un factor
GEneroF <- as.factor(GEnero)
Ojos <-   c(4,7,4,5,5,
            4,4,8,4,7,
            7,7,4,8,5 )
Altura <- c(170, 165, 157, 171, 170,
            170, 160, 154, 158, 148,
            158, 158, 180, 180, 182)
ojos <- Ojos/Altura
```

```
#Partimos ojos en dos categorías
#que ofrezcan el máximo contraste
ojos.cat<-cut(ojos,breaks =2)
tablaCont <-table(GEnero, ojos.cat)
tablaCont
#Hacemos un test chi-cuadrado de independencia
chisq.test(tablaCont)
#Test exacto de independencia de Fihser
fisher.test(tablaCont)
```

Mientras que el test chi-cuadrado no nos sirvió, el de Fisher sí: las mujeres tienen ojos grandes y los hombres los tienen pequeños. En realidad, el test de Fisher nos dice que hay una desvación de lo esperado, que las proporciones fuesen iguales. Pero para saber a qué lado, hay que mirar la tabla: hay 7 hombres de ojos pequeños en contra de 1 con ojos grandes, y hay 2 mujeres con ojos pequeños, en contra de 5 de ojos grandes.

10.2. Asociaciones

La pregunta natural para una tabla de contingencia es si sus factores son o no independientes. Si uno llegase a tener datos suficientes para decir que no lo son, entonces podrían surgir varias preguntas acerca de la naturaleza de las asociaciones existentes entre los diversos factores considerados. Ventilemos varios tópicos al respecto.

91 ***Acuerdo vs desacuerdo***

En una clase de ciencias se les pregunta a parejas formadas por un niño y su mejor amiguita sobre si les gustaría prepararse para ir al planeta Marte. Primero se explica el costo para cada uno, los beneficios y los riesgos. Luego se les da a escoger entre 3 opciones: I = estoy interesado, C = siento curiosidad, N = no me interesa. La tabla de **datos apareados** (niño, mejor amiguita) es la siguiente:

```
#==============================
#TEST ACUERDO-DESACUERDO PARA DATOS APAREADOS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
Varones    <- c("I", "I", "C", "I", "N", "I", "C", "I", "I", "C")
Amiguitas <-  c("C", "C", "I", "N", "N", "C", "N", "C", "C", "N")
datos <- table(Varones, Amiguitas)
datos
library(vcd)
K <- Kappa(datos)
K
confint(K)
#Diagrama acuerdo-desacuerdo:
acuerdo <- agreementplot(datos, main="Quiero ir a Marte")
unlist(acuerdo)
#Hacemos un test chi-cuadrado de independencia
chisq.test(datos)
fisher.test(datos)
```

la siguiente variante podría ser de ayuda:

```
#==============================
#TEST ACUERDO-DESACUERDO PARA DATOS APAREADOS
```

```
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
Varones    <- c("I", "I", "C", "I", "N", "I", "C", "I", "I", "C")
Amiguitas <-  c("C", "C", "I", "N", "N", "C", "N", "C", "C", "N")
datos <- table(Varones, Amiguitas)
datos
library(vcd)
(K <- Kappa(datos))
K
confint(K)
#Diagrama acuerdo-desacuerdo:
pushViewport(viewport(layout = grid.layout(ncol = 2)))
pushViewport(viewport(layout.pos.col = 1))
acuerdo <- agreementplot(t(datos), main = "Varones vs amiguitas",
                  newpage = FALSE)
unlist(acuerdo)
#Hacemos un test chi-cuadrado de independencia
chisq.test(datos)
fisher.test(datos)
popViewport()
#Contraste: Varones vs Varones = total acuerdo.
totalAcuerdo <- table(Varones, Varones)
pushViewport(viewport(layout.pos.col = 2))
agreementplot(t(totalAcuerdo), main = "Varones vs varones",
                  newpage = FALSE)
popViewport(2)
dev.off()
#assocstats(acuerdo)
```

La función Kappa() del paquete vcd fue usada para medir cuantitativamente el grado de acuerdo. Sirve para tablas de dos vías con el mismo número de categorías. ASE = asymptotic standard error. Si las categorías son ordinales, use kappa con pesos (weighted). No hay acuerdo sobre la interpretación de los valores de kappa. Una regla cualitativa simple es: entre más grande y cercano a uno más acuerdo, entre menor sea con respecto a uno, más desacuerdo. Otra más complicada: arriba de 0.75 el acuerdo es excelente, ente 0.50 y 0.75 como bueno, y abajo de 0.50 como bajo.

10.3. Modelos lineales generalizados y loglineales

Las diversas asociaciones entre factores pueden estudiarse por medio de modelos lineales generalizados, generalazed linear models: glm(). Los resultados y forma de interpretación coinciden con el análisis por medio de modelos loglineales, el cual es más popular.

Para calcular la hipótesis nula de una tabla de contingencia se parte del supuesto de independencia entre factores, lo cual predice que la probabilidad de dos eventos A y B es

$p(AB) = p(A)p(B)$

Esto nos lleva a estudiar la forma como se distribuyen los productos, lo cual ha resultado tremendamente difícil y no tenemos más que aproximaciones. Una salida nos invita a considerar la idea de tomar logaritmos:

$lg(p(AB)) = log(p(A)) + log(p(B))$.

Esta idea ha resultado muy interesante y muy fructífera y se conoce como **modelos loglineales**.

La forma de programar *R* y de interpretar los resultados que arroje son los estándares para los modelos lineales generalizados. Para fijar ideas, consideremos una tabla de 3 vías: A,B,C. Las diversas interacciones posibles aparecen en la tabla siguiente (Friendly, 2010):

Cuadro 10.1: Interpretación de modelos lineales generalizados y loglineales

Modelo	Fórmula	Símbolo	Interpretación
Independencia mutua	$\sim$ A + B + C	[A][B][C]	A $\perp$ B $\perp$ C
Independencia conjunta	$\sim$ A*B + C	[AB][C]	(A B) $\perp$ C
Independencia condicional	$\sim$ (A+B)*C	[AC][BC]	(A $\perp$ B) | C
Asosc. de dos vías	$\sim$ A*B + A*C + B*C	[AB][AC][BC]	Asociación homogénea
Modelo saturado	$\sim$ A*B*C	[ABC]	Asociación de 3 vías
Símbolos	$\sim$ = fórmula del modelo	$\perp$ = es indep. de	| = dado (condicional)

92 ***Ejemplo: los pollitos***

Supongamos que queremos estudiar el efecto de tres variables *temp*(temperatura), *luz* (iluminación) y *alimento* sobre el crecimiento de unos pollitos en un sistema de incuvadoras. La variable *temp* viene en tres niveles: alta, media, y baja. La variable *luz* puede tomar dos valores: *buena, mala*. La variable alimento puede tomar 4 valores: *natural, concentrado1, concentrado2, mixto*. Para hacer el experimento, armamos incuvadoras con todas las posibles opciones y al cabo de 2 semañas pesamos los pollitos para estimar el crecimiento.

Lo único que nos interesa en la presente etapa de investigación es saber si la variable crecimiento está ligada a alguna de las variables experimentales, dados los niveles tomados. Para dilucidar el interrogante usamos la metodología de los **modelos loglineales** que se aplican a **tablas de contingencia multidimensionales**, con varios factores y varios niveles pero con una variable respuesta que ha de ser cuantitativa (en nuestro ejemplo, peso del pollito al cabo de dos semanas de nacido).

Para hacerle llegar los datos a *R*, lo hacemos a través de la codificación **gl** que provee una función que genera factores por medio de la especificación del patrón de sus niveles. La tabla de datos debe tener $3 \times 2 \times 4 = 24$ renglones.

```
#============================
#MODELOS LOGLINEALES
#Limpia la memoria
rm(list = ls())
#Generamos los factores con sus niveles:
tempN <- c("alta","media","baja")
luzN <- c("Buena","Mala")
alimentoN <- c("natural","concentrado1", "concentrado2", "mixto" )
#Generamos las columnas de datos
#Alimento tiene 4 niveles, queremos que se liste 1 x 1:
alimento <- gl(4,1,24,alimentoN)
#Luz tiene 2 niveles, cada uno debe repetirse 4 veces,
#una vez por cada nivel de alimento:
luz      <- gl(2,4,24,luzN)
#temp tiene 3 niveles, cada uno debe repetirse 2 x 4 = 8 veces,
#una vez por cada pareja de niveles alimento x temperatura:
temp     <- gl(3,8, 24, tempN)
#Después de hacer el experimento,  los datos son
peso <- c(1.1, 1.2, 1.1, 1.5,
         0.9, 1.0, 1.1, 1.6,
         1.1, 1.2, 1.0, 1.4,
```

```
         1.1, 1.2, 1.3, 1.4,
         1.0, 1.1, 1.1, 1.4,
         1.0, 1.0, 1.2, 1.3)
#Pegamos los datos en una tabla
pollitos <- data.frame(temp, luz, alimento, peso)
pollitos
library(MASS)
solos <- loglm(peso ~  temp + luz + alimento, data = pollitos)
summary(solos)
inter2 <- loglm(peso ~ temp:luz + temp:alimento + luz:alimento)
summary(inter2)
inter3 <- loglm(peso ~ temp:luz:alimento)
summary(inter3)
saturado2 <- loglm(peso ~ temp + luz + alimento
                        + luz + temp : alimento + luz:alimento)
summary(saturado2)
saturado3 <- loglm(peso ~ temp + luz + alimento
          + luz + temp : alimento + luz:alimento + temp:luz:alimento)
summary(saturado3)
anova(solos, inter2)
anova(solos,saturado2,saturado3)
anova(solos, inter2,saturado2)
drop1(saturado3,test="Chisq")
coef(saturado3)
```

Para saber más:

```
Quinn K (2010) R/S-PLUS example: loglinear models for 2-way and 3-way tables
http://www.stat.washington.edu/quinn/classes/536/S/loglinexample.html

 Friendly M (2010) Working with categorical data with R and the vcd
and vcdExtra packages
www.math.yorku.ca/SCS/Courses/VCD/vcd-tutorial.pdf

 Darlington R (2010)
Measures of association in crosstab tables
http://comp9.psych.cornell.edu/Darlington/crosstab/TABLE0.HTM
```

10.4. Análisis de correspondencia

Es una forma gráfica de estudiar tablas de correspondencia: si dos elementos están cerca, sus comportamientos son semejantes. Pero una lejanía no representa una diferenciación proporcional sino una no proporcional.

Se requiere instalar el paquete ca (correspondence analysis).

Para saber más:

```
 de Leeuw J, p Mair (2010) Simple and Canonical Correspondence Analysis
Using the R Package anacor
http://cran.r-project.org/web/packages/anacor/vignettes/anacor.pdf


Bee- Leng Lee (2010) correspondence analysis
http://forrest.psych.unc.edu/research/vista-frames/pdf/chap11.pdf
```

Capítulo 11

Correlación canónica

```
Primero sintetice, luego relacione
```

93 ***Objetivo.*** *Estudiamos una metodología que combina la síntesis con la correlación*

Hay preguntas muy delicadas de la vida real y que pueden estudiarse al menos parcialmente con la tecnología de correlación canónica:

11.1. Combinaciones o contrastes

De todo lo que se les eseña a los estudiantes, ¿qué es lo importante? A una pregunta tan complicada podría haber muchas respuestas tentativas. Una simple y prometedora es la siguiente: los estudiantes deben salir con un buen manejo tanto del lenguaje como de la abstracción, es decir, deber salir buenos en matemáticas y lenguaje. Si codificamos esta propuesta desde la perspectiva de los modelos lineales, estamos diciendo que hay una macrovariable llamada matemáticas + lenguaje, la cual debe tener un alto poder predictivo sobre el desempeño futuro de los estudiantes.

Pero mirando bien, ¿qué significa esa macrovariable, matemáticas + lenguaje? ¿Es simplemente una invitación a hacer un estudio de regresión múltiple con esas dos variables explicativas? Evidentemente, que esa sería una forma de interpretarlo. Pero existe otra forma de hacerlo y que parte de una sospecha operacional incisiva: ¿qué pasaría si reemplazáramos la tal macrovariable con un promedio aritmético de los puntajes en las dos materias? ¿ Pero es acaso que el promedio la única forma de **contrastar** o combinar matemáticas y lenguaje? ¿No sería más sabio decir que el éxito se construye sobre el 80 % de lenguaje y 20 % de matemáticas? ¿Y qué del resto de las posibles combinaciones? Y entre todas las forma de combinar, ¿con cuál criterio vamos a escoger la que nos sirve?

94 ***Ejemplo.*** *El diseño de un perfume*

Supongamos que estamos desarrollando un perfume y nos preocupa gastar los grandes capitales que algunas empresas familiares necesitaron invertir a través de varios siglos para desarrollar marcas, sea en Francia, Ingalterra, Egipto o Arabia, y que ahora las hacen tan famosas.

El diseño comienza haciendo una lista de materiales olorosos: flores, hojas y raíces de hiervas y plantas, cáscaras de semillas, frutas y troncos de árboles, maderas, partes de animales, productos químicos. Segundamente hacemos una lista de las propiedades olorosas, como la siguiente:

- Delicado, fuerte.
- Insinuante, directo
- Pasional, sensual.
- Limpio, santo.
- Purificante, trastornante.

- Majestuoso, cursi.
- Fresco, acalorante.
- Dulce, agrio, insidioso.
- Feminidad, masculinidad.
- Salvaje, depurado, fino.
- Juventud, seriedad.

En tercer término hacemos una lista asociativa en la cuál aparecen los productos olorosos con sus propiedades. Por ejemplo, que el aceite de pino es fuerte, limpio, majestuoso, masculino, serio.

El cuarto punto es reconcer que cada uno de los materiales naturales tiene varios componentes olorosos. Por ejemplo, cada persona tiene su olor característico, lo cual se deriva de las sustancias que expele la piel que pueden clasificarse en más de 40 tipos diferentes. La consecuencia de usar productos totalmente naturales es que uno no puede aislar un perfil. ¿Qué quiere decir esto?

Eso significa que si alguien desea confeccionar un perfume que sea fuerte, limpio, majestuoso, masculino, serio entonces debe pensar en aceite de pino (los cuales son distintos para cada especie). Pero el problema es que si nos untamos el cuello con aceite de pino tal cual es, a lo mejor alguien nos va a tomar por capos recién enriquecidos que sueñan con incrustarse en la clase alta sin tener el sabor aristocrático debido.

Ese durísimo trabajo de separar los productos olorosos en compuestos químicos puros, en esencias, lo hacen las empresas químicas y farmacéuticas. Ellas también se preocupan de modificar sus esencias para que tengan además propiedades industriales deseables, como por ejemplo, que no se descompongan en menos de un año. La industria también procura confeccionar métodos industriales de síntesis para no tener que esperar los 30 años que un árbol puede necesitar para crecer.

El problema del diseño de perfumes es entonces unir un perfil con una mezcla de esencias. Por ejemplo, la artista Shakira eligió el perfil que se define como: aromático, dulce, fresco, luminoso, aterciopelado, sensual y femenino. Para resolver el problema de qué mezclas producían el perfil designado, ella mezcló (en proporciones secretas) vainilla, jazmín y sándalo. En sus propias palabras podemos verlo muy claramente:

"Hemos creado una fragancia que traduce todos mis gustos en el mundo olfativo. Reúne los aromas que más me gustan: tiene vainilla, que es el punto dulce, jazmín, es la frescura y luminosidad porque quería un perfume fresco y aterciopelado, y sándalo, que simboliza la sensualidad, la feminidad".

Es preciso ser realistas y nos conviene tener claro que algunas damas olerán una muestra de nuestro perfume y de una vez y para siempre decidirán no comprarlo pues les fastidia la idea de oler a camarera de hotel de 5 estrellas.

Lo que estamos diciendo es que aíslar un perfil, que lo distinga y caracterice de todo lo demás, es una tarea muy difícil y muy artística. Si le interesa el tema de los perfumes, pruebe con sustancias antagónicas en cantidades subliminales y después invierta millonadas en propaganda por canales de covertura mundial. La inversión en propaganda es necesaria porque, como lo hemos insinuado, la idea de perfil no es algo que dependa únicamente del producto sino del contexto social en que se consuma y que puede ser muy particular. El objetivo de la propaganda es, disculpen la sinceridad, lavar el cerebro de la gente para forzar una unificación de criterios en una dirección deseable para el negocio, por ejemplo, distinción y exclusividad.

Como vemos, la vida real es tremendamente compleja. La estadística nos ayuda en el diseño de perfiles cuando es posible ignorar el efecto del contexto y podemos restringirnos a los datos asociados al producto.

El diseño de perfiles se comienza a estudiar en estadística por medio de la teoría de **análisis de correlación canónica**. Su punto de partida es sacarle el máximo provecho a los datos, tomándolos como un todo y no por partes, pues los datos pesan más juntos que separados. Su propósito y metodología es la siguiente:

Primero se establace cuál es el conjunto de variables dependientes y cuál el de variables independientes. Después se encuentra U1, la forma de combinar las variables independientes y, aparte, V1, las independientes de tal forma que se cree la máxima correlación posible entre dichas combinaciones. Este procedimiento produce lo que podría llamarse la primera asociación de variables canónicas (first canonical variates), U1 con V1. Después se forma la segunda asociación canónica que también busca una máxima correlación pero con la restricción de tener zero correlación con U1 y V1. Y así sucesivamente.

No se acostumbra a ser muy riguroso con el análisis de correlación canónica pues casi nunca se cumple el supuesto de normalidad multidimensional. Se sobreentiende que un análisis tan fino no vale la pena para proyectos con unos pocos datos.

La implementación en *R* de la correlación canónica no viene con el paquete básico, sino que viene en un paquete adicional que se usa conjuntamente con otros que deben bajarse de la red. Necesitamos aprender a bajar e instalar paquetes.

95 ***Instalación de paquetes***

La CRAN (la casa madre de R) tiene paquetes para todo lo que uno se imagine, especialmente para hacer las cosas mucho mejor, y para mil cosas más. Hay alrededor de 2500 paquetes para escoger en la siguiente dirección:

```
cran.r-project.org
```

Para uno adueñarse de un paquete, primero se baja e instala el paquete escogido conjuntamente con sus dependencias. Luego viene la activación que debe hacerse en cada sesión y puede hacerse por comandos en la consola o por medio del *R-comander* en *tools* → *load packages*.

Para bajar paquetes sobre Windows:
se usa el menu de R: si el paquete aparece listado, se procede por inercia. Si el paquete no aparece listado, se baja la versión para Windows desde la CRAN. Para ello, uno busca *cran* en cualquier motor de búsqueda, digamos Google, y de allí se conecta a la página web de *R*. Sobre dicha página, uno busca un link que lo lleve a los paquetes (packages). Siguiendo dicho link, uno llega a una lista alfabética de paquetes y allí busca el paquete deseado. Al buscar, hay que tener presente que la lista distingue mayúsculas de minúsculas tanto en la primera letra como en las demás y que los números van primero que las letras. Uno puede examinar la lista de paquetes con una descripción de media línea sobre su función.
Cuando uno haya bajado el paquete y lo haya puesto en alguna carpeta, vuelve a usar el menu de R para instalarlo. Y después ya lo puede correr.

Para bajar e instalar paquetes en Linux:
Primero que todo, uno busca *cran* en cualquier motor de búsqueda, digamos Google, y de allí se conecta a la página web de *R*. Sobre dicha página, uno busca un link que lo lleve a los paquetes (packages). Siguiendo dicho link, uno llega a una lista alfabética de paquetes y allí busca el paquete deseado. Al buscar, hay que tener presente que la lista distingue mayúsculas de minúsculas tanto en la primera letra como en las demás. Los números van primero que las letras. La lista de paquetes viene con una descripción de media línea sobre su función.
Para instalar un paquete en Linux: cuando uno haya bajado y salvado su archivo en una carpeta apropiada, uno abre una terminal encima de dicha carpeta y con derechos *su* (superuser), uno teclea el comando de instalación.
Para fijar ideas, pensemos en montar la maquinaria para correr correlación canónica sobre *R*. Lo primero que hacemos es instalar el gcc-fortran, que normalmente viene con todas las distribuciones de Linux. Para ello se abre una terminal de Linux, se teclea *su* (superuser) se da la clave y se copia+pega el siguiente comando sobre la terminal:

```
zypper install gcc-fortran
```

Una vez que se ha instalado gcc.fortran, uno ya puede bajar e instalar los iguientes paquetes:
fda, spam, fields, catspec, CCA

Una vez bajados y puesto en la carpeta downloads, abrimos una terminal encima de downloads, tecleamos *su*, y la clave y después usamos copie+pegue sobre los siguientes comandos, en el orden respectivo (por favor, cambiar las versiones, las cuales vienen con el nombre de cada archivo):

```
R CMD INSTALL fda_2.2.2.tar.gz
R CMD INSTALL spam_0.22-0.tar.gz
R CMD INSTALL fields_6.01.tar.gz
R CMD INSTALL catspec_0.95.tar.gz
R CMD INSTALL CCA_1.2.tar.gz
```

Una vez que uno ha montado todos estos paquetes, uno ya puede correr el demo que está más abajo. El objetivo puede ser verificar que uno tiene todo lo que necesita y en dicho caso uno simplemente revisa que no haya un sólo mensaje de error.

```
#TEST DE LIBRERIAS
library(fda)
library(spam)
library(fields)
library(catspec)
library(CCA)
```

Muchos paquetes son grandes proyectos que se van haciendo poco a poco y que reciben mejoras con frecuencia. Para actualizar los paquetes, *R* tiene la opción de actualizarlos todos con una sóla orden:

```
update.packages()
```

96 *Demo de correlación canónica*

El demo es el siguiente:

```
#==============================
#DEMO SOBRE CORRELACION CANONICA
#Autor: UCLA: Academic Technology Services, Statistical Consulting Group.
#www.ats.ucla.edu/stat/R/dae/canonical.htm
#Los comentarios han sido fuertemente simplificados.
#
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
# Leer datos
mm <- read.table("http://www.ats.ucla.edu/stat/R/dae/mmreg.csv",
                                        sep = ",", header = TRUE)
attach(mm)
library(fields)
#
#Estadística básica
t(stats(mm))
#
library(catspec)
#Tablas de porcentaje
ctab(table(female), addmargins=TRUE)
#
```

```
#Objetivo del analisis de correlación canónica:
#hallar las correlaciones canónicas entre el conjunto de
#variables independientes = psych (de tipo sicológico) y el de
#variables dependientes = acad (de rendimiento académico)
#columnas 1,2,3:
psych<-mm[,1:3]
#columnas 4,5,6,7,8.
acad<-mm[,4:8]
#
# Correlaciones ordinarias
library(CCA)
matcor(psych,acad)
#
#Calcule las correlaciones canónicas =
#raíz cuadrada de los valores propios.
cc1 <- cc(psych,acad)
#
# Muestre las correlaciones canónicas:
#En este caso hay 3 funciones, las cuales determinan cada unamáxima
#un perfil, una función entre una combinación de las variables de psych
#y las variables de acad.
cc1[1]
#
#Muestre las combinaciones para cada uno de los perfiles,
#tanto en el conjunto de variables dependientes
#como en el de variables dependientes
cc1[3:4]
#
#Compute  las 'canonical loadings'=
#correlaciones entre las variables ordinarias (X,Y)
#y las canónicas (U =xscores,V = yscores).
cc2<-comput(psych, acad, cc1)
#
# Muestre las 'canonical loadings'.
# Hay 4 por todas:
# (X,U), (Y,U), (X,V), (Y,V):
cc2[3:6]
#
# Test sobre las dimensiones  canónicas:
# De las 3 funciones canónicas,
#calcule  cuáles son significativas, con p-value menor que 0.005:
ev<-cc1$corˆ2
ev2<-1-ev
n<-dim(psych)[1]
p<-length(psych)
q<-length(acad)
m<-n -3/2 - (p+q)/2
w<-cbind(NULL)  # initialize wilks lambda

for (i in 1:3){
          w<-cbind(w,prod(ev2[i:3]))
        }

d1<-cbind(NULL)
d2<-cbind(NULL)
```

```
f<-cbind(NULL)  # initialize f
for (i in 1:3){
          s<-sqrt((pˆ2*qˆ2-4)/(pˆ2+qˆ2-5))
          si<-1/s
          df1<-p*q
          d1<-cbind(d1,df1)
          df2<-m*s-p*q/2+1
          d2<-cbind(d2,df2)
          r<-(1-w[i]ˆsi)/w[i]ˆsi
          f<-cbind(f,r*df2/df1)
          p<-p-1
          q<-q-1
      }

pv<-pf(f,d1,d2,lower.tail=FALSE)
dmat<-cbind(t(w),t(f),t(d1),t(d2),t(pv))
colnames(dmat)<-c("WilksL","F","df1","df2","p")
rownames(dmat)<-c(seq(1:length(w)))
#
#Liste el p-value de   las funciones canónicas:
#En la primera línea se hace un test sobre la
#significancia de todo el conjunto con las 3 variables canónicas.
#En la segunda, se estudia la significancia de todo lo que queda aparte
#de la primera, es decir, segunda y tercera función.
#En la tercera, se estudia la significancia de lo que queda sin contar
#las primeras dos dimensiones: queda la tercera,
#que  no es significativa.
dmat
#
# Coeficientes canónicos estandarizados para psych:
#El coeficiente [1,1] da -0.8404196
#significa que  un aumento en  la variable locus_of_control
#de una desviación estándard causa una disminución de 0.84
#desviaciones estándard de la primera variable canónica de psych
#cuando las otras variables permanecenconstantes
sd<-sd(psych)
s1<-diag(sd)       # diagonal matrix of psych sd's
s1 %*% cc1$xcoef
#
#Coeficientes canónicos estandarizados para acad:
#El coeficiente [1,1] da -0.45080116
#significa que  un aumento en  la variable read
#de una desviación estándard causa una disminución de 0.45
#desviaciones estándard de la primera variable canónica de acad
#cuando las otras variables se conservan constantes.
sd<-sd(acad)
s2<-diag(sd)
s2 %*% cc1$ycoef
```

El análisis dado por los autores de todo el estudio es el siguiente:

Table 1: Tests of Canonical Dimensions

Dimension	Canonical Corr.	F	df1	df2	p

```
    1         0.46     11.72   15   1634.7  0.0000
    2         0.17      2.94    8   1186    0.0029
    3         0.10      2.16    3    594    0.0911

Table 2: Standardized Canonical Coefficients
                            Dimension
                           1         2
Psychological Variables
    locus of control    -0.84     -0.42
    self-concept         0.25     -0.84
    motivation          -0.43      0.69
Academic Variables plus Gender
    reading             -0.45     -0.05
    writing             -0.35      0.41
    math                -0.22      0.04
    science             -0.05     -0.83
    gender (female=1)   -0.32      0.54
```

Tests of dimensionality for the canonical correlation analysis, as shown in Table 1, indicate that two of the three canonical dimensions are statistically significant at the .05 level. Dimension 1 had a canonical correlation of 0.46 between the sets of variables, while for dimension 2 the canonical correlation was much lower at 0.17. Table 2 presents the standardized canonical coefficients for the first two dimensions across both sets of variables. For the psychological variables, the first canonical dimension is most strongly influenced by locus of control (.84) and for the second dimension self-concept (-.84) and motivation (.69). For the academic variables plus gender, the first dimension was comprised of reading (.45), writing (.35) and gender (.32). For the second dimension writing (.41), science (-.83) and gender (.54) were the dominating variables.

```
UCLA (2010) R Data Analysis Examples
Canonical Correlation Analysis
UCLA: Academic Technology Services, Statistical Consulting Group.
http://www.ats.ucla.edu/stat/R/dae/canonical.htm
(accessed Jun 2, 2010)
```

La ayuda de *R* bajo el input cc presenta el siguiente ejemplo:

```
#==============================
#CORRELACION CANONICA
#Demo de R
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
data(nutrimouse)
X=as.matrix(nutrimouse$gene[,1:10])
Y=as.matrix(nutrimouse$lipid)
res.cc=cc(X,Y)
plot(res.cc$cor,type="b")
plt.cc(res.cc)
```

Capítulo 12

Análisis no paramétrico

`Válido para todas las distribuciones`

97 ***Objetivo.*** *Conoceremos algunos test que son útiles para analizar datos que provengan no importa de qué tipo de distribución.*

12.1. Prueba t y anovas

La estadística no paramétrica consiste en un cuerpo de métodos que, en general, son universales y sirven para datos con cualquier distribución. Por su generalidad son muy ineficientes necesitando muchos datos para rechazar una hipótesis nula. Sin embargo, sus conclusiones son mucho más robustas que las de la estadística de la normal en el sentido que una perturbación marginal de los datos no necesariamente desestabiliza las decisiones previas. La estadística no paramétrica no trabaja con la media sino con la mediana, con cuartiles y rangos.

98 ***Ejemplo*** *Normalidad frustrada*

Cuando uno había pensado hacer una preuba t para comparar las medias de datos apareados pero no pudo porque los datos no se ajustaban a una distribución normal, uno puede entonces usar el test del rango con signo de Wilcoxon, que puede ser para datos apareados o independientes. La hipótesis nula en ambos casos es que la diferencia de medias vale cero. También tenemos tests similares a las anovas y anovas con bloqueo.

```
#==============================
#ESTADISTICA NO PARAMETRICA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
# Test del rango con signo de  Wilcoxon para datos apareados
#(Wilcoxon  Signed Rank)
x <- c(1.1, 1.3,  1.6,  1.7, 1.6, 1.3, 1.5, 1.5, 1.4)
y <- c(0.9, 1.1,  1.4,  1.5, 1.8, 1.2, 1.3, 1.6, 1.5)
#test de dos colas
wilcox.test(x, y, paired = TRUE)
#test de cola superior
wilcox.test(x, y, paired = TRUE, alternative = "greater")
#
#Test U de Mann-Whitney para 2 grupos independientes
# (independent 2-group Mann-Whitney U Test).
#x,y son numéricos.
x <- c(1.1, 1.3,  1.6,  1.7, 1.6, 1.3, 1.5, 1.5, 1.4)
```

```
y <- c(0.9, 1.1,  1.4,  1.5, 1.8)
wilcox.test(x,y,paired=FALSE)
#
#Test anova  por   rangos de Kruskal Wallis
# (Kruskal Wallis Test One Way Anova by Ranks).
x <- c(1.1, 1.3,  1.6,  1.7, 1.6, 1.3, 1.5, 1.5, 1.4)
y <- c(0.9, 1.1,  1.4,  1.5, 1.8)
z<- c(0.4, 1.1, 0.9, 1.2, 0.7, 1.3, 0.5)
kruskal.test(list(x, y, z))
#
#Test de Friedman para diseño de bloques al azar
# (Friedman Test for Randomized Block Design)
#
x <- c(1.1, 1.3,  1.6,  1.7, 1.6, 1.3, 1.5, 1.5, 1.4)
y <- c(0.9, 1.1,  1.4,  1.5, 1.8, 1.7, 1.3, 1.9, 1.5)
z<-  c(0.4, 1.1,  0.9,  1.2, 0.7, 1.3, 0.5, 0.9, 0.8)
bloques <- cbind(x,y,z)
friedman.test(bloques)
#
#non parametric multiple comparisons
#Comparaciones múltiples no paramétricas
#library(npmc)
#npmc(bloques)
```

Un test anova no paramétrico sólo dice si todas las medias son iguales o si hay algún par de medias diferentes. Si uno decide que hay algún par de medias diferentes y a uno le gustaría saber cuál, uno puede correr el siguiente test que se llama test no paramétrico de comparaciones múltiples, el cual compara todos los tratamientos por pares.

Para poder correr el test, necesitamos los dos paquetes siguientes :

mvtnorm npmc

Para saber si están instalados, corra el siguiente programa, que da error por cada paquete no instalado.

```
library(mvtnorm)
library(npmc)
```

Para instlar los paquetes sobre *Windows*, use el menu de *R* o alguna *GUI*. Para installar los paquetes sobre *Linux*: baje los paquetes de *cran*, sálvelos a la carpeta *downloads*, y sobre ella abra una terminal con derechos *su* y clave y *copie+peque* el siguiente programa o una modificación adecuada al serial de los paquetes:

```
R CMD INSTALL mvtnorm_0.9-92.tar.gz
R CMD INSTALL npmc_1.0-7.tar.gz
```

Después puede correr el siguiente script sobre *R*:

```
#==============================
#TEST NO PARAMETRICO DE COMPARACIONES MULTIPLES
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1: literatura
data(brain)
#Para salir de la ayuda, teclee q
help(brain)
```

```
#Parte 2: test de comparaciones múltiples.
#Es semejante al test de Tukey para anovas.
#En este caso hay compración entrre 3 grupos:
#c=control, l=hemisferio izquierdo, r= hemisferio derecho.
summary(npmc(brain), type="BF")
```

Capítulo 13

Análisis discriminante

```
Los diferentes van separados
```

99 ***Objetivo.*** *Aprender métodos formales para clasificar objetos disímiles en categorías diferentes.*

13.1. Idea básica

Se usa para discriminar unos objetos en diferentes clases. La metodología para el análisis discriminante lineal es como sigue:

Imaginémonos que los datos se representan sobre un plano y se agrupan formando dos montañitas que semejan campanas de Gauss bidimensionales. El problema es encontrar una línea que pase por el orígen tal que al proyectar las dos montañitas, la sombras sobre dicha línea sean tan diferentes como sea posible. Este procedimiento da la primera linea discriminante. Pero quizá exista otro punto de vista, otra línea que mejore el poder discriminatorio que ya se tenía: así se origina el segundo componente discriminante.

Cuando la discriminación se hace por medio de líneas se llama lineal. Cuando se hace por parábolas y líneas se llama cuadrática. Hoy en día existen procedimientos discriminantes por medio de curvas muy complicadas.

13.2. Clasificación en especies

El análisis discriminante se ha usado para predecir particionamiento en especies de diferentes objetos biológicos a los cuales se les hacen diversas mediciones. El ejemplo de *R* es el de la flores de la especie iris, confeccionado por Anderson y Fisher.

```
Fisher, R. A. (1936) The use of multiple measurements in taxonomic
     problems.  _Annals of Eugenics_, *7*, Part II, 179-188.

     The data were collected by Anderson, Edgar (1935).  The irises of
     the Gaspe Peninsula, _Bulletin of the American Iris Society_,
     *59*, 2-5.
```

Nuestra versión de este ejemplo es la siguiente, la cual necesita el paquete klaR, que se instala en Linux con la adatación al serial del comando siguiente:

```
R CMD INSTALL   klaR_0.6-3.tar.gz
```

```
#==============================
#ANALISIS DISCRIMINANTE
#Correr por partes
#Limpia la memoria
rm(list = ls())
```

```
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1
#Teclee q para salir de la ayuda
help(iris)
#
#Parte 2
Iris <- data.frame(rbind(iris3[,,1], iris3[,,2], iris3[,,3]),
                         Sp = rep(c("s","c","v"), rep(50,3)))
train <- sample(1:150, 75)
table(Iris$Sp[train])
#lda: linear model for discriminant analysis
#análisis discriminante lineal
z <- lda(Sp ~ ., Iris, prior = c(1,1,1)/3, subset = train)
#Vista discriminadora
plot(z)
#
#Parte 3
prediccion <- predict(z, Iris[-train, ])$class
summary(prediccion)
z1 <- update(z, . ~ . - Petal.W.)
z1
# Histograma de las sombras sobre
#la primera línea discriminadora
plot(z, dimen=1, type="both")
#
#parte 4
#qda: quadratic discriminant analysis
#  análisis discriminante cuadrático
library(MASS)
zq <- qda(Sp ~ ., Iris, prior = c(1,1,1)/3, subset = train)
prediccionq <- predict(zq, Iris[-train, ])$class
summary(prediccionq)
zq <- update(zq, . ~ . - Petal.W.)
zq
3
# Exploratory Graph for LDA or QDA
library(klaR)
partimat(Sp ~ .,data=Iris,method="lda")
partimat(Sp ~ .,data=Iris,method="qda")
```

Capítulo 14

Análisis de conglomerados (clusters)

Arboles de similitud

100 ***Objetivo*** *Para reunir varios objetos en grupos de semejanza o similitud usamos el* ***análisis de clusters*** *o conglomerados. El output final es un árbol en el que uno puede leer qué tan semejante o diferentes son los diversos objetos. R provee varias metodologías para hacerlo.*

14.1. Metodología

Para poder hacer análisis de clusters, se requiere hallar la distancia entre cada par de objetos según los descriptoes que se tenga. Con dicha matriz, puede usarse la función **hclust** para hallar los clusters. Dicha función comienza tomando a cada objeto como su propio cluster y va añadiendo a cada cluster aquel cluster que más se le parezca, es decir, que esté más cercano. Se termina cuando todo el mundo esté en un único cluster. Se necesita entonces una matriz que de las distancias entre los objetos, la cual se puede calcular con la función **dist**, de la cual hay varias versiones. por ejemplo, la distancia euclidiana calcula la distancia por la fórmula de Pitágoras. El output es un árbol o dendograma done las distancias se representan verticalmente y los conglomerados se pueden marcar explícitamente.

101 ***Ejemplos varios***

Si los datos son cuantitativos, uno primero los estandariza:

```
#============================
#ANALISIS DE CLUSTERS DATOS CUANTITATIVOS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Hay 4 individuos
x <- c(1.1, 2, 3, 4, 2, 3, 2)
y <- c(0.8, 0.5, 0.7, 0.6, 0.4,  0.5, 0.6)
z<- c(2, 3, 2, 3, 3, 2.3, 3)
w <- c(0, 0, 0, 0, 0, 0)
datos <- rbind(x,y,z,w)
#Datos cuantitativos
# Revisar omisiones
datos <- na.omit(datos)
datos
#estandarizar
datos <- scale(datos)
datos
#
```

```
#euclidean: distancia por Pitágoras
hc <- hclust(dist(datos, method = "euclidean"), method = "complete")
plot(hc)
plot(hc, hang = -1)
```

Cuando uno tiene datos de presencia-ausencia se procede como sigue:

```
#==============================
#ANALISIS DE CLUSTERS DATOS = PRESENCIA-AUSENCIA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Hay 5 individuos.
#Individuo x
x <- c(0, 0, 1, 1, 1, 1)
y <- c(1, 0, 1, 1, 0, 1)
z<- c(1, 1, 1, 1, 1, 1)
w <- c(0, 0, 0, 0, 0, 0)
t<- c(0, 1, 0 ,1 , 0 ,1 ,0)
datos <- rbind(x,y,z,w,t)
#cálculo del dendograma.
#euclidean: distancia por Pitágoras
hc <- hclust(dist(datos, method = "euclidean"), method = "complete")
plot(hc)
#hang = parámetro de dibujo
plot(hc, hang = -1)
#maximum: se toma el valor absoluto entre cada componente
#y se escoge el máximo valor.
hc <- hclust(dist(datos, method = "maximum"),method = "complete")
plot(hc)
plot(hc, hang = -1)
#manhattan: se toman los valores absolutos entre cada par de componentes
#y se suman.
#method = "complete" implica que se tiene todo en cuenta.
hc <- hclust(dist(datos, method = "manhattan"),method = "complete")
plot(hc)
plot(hc, hang = -1)
#binary: una medida de la proporción de los que tienen uno.
hc <- hclust(dist(datos, method = "binary"),method = "complete")
plot(hc)
plot(hc, hang = -1)
#canberra:  sum(|x_i - y_i| / |x_i + y_i|)
hc <- hclust(dist(datos, method = "canberra"),method = "complete")
plot(hc)
plot(hc, hang = -1)
#method = "cen":
#se usan los centrodies de cada cluster para hallar el siguiente
#nivel de aglomeramiento
hc <- hclust(dist(datos, method = "canberra"), method = "cen")
plot(hc)
plot(hc, hang = -1)
#
# K-Means Cluster Analysis:
#(K clusters por encargo)
#se especifica el número deseado de clusters. Aquí son 2
```

```
k <- kmeans(datos, 2)
# get cluster means
aggregate(datos,by=list(k$cluster),FUN=mean)
# append cluster assignment
nuevosDatos <- data.frame(datos, k$cluster)
#hc <- hclust(dist(nuevosDatos, method = "canberra"), method = "cen")
#
# Señalamiento de clusters
d <- dist(datos, method = "euclidean")
dendograma <- hclust(d, method="ward")
plot(dendograma)
#Tome 2 clusters
groups <- cutree(dendograma, k=2)
# Encierre los clusters en rojo
rect.hclust(dendograma, k=5, border="red")
#
# Clustering por modelamiento
library(mclust)
dendo <- Mclust(datos)
plot(dendo, datos)
print(dendo)
```

Para el análisis estadístico necesitamos el paquete pvclust. Para saber si está o no está en la librería se corre el siguiente comando:

```
library(pvclust)
```

Si no está, se baja y se instala. Sobre Windows, se baja y se usa el menú de *R* o la *GUI*. Sobre Linux, se baja a *downloads* y allí se abre una terminal con derechos *su* y se copia+pega una adpatación al serial del siguiente comando:

```
R CMD INSTALL  pvclust_1.2-1.tar.gz
```

Una vez instalado, se puede correr el siguiente programa

```
#ANALISIS ESTADISTICO DE CLUSTERS
#Corre despacio
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(1,1), pch=16)
#x contiene la información sobre la
#presencia o ausencia de la característica x
x <- c(0, 0, 1, 1, 1, 1)
#y = está o no está la característica y
y <- c(1, 0, 1, 1, 0, 1)
z<- c(1, 1, 1, 1, 1, 1)
w <- c(0, 0, 0, 0, 0, 0)
t<- c(0, 1, 0 ,1 , 0 ,1 ,0)
datos <- rbind(x,y,z,w,t)
# Análisis estadístico
#pvclust = p-value de agrupamiento
library(pvclust)
dendo <- pvclust(datos, method.hclust="ward",
   method.dist="euclidean")
plot(dendo) # dendogram with p values
# Rodee los clusters bien definidos con
```

```
#rectángulos en rojo.
pvrect(dendo, alpha=.95)
```

Capítulo 15

Bioinformática

`Análisis de secuencias de DNA`

102 ***Objetivo.*** *Conocer el enorme potencial de R para el estudio de las secuencias de ADN o de proteínas.*

103 ***Usando R-base***

El paquete básico de *R* tiene funciones que pueden servir para el análisis de datos sobre el genoma, que son genéricas y sirven, por ejemplo, para contar las letras de una cadena, o para decidir si dos cadenas son iguales o no. Sin embargo, la bioinformática es algo tan tremendamente importante que hay una comunidad especialmente dedicada a perfeccionar poderosas herramientas de software libre para su análisis. Nuestra opción es estudiar aquella contribución que ha tomando *R* como su base de operaciones.

15.1. Bioconductor

El producto básico para bioinformática es el proyecto **Biocunductor** (2010) pero, como veremos, hay muchos otros proyectos. Para obtener información sobre *bioconductor* podemos ir

`http://www.bioconductor.org/`

Un manual para estudio es el siguiente:

`http://manuals.bioinformatics.ucr.edu/home/R_BioCondManual`

En este sitio también hay un manual de Linux.

104 ***Instalación***

Bioconductor no es un paquete sino un enorme ramillete de paquetes que reflejan la gran diversidad de tareas de la bioinformática. Para ver una lista de todos los paqutes, en la página de Bioconductor, elegir *downloads + software*. Hay, por supuesto, una conjunto básico de paquetes que se instala con copie+pegue a la consola de *R* del siguiente programa:

```
source("http://bioconductor.org/biocLite.R")
biocLite()
```

Este script trata de instalar los siguientes paquetes: affy, affydata, affyPLM, annaffy, annotate, Biobase, Biostrings, DynDoc, gcrma, genefilter, geneplotter, hgu95av2.db, limma, marray, matchprobes, multtest, ROC, vsn, xtable, affyQCReport.

Esta operación es demorada y puede tener éxito parcial. Mientras dure, hay actividad en la consola, al terminar, aparece el signo > y ya no hay más.

Para chequear si la descarga funcionó, al menos parcialmente, pruebe el siguiente programa que nos enseña como citar los recursos de Bioconductor. El comando *openVignette()* nos muestra la documentación del paquete recién abierto, en este caso Biobase y affy. Uno escoge la opción deseada, que para empezar puede ser la uno. Para salir del menú de opciones, se teclea 0.

```
#============================
#BIOCONDUCTOR
#Parte1
library("Biobase")
citation("Biobase")
options(pdfviewer = 'okular')
openVignette()
#Parte 2
citation("affy")
library(affy)
options(pdfviewer = 'okular')
openVignette()
```

Si tiene problemas porque la pantalla no aguanta mucha información: dirija el output a un archivo previamente indicado desde el menu de *R*.

15.2. Calentando motores

La idea fundamental de la bioinformática es que en el núcleo de la vida hay un tremendo flujo de información que abarca desde el nivel molecular hasta el macrocosmos. Ya no estamos en capacidad de estudiar dicho flujo a la topa tolondra sino que todo estudio al respecto se organiza dentro un marco teórico muy buen definido: la teoría evolutiva. En el presente momento histórico la bioinformática existe un sesgo hacia el nivel molecular que aún no terminamos de digerir, pero es de suponer que evolucione hacia un equilibrio que represente las muchas facetas de la empresa.

105 ***Marco teórico esencial***

Nuestra versión del marco teórico que sostiene y da forma a la bioinformática es como sigue:

EL DNA tiene la información genética que está en los cromosomas. Una parte de dicha información, llamada estructural, funciona como una biblioteca, de la cual se consultan los libros apropiados en el momento adecuado. Los libros no se prestan sino que se prestan fotocopias en forma de RNA y se traduce en proteínas. Otra parte de la información genética, llamada regulatoria, tiene como objetivo regular los procesos celulares relacionados con el DNA. Una tercera parte de la información de los cromosomas es de sostén y es responsable porque el DNA pueda servir como molécula para todas las funciones que se le adscriben.

La parte estructural que codifica para proteínas y enzimas se convierte en vida mediante el siguiente proceso: el ribosoma toma una cadena de RNA, lo lee y lo va traduciendo a proteínas, las cuales se repliegan sobre sí mismas apenas van saliendo formando estructuras tridimensionales terciarias. Estas estructuras adquieren una función de acuerdo a la especificidad de su estructura terciaria, por ejemplo enzimática, para acelerar reacciones químicas, o estructural, para formar, por ejemplo, la estructura en un hueso.

La reproducción consiste en fabricar el DNA de los hijos a partir del de los padres. La reproducción puede incluir cambios o mutaciones que crean diferencias: hay evolución. Esta evolución es una hecho del cual la teoría evolutiva lo hace responsable de nuestra existencia, es decir, de la formación de nuevas especies a partir de un conglomerado de organismos unicelulares, los cuales surgieron por autoorganización de la materia inerte.

Para ver la enorme diversidad de las tareas generadas al tratar de entender y quizá cuestionar este marco teórico, podemos ver la lista de los paquetes de Bioconductor o de Cran. También podemos echarle una ojeada a los siguiente manuales, que son nuestro consultorio básico y que nos llevan a links muy importantes:

```
Thomas Girke (2010)R & Bioconductor Manual
http://manuals.bioinformatics.ucr.edu/home/R_BioCondManual
```

```
Tyler Backman, Rebecca Sun and Thomas Girke (2010)
 HT Sequence Analysis with R and Bioconductor
 http://manuals.bioinformatics.ucr.edu/home/ht-seq

Link para explorar: Minicurso sobre  Microarrays en R:
http://biocluster.ucr.edu/%7Etgirke/HTML_Presentations/
Manuals/Microarray/arrayBasics.pdf
```

En este capítulo implementaremos algunas facetas de la bioinformática. Nos parece que si alguien desea volverse un experto en el tema, ha escogido una carrera fascinante. En dicho cao, lo mejor es tener en cuenta que hay muchas aplicaciones gratis y que vienen en Linux y que hasta el día de hoy no han sido adaptada a MS Windows. Por eso, es conveniente invertir un tiempo para adquirir acceso a Linux. También hay otras aplicaciones que vienen sólo para Windows.

106 *Biostrings*

La librería básica es **Biostrings**. Para conocer todos los métodos que contiene, se corre el siguiente comando:

```
library(Biostrings)
library(help=Biostrings)
```

107 *DNA a proteína*

Para pasar de DNA a proteína, necesitamos el código genético. Cuando se dice código genético, se asume que se trata del código genético estándard, el que aparece en la gran mayoría de seres vivos, pero se han descubierto más de 10. El código genético puede ser leído de Internet:

```
#==============================
library(Biostrings)
AAdf <- read.table(file="http:
//faculty.ucr.edu/~tgirke/Documents/R_BioCond/
My_R_Scripts/AA.txt", header=T,"\t")
AAdf
```

o puede usar nuestra adaptación, que está incluída en el siguiente programa:

```
#==============================
#DNA A PROTEINA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1
#El código genético:
#64 codones
Codon <- c("TCA", "TCG", "TCC", "TCT", "TTT", "TTC", "TTA", "TTG",
           "TAT", "TAC", "TAA", "TAG", "TGT", "TGC", "TGA", "TGG",
           "CTA", "CTG", "CTC", "CTT", "CCA", "CCG", "CCC", "CCT",
           "CAT", "CAC", "CAA", "CAG", "CGA", "CGG", "CGC", "CGT",
           "ATT", "ATC", "ATA", "ATG", "ACA", "ACG", "ACC", "ACT",
           "AAT", "AAC", "AAA", "AAG", "AGT", "AGC", "AGA", "AGG",
           "GTA", "GTG", "GTC", "GTT", "GCA", "GCG", "GCC", "GCT",
           "GAT", "GAC", "GAA", "GAG", "GGA", "GGG", "GGC", "GGT")
#Notación de una letra de los aminoácidos
AA_1 <- c("S", "S", "S", "S", "F", "F", "L", "L",
          "Y", "Y", "*", "*", "C", "C", "*", "W",
          "L", "L", "L", "L", "P", "P", "P", "P",
```

```
          "H", "H", "Q", "Q", "R", "R", "R", "R",
          "I", "I", "I", "M", "T", "T", "T", "T",
          "N", "N", "K", "K", "S", "S", "R", "R",
          "V", "V", "V", "V", "A", "A", "A", "A",
          "D", "D", "E", "E", "G", "G", "G", "G")
#Notación de 3 letras de los aminoácidos
AA_3 <- c("Ser", "Ser", "Ser", "Ser", "Phe", "Phe", "Leu", "Leu",
          "Tyr", "Tyr", "Stop", "Stop", "Cys", "Cys", "Stop", "Trp",
          "Leu", "Leu", "Leu", "Leu", "Pro", "Pro", "Pro", "Pro",
          "His", "His", "Gln", "Gln", "Arg", "Arg", "Arg", "Arg",
          "Ile", "Ile", "Ile", "Met", "Thr", "Thr", "Thr", "Thr",
          "Asn", "Asn", "Lys", "Lys", "Ser", "Ser", "Arg", "Arg",
          "Val", "Val", "Val", "Val", "Ala", "Ala", "Ala", "Ala",
          "Asp", "Asp", "Glu", "Glu", "Gly", "Gly", "Gly", "Gly")
#Nombre completo de los 20 aminoácidos oficiales
AA_Full <- c("Serine", "Serine", "Serine", "Serine",
             "Phenylalanine", "Phenylalanine", "Leucine","Leucine",
             "Tyrosine", "Tyrosine", "Stop", "Stop",
             "Cysteine", "Cysteine", "Stop", "Tryptophan",
             "Leucine", "Leucine", "Leucine", "Leucine",
             "Proline", "Proline", "Proline", "Proline",
             "Histidine", "Histidine", "Glutamine", "Glutamine",
             "Arginine", "Arginine", "Arginine", "Arginine",
             "Isoleucine", "Isoleucine", "Isoleucine", "Methionine",
             "Threonine", "Threonine", "Threonine", "Threonine",
             "Asparagine", "Asparagine", "Lysine", "Lysine",
             "Serine", "Serine", "Arginine", "Arginine",
             "Valine", "Valine", "Valine", "Valine",
             "Alanine", "Alanine", "Alanine", "Alanine",
             "Aspartic acid", "Aspartic acid", "Glutamic acid", "Glutamic acid",
             "Glycine", "Glycine", "Glycine", "Glycine")
#Cada codón tiene su anticodón:
AntiCodon <- c("TGA", "CGA", "GGA", "AGA", "AAA", "GAA", "TAA", "CAA",
               "ATA", "GTA", "TTA", "CTA", "ACA", "GCA", "TCA", "CCA",
               "TAG", "CAG", "GAG", "AAG", "TGG", "CGG", "GGG", "AGG",
               "ATG", "GTG", "TTG", "CTG", "TCG", "CCG", "GCG", "ACG",
               "AAT", "GAT", "TAT", "CAT", "TGT", "CGT", "GGT", "AGT",
               "ATT", "GTT", "TTT", "CTT", "ACT", "GCT", "TCT", "CCT",
               "TAC", "CAC", "GAC", "AAC", "TGC", "CGC", "GGC", "AGC",
               "ATC", "GTC", "TTC", "CTC", "TCC", "CCC", "GCC", "ACC")
Codigo <- data.frame(Codon, AA_1,AA_3,AA_Full, AntiCodon)
Codigo
#
#Translación en notación de una letra
AA1 <- Codigo[,2];
# Asocia los aminoácidos con codones
names(AA1) <- Codigo[,1]
#Genera 5 secuencias DNA al azar cada una con 20 bases
sapply(1:5, function(x) paste(sample(c("A","T","G","C"), 20,
                                     replace=T), collapse=""))
#Generamos una secuencia de 90 bases:
gen<- sapply(1:1, function(x) paste(sample(c("A","T","G","C"), 90,
                                           replace=T), collapse=""))
#Lista completa de gen
gen
```

```
#
#DNA a RNA
library(Biostrings)
d <- DNAString(gen)
r <- RNAString(d)
#
#Lista completa de gen como DNA STRING
d
#Lista parcial
d[2:6]
#Leer al revés:
dReves <- d[length(d):1]
dReves
#
#DNA a proteína
# Insertamos "_" después de cada tripleta
gen <- gsub("(...)", "\\1_", gen)
# Convierte gen en un vector de tripletas.
gen <- unlist(strsplit(gen, "_"))
# Quita las tripletas incompletas
gen <- gen[grep("^...$", gen)]
#Translación:
AA1[gen]
#
#Translación en notación de 3 letras
AA3 <- Codigo[,3];
# Creates named vector with genetic code
names(AA3) <- AAdf[,1]
#Translación:
AA3[gen]
#
#Translación en notación completa
AAC <- Codigo[,4]
AAC
```

Atención: si uno tiene una secuencia pero no sabe en donde empieza el gen en estudio, entonces uno tiene que hacer varios tanteos: empezando con la primera letra, con la segunda, con la tercera y otras 3 al revés. Se dice que hay 6 frames o modos de lectura.

108 *Leyendo bases de datos*

Secuenciar es costoso: cuando lo haga, publique sus resultados y póngalos en una base de datos, por ejemplo la NCBI (US National Center forBiotechnology Information)

```
http://www.ncbi.nlm.nih.gov/
```

la cual ofrece en su página links a un amplio espectro de tutoriales. Los usuarios podrán ir allí a buscar secuencias y diverso tipo de información.

R puede tratar de bajar automáticamente una secuencia como sigue:

```
#============================
#BAJAR SECUENCIA: NO CORRA ESTE PROGRAMA
#A MENOS QUE SEA EN SERIO
library(Biostrings)
#La siguiente instrucción se corre una única vez
#y el resultado se guarda en memoria.
#Para ello se redirige el output de R a un archivo específico.
```

```
sec1 <- readFASTA("ftp://ftp.ncbi.nih.gov/genbank/genomes/
Bacteria/Halobacterium_sp/AE004437.ffn", strip.descs=T)
#Publique las primeras 10 bases
writeLines(strtrim(sec1, 10))
#Ponga la secuencia en un archivo  y grábelo
writeLines(as.vector(t(cbind("AE004437.ffn", sec1))), "sec1.fasta")
```

Fasta es uno de los formatos más populares para secuencias tanto de ácidos nucleicos como de proteínas. Su nombre significa *Fast Alignment* pues fue introducido por FASTP un programa para el estudio de comparación de secuencias, muy popular y ahora con varias a actualizaciones en las bases de datos. El algoritmo usado por la NCBI para el estudio de secuencias es el Blast(Basic Local Alignment Search Tool), el cual puede bajarse.

En general, hay demasiadas herramientas de software libre para bioinformatica. Incluso, tenemos la opción de usar calculadoras en la web. Un ejemplo muy digno es *Genepop* que hace tareas relacionadas con genética de poblaciones:

```
http://genepop.curtin.edu.au/
```

El link a la calculadora es el mismo título de la página.

En medio de tanta diversidad, el sueño de la comunidad *R* es tener todo junto y sobre el mismo mesón de trabajo. Sin embargo, este sueño es bien difícil de llevar a cabo y es mejor irse haciendo a la idea de manejar paquetes en Fortran, Java, C o Perl o en algún lenguaje apropiado para inteligencia artificial. De hecho, el análisis de secuencias no es algo simple y exige una programación de alto nivel con cosas que se asocian con la inteligencia humana, como aprendizaje para el reconocimiento de patrones linguísticos y que diferencie, por ejemplo, una secuencia que codifica una proteína de otra que sirve de objetivo a alguna enzima de restricción. Para ver una lista de software relacionado específicamente con secuencias uno puede darle a un buscador la frase: *sequence alignment software* o también uno puede visitar el sitio

```
http://www.sequence-alignment.com/index.html
```

Otra base de datos muy famosa es la del (SIB) Swiss Institute of Bioinformatics que se consigue en la siguiente dirección:

```
http://us.expasy.org/
```

Es inteligente tener en cuenta que la NCBI queda en los E.U. y que el SIB queda en Europa central. En horas hábiles dichas agencias están saturadas. Hay dos salidas: visitar las agencias en horas de descanso según el horario local, o, para el caso del SIB, abrir una cuenta, lo cual le permitirá a uno acceder a un servicio mucho menos dependiente del flujo de visitas.

Para ver una lista de las bases de datos para biología, uno puede ir a

```
BioTech FYI Center (2010) Biological databases
http://biotech.fyicenter.com/resource/Biological_databases.html
```

Supongamos pues que hemos bajado una secuencia y que *R* la tiene guardada bajo el nombre *sec1*. Lo que *sec1* guarda tiene varios componentes:

```
#============================
#Secuencias solas:
secs <- lapply(sec1, function(x) x$seq)
#Identificadores
identificadores <- sapply(sec1, function(x) x$desc)
```

Lo primero que uno hace para saber la función de una secuencia es compararla con las secuencias de las bases de datos para ver que secuencias parecidas hay y qué funciones tienen. Las agencias tienen facilidades apropiadas a las cuales uno puede acceder. En esencia, lo que las agencias hacen es correr un programa semejante al siguiente:

Si uno tiene una secuencia y una base de datos, uno puede averiguar si el oligonucleótido está presente o no en la gran cadena:

```
#============================
#BUSCAR MOTIVO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(Biostrings)
#Base de datos con todas las secuencias (3 en este ejemplo)
baseDatos <- c("TGCTATGCTACGTA",
               "TTCGGCAATGCAGCTA",
               "TTTTCAGCGCGTACCCAT")
#Reporta todas las secuencias que tienen    "TGC" en algún lado
baseDatos[grep("TGC", baseDatos)]
#Reporta la posición de "TGC" en cada una de las secuencias
#si no está, reporta -1
pos <- regexpr("TGC", baseDatos)
as.numeric(pos);
#Por cada secuencia reporta el número de caracteres que coinciden,
#si no hay, reporta -1
attributes(pos)$match.length
```

110 ***Estructuras de las proteínas***

El DNA tiene información unidimensional con la cual se construyen las cadenas polipeptídicas o proteicas. Estas cadenas recién salidas del ribosoma se llaman estructuras primarias. Por otro lado, la función de las enzimas depende de su estructura en el espacio o estructura terciaria. Eso crea un interrogante: ¿cómo es posible que una estructura primaria codifique para una terciaria? La respuesta es que la estructura terciaria depende tanto del medio donde se encuentra la enzima como de su estructura primaria, es decir, la secuencia de aminoácidos que se lee directamente de DNA. La información del DNA asume que la enzima en cuestión está dirigida, por ejemplo, al citoplasma, para lo cual hay instrucciones precisas, y en dicho medio la enzima se repliega sobre sí misma. En el proceso de repliegue lo primero que se estabiliza son la estructuras cortas determinadas por las interacciones entre aminoácidos vecinos: se forma la estructura secundaria. Más luego la estructura secundaria sigue intraactuando hasta formar, por ejemplo, un corpúsculo globular, la estructura terciaria. Dicha estructura puede ser determinada por ella misma en su totalidad, o bien puede ser inducida por otras enzimas llamadas chaperonas. De todas formas, su estado final es estable aunque dinámico y depende únicamente de su estrcutrua primaria.

En suma: la estructura terciaria es un reflejo de la estructura primaria. No se sabe exácta y mecanicísticamente cómo es que eso sucede y averiguarlo es hoy en día un problema fundamental. Sin embargo, hay muchos programas que proponen estructuras terciarias para cualquier polipéptido y aunque no son muy precisos, nos dan una idea de la estructura espacial de la enzima o proteína en cuestión. Las agencias también tienen este servicio y uno puede usarlo con responsabilidad: estas agencias son muy pocas y atienden a toda la tierra, para no saturarlas es conveniente instalar en el entorno personal programas que puedan hacer lo mismo pero en casa: en vista de éso, la comunidad de *R* está en el proyecto de armar un programa con todas las facilidades necesarias. Su nombre es: **Bio3D** para *R*. Este es un paquete que promete correr en cualquier plataforma. Dicho paquete se baja de la página *Bio3D* en el siguiente link:

`http://mccammon.ucsd.edu/~bgrant/bio3d/index.html`

Los autores suguieren la siguiente forma de citar el uso de *Bio3D*:

```
Bio3D: An R package for the comparative analysis of protein structures.
Grant, Rodrigues, ElSawy, McCammon, Caves, (2006) Bioinformatics 22, 2695-2696
```

Hay un manual al final de la sección sobre documentación.

Antes de bajar el paquete, es necesario asegurarse que se tiene una versión reciente de un compilador de Perl. Para Linux no debe haber problema. Sobre OpenSUSE la investigación e instalación se hace con YAST2.

Para bajar el paquete Bio3D, uno va a downloads y allí elige la versión apropiada: hay una para Windows y otra que corre en cualquier plataforma, ésa es la que sirve para Linux.

Para Windows, los autores sugieren el siguiente protocolo basado en la GUI. Primero, bajar la fuente binaria, en formato zip. Segundo, desde R, encender el Rcmdr. Tercero, hacer click en packages y seleccionar la funete binaria recién bajada. Para terminar, oprimir Open.

Para Linux: cuando uno haga click sobre el link para bajar el paquete, se despliega un diálogo que pregunta si uno desea guardar el paquete. Uno lo guarda, lo cual se hace automáticamente sobre la carpeta Downloads. Encima de dicha carpeta uno abre una terminal con derechos de superuser, para lo cual uno teclea *su* y más luego, cuando se lo pidan, la clave. Después uno teclea:

```
R CMD INSTALL bio3d_*.tar.gz
```

A continuación uno llama a *R* tecleando *R* sobre la terminal. Y allí corre el programa siguiente:

```
#=============================
#ESTRUCTURA PROTEINA KINESINA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(bio3d)
data(kinesin)
attach(kinesin)
#matrix de alineación
kinesin$aln
#Alineación 3d
kinesin$pdbs
#Lista de núcleos
kinesin$core
# Matriz de coordenadas de núcleos
kinesin$xyz
#Histograma de volumenes de núcleos
col=rep("black", length(core$volume))
col[core$volume<2]="pink"; col[core$volume<1]="red"
plot(core, col=col)
#Objetos diversos
kinesin$pc.xray
#Estructura secundaria
kinesin$sse
```

En la sección de documentación uno puede encontrar un link a un tutorial con más de 20 ejemplos. Y asociados a la lista de funciones uno puede encontrar más de 100 ejemplos más.

111 *Mutación*

La replicación del material genético no es perfecta y suceden mutaciones. Un tipo de ella se simula con el método substitución genética, *gsub()* que cambia una secuencia específica por otra también específica:

```
#=============================
#MUTACION
```

```
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(Biostrings)
#Genoma con todos los genes
genoma <- c("TGCTATGCTACGTA", "TTCGGCAATGCAGCTA", "TTTTCAGCGCGTACCCAT")
#Substitución de "TGC" por "tgc"  en todas los genes que empiezan por "TGC"
gsub("^TGC", "tgc", genoma)
#Substitución de "TGC" por "tgc", en todos los genes  en toda parte
gsub("TGC", "tgc", genoma)
```

También hay inserciones, deleciones, reversiones, traslocaciones. Consideramos que simulaciones detalladas de todos estos fenómenos es mejor hacerlas en Java. Por supuesto, existe la forma de hacer que *R* y Java trabajen en conjunto.

112 ***Polimorfismo***

Los estudios de secuenciación han demostrado que el genoma tolera muy bien bastantes mutaciones. la primera implicación que tenemos es que ya no podemos decir que una enzima dada corresponda a una secuencia sino que corresponde a un conjunto de secuencias,cuyo conocimiento siempre está en expansión. En el núcleo de un estudio de este tipo se encuentra la tarea de comparar secuencias. Un método muy recomendado es el de la matrices de puntos (Dot Matrices), las cuales ponen un punto cada vez que hay una coincidencia. A estas matrices también se les llama matrices de presencia-ausencia. Para secuencias largas, las coincidencias se pueden escoger de tamaño apropiado.

Al estudio de comparación de secuencias se le llama oficalmente de **alineamiento**. El problema es que dos secuencias pueden ser muy parecidas pero diferir por una inserción, o por una reversión, etc. Por ello, se han confeccionado algoritmos poderosos para estudiar la forma que de la mejor alineación:

```
#==============================
#ALINEAMIENTO DE SECUENCIAS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(Biostrings)
#Secuencias dadas.
s1 <- DNAString("GGGCCCCC")
s2 <- DNAString("GGGTTCCC")
#Computa un alineamiento test local a pares con el algoritmo de  Smith-Waterman.
myalign <- pairwiseAlignment(s1, s1, type="local")
subject(myalign)
pattern(myalign)
attributes(myalign)[-3]
#Computa un alineamiento local a pares con el algoritmo de  Smith-Waterman.
myalign <- pairwiseAlignment(s1, s2, type="local")
subject(myalign)
pattern(myalign)
attributes(myalign)[-3]
# Computa un alineamiento global con el algoritmo de  Needleman-Wunsch
myalign <- pairwiseAlignment(s1, s2, type="global")#Publicamos resultados
#Publicamos resultados
subject(myalign)
pattern(myalign)
attributes(myalign)[-5]
```

Hay unos ejemplos en las librerías. El siguiente es muy instructivo:

```
#==============================
#ALINEACION DE SECUENCIAS
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Ejemplos de R
library(Biostrings)
## Nucleotide global, local, and overlap alignments
s1 <-
     DNAString("ACTTCACCAGCTCCCTGGCGGTAAGTTGATCAAAGGAAACGCAAAGTTTTCAAG")
s2 <-
     DNAString("GTTTCACTACTTCCTTTCGGGTAAGTAAATATATAAATATATAAAAATATAATTTTCATC")

# First use a fixed substitution matrix
mat <- nucleotideSubstitutionMatrix(match = 1, mismatch = -3, baseOnly = TRUE)
globalAlign <-
         pairwiseAlignment(s1, s2, substitutionMatrix = mat,
                           gapOpening = -5, gapExtension = -2)
localAlign <-
         pairwiseAlignment(s1, s2, type = "local", substitutionMatrix = mat,
                           gapOpening = -5, gapExtension = -2)
overlapAlign <-
         pairwiseAlignment(s1, s2, type = "overlap", substitutionMatrix = mat,
                           gapOpening = -5, gapExtension = -2)

# Then use quality-based method for generating a substitution matrix
pairwiseAlignment(s1, s2,
    patternQuality = SolexaQuality(rep(c(22L, 12L), times = c(36, 18))),
    subjectQuality = SolexaQuality(rep(c(22L, 12L), times = c(40, 20))),
    scoreOnly = TRUE)

# Now assume can't distinguish between C/T and G/A
pairwiseAlignment(s1, s2,
    patternQuality = SolexaQuality(rep(c(22L, 12L), times = c(36, 18))),
    subjectQuality = SolexaQuality(rep(c(22L, 12L), times = c(40, 20))),
    type = "local")
mapping <- diag(4)
dimnames(mapping) <- list(DNA_BASES, DNA_BASES)
mapping["C", "T"] <- mapping["T", "C"] <- 1
mapping["G", "A"] <- mapping["A", "G"] <- 1
pairwiseAlignment(s1, s2,
   patternQuality = SolexaQuality(rep(c(22L, 12L),
                                  times = c(36, 18))),
   subjectQuality = SolexaQuality(rep(c(22L, 12L),
                                  times = c(40, 20))),
   fuzzyMatrix = mapping, type = "local")
## Amino acid global alignment
pairwiseAlignment(AAString("PAWHEAE"), AAString("HEAGAWGHEE"),
                          substitutionMatrix = "BLOSUM50",
                          gapOpening = 0, gapExtension = -8)
```

La teoría sobre alineamiento puede leerse en las siguientes filminas:

```
Murphy R (2010) Introduction to Computational Biology, 03-310/42-334,
 Spring Term, 2006
```

```
http://www.cmu.edu/bio/education/courses/03310/
```

113 ***Un árbol genético***

Una manera muy cómoda de resumir un estudio de comparación de secuencias es por medio de un árbol genético. El paquete apropiado es

ape: Analyses of Phylogenetics and Evolution

El objetivo de este paquete es impresionante:

ape provides functions for reading, writing, plotting, and manipulating phylogenetic trees, analyses of comparative data in a phylogenetic framework, analyses of diversification and macroevolution, computing distances from allelic and nucleotide data, reading nucleotide sequences, and several tools such as Mantel's test, computation of minimum spanning tree, generalized skyline plots, estimation of absolute evolutionary rates and clock-like trees using mean path lengths, non-parametric rate smoothing and penalized likelihood. Phylogeny estimation can be done with the NJ, BIONJ, and ME methods.

Para poder correr ape se necesita tener varios paquetes instalados. Para hacer un chequeo, corra el siguiente programa:

```
library(gee)
library(nlme)
library(lattice)
```

Si todos esos paquetes están, *R* no alegará nada. Pero si alguno le falta, dirá Error. Entonces, hay que montar los paquetes que falten. Por ejemplo, puede faltar *gee*. Se baja de *cran*:

```
http://cran.at.r-project.org/web/packages/ape/index.html
```

y se instala en Windows usando el menu de *R* y de ser necesario bajando el archivo para después instalarlo con ayuda del menu. Sobre Linux se instala con la orden de instalarlo, que puede lucir algo así:

```
R CMD INSTALL gee_4.13-14.tar.gz
```

Una vez que todos los paquetes están instalados, se puede instalar *ape*. Para bajar *ape*, vaya a alguna de las siguientes direcciones:

```
http://cran.at.r-project.org/web/packages/ape/index.html
http://ape.mpl.ird.fr/
```

Una vez que se tenga el archivo en alguna carpeta, se abre sobre ella una terminal, se adquieren derechos *su* y después se ordena:

```
R CMD INSTALL ape_2.5-3.tar.gz
```

La documentación se puede bajar de

```
http://ape.mpl.ird.fr/
```

en donde uno va a *features* y después busca un link a *ape Manual*. Cuando ya se haya instalado, se puede correr sobre *R* el siguiente programa, el cual viene en el manual adjunto a la función *ace* (ancestral character estimation)

```
#==============================
#ARBOL GENETICO
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(1,1), pch=16)
library(ape)
### Just some random data...
data(bird.orders)
x <- rnorm(23)
### Compare the three methods for continuous characters:
ace(x, bird.orders)
ace(x, bird.orders, method = "pic")
ace(x, bird.orders, method = "GLS",
corStruct = corBrownian(1, bird.orders))
### For discrete characters:
x <- factor(c(rep(0, 5), rep(1, 18)))
ans <- ace(x, bird.orders, type = "d")
#### Showing the likelihoods on each node:
plot(bird.orders, type = "c", FALSE, label.offset = 1)
co <- c("blue", "yellow")
tiplabels(pch = 22, bg = co[as.numeric(x)], cex = 2, adj = 1)
nodelabels(thermo = ans$lik.anc, piecol = co, cex = 0.75)
### An example of the use of the argument 'ip':
tr <- character(4)
tr[1] <- "((((t10:5.03,t2:5.03):2.74,(t9:4.17,"
tr[2] <- "t5:4.17):3.60):2.80,(t3:4.05,t7:"
tr[3] <- "4.05):6.53):2.32,((t6:4.38,t1:4.38):"
tr[4] <- "2.18,(t8:2.17,t4:2.17):4.39):6.33);"
tr <- read.tree(text = paste(tr, collapse = ""))
y <- c(rep(1, 6), rep(2, 4))
### The default 'ip = 0.1' makes ace fails:
ace(y, tr, type = "d")
ace(y, tr, type = "d", ip = 0.01)
### Surprisingly, using an initial value farther to the
### MLE than the default one works:
ace(y, tr, type = "d", ip = 0.3)
add.scale.bar()
```

Usemos clusters y árboles para comprender las relaciones entre secuencias:

```
#==============================
#FILOGENIA CON SECUENCIAS DE ADN
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(ape)
library(Biostrings)
#Parte 1:
#Calculamos distancias entre
#secuencias inventadas
MD <- stringDist(c(
             "GGGCCCCC",
             "GGGTTCCC",
             "CGAATCGA",
```

```
             "GGATCCCC",
             "GGATTCGC",
             "CGAAATC",
             "ATTCCCC",
             "GATCCCC",
             "TACGCGA"))

MD
#Calculamos y dibujamos un árbol
 plot(hclust(MD, method = "single"))
#Parte 2:
trw <- bionj(MD)
plot(trw)
#
#Parte 2
#Ejemplo 1 de R
data(phiX174Phage)
phiX174Phage
plot(hclust(stringDist(phiX174Phage), method = "single"))
#Parte 3
#Ejemplo 2 de R
data(srPhiX174)
srPhiX174
MDP <- stringDist(srPhiX174[1:7], method = "quality",
                  quality = SolexaQuality(quPhiX174[1:7]),
                  gapOpening = -10, gapExtension = -4)
 plot(hclust(MDP, method = "single"))
```

114 ***Amovas***

Amova significa *analysis of molecular variance* y es un método para el estudio de la variación molecular dentro de una especie a lo largo de una distribución geográfica que se ha registrado.

Para correr una amova en *R* necesitamos bajar los siguientes paquetes: adegenet, pegas. Sobre WINDOWS, use el menú de *R* y/o el *Rcmdr* para bajarlos e instalarlos. Sobre Linux, bájelos a la carpeta *downloads*, sobre ella abra una terminal con derechos *su* e instálelos así:

```
R CMD INSTALL adegenet_1.2-4.tar.gz
R CMD INSTALL pegas_0.3-2.tar.gz
```

El algoritmo de la *amova* se basa en permutaciones, cuyo número se especifica en *nperm*. Se toma 100 para que no se demore demasiado, pero 1000 es un valor mucho mejor aunque es necesario tener paciencia con el algoritmo pues se demora.

```
#==============================
##AMOVA: EJEMPLO REAL
#Correr por partes
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1
library(pegas)
library(ape)
data(woodmouse)
#teclee q para salir de la ayuda
help(woodmouse)
#
```

```
#Parte 2
d <- dist.dna(woodmouse)
g <- factor(c(rep("A", 7), rep("B", 8)))
g
p <- factor(c(rep(1, 3), rep(2, 4), rep(3, 4), rep(4, 4)))
p
#Las muestras   de ratones fueron tomadas de dos localidades
#A y B, que se dividen a su vez cada una en otras dos
#sublocalidades, 1 y 2, 3 y 4.
tabla <-data.frame(g,p)
tabla
#Análisis jerárquico de dos niveles
amova(d ~ g/p, nperm = 100)
#Análisis no jerárquico, de un nivel
#Versión dividida
amova(d ~ g, nperm = 100)
#Versión subdividida
amova(d ~ p, nperm = 100)
```

El siguiente ejemplo inventado presenta un análisis jerárquico con 3 niveles:

```
#==============================
##AMOVA: EJEMPLO INVENTADO CON 3 NIVELES
#Corre muy despacio
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(pegas)
library(ape)
pop <- gl(64, 5, labels = paste("pop", 1:64))
region <- gl(16, 20, labels = paste("region", 1:16))
conti <- gl(4, 80, labels = paste("conti", 1:4))
tabla <-data.frame(conti,region,pop)
#Estructura geográfica
tabla
#Distancias genéticas tomadas de la distribución uniforme.
dd <- as.dist(matrix(runif(320^2), 320))
#Análisis de 3 niveles
amova(dd ~ conti/region/pop, nperm = 100)
#Análisis de 2 niveles
amova(dd ~ conti/pop, nperm = 100)
amova(dd ~ conti/region, nperm = 100)
#Análisis de 1 nivel
amova(dd ~ conti, nperm = 100)
amova(dd ~ region, nperm = 100)
amova(dd ~ pop, nperm = 100)
```

En el siguiente ejemplo corremos una amova para datos de presencia ausencia:

```
#==============================
#AMOVA DE DATOS DE PRESENCIA-AUSENCIA
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(ape)
```

```
library(pegas)
#Individuos (5) y sus características (12)
#Individuo x
x <- c(0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1)
#Individuo y
y <- c(1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1)
z <- c(1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1)
w <- c(0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 1)
t <- c(0, 1, 0 ,1, 0 ,1, 1, 1, 0, 0, 1, 0 )
tabla <- data.frame(x,y,z,w,t)
#La distancia se calcula entre filas, por lo tanto
#transponemos:
tabla <- t(tabla)
tabla
#Estructura geográfica de los 5 individuos
#en el orden en que aparecen en la tabla
zona <- c("A", "A","A","B","B")
#Calculamos la matriz de distancia euclídea
dd <- dist(tabla, method = "euclidean")
dd
#Amova por zona
amova(dd ~ zona, nperm = 100)
#Todos los datos:
tabla<- cbind(tabla,zona)
tabla
```

Las amovas también pueden correrse sobre secuencias de AND:

```
#==============================
#AMOVA CON SECUENCIAS DE ADN
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
library(Biostrings)
library(ape)
library(pegas)
#Parte 1:
#Calculamos distancias entre
#Secuencias inventadas (9)
secs <- c(
             "GGGCCCCC",
             "GGGTTCCC",
             "CGAATCGA",
             "GGATCCCC",
             "GGATTCGC",
             "CGAAATC",
             "ATTCCCC",
             "GATCCCC",
             "TACGCGA")
secs
#Calculamos la distancia entre secuencias
#Se compara base por base: si son iguales,
#se lleva cero. Si son diferenetes, uno.
#Y se suma. Para sacar la distancia euclídea,
#se saca la raíz cuadrada
```

```
MD <- stringDist(secs)
MD
MD <- sqrt(MD)
MD
#Calculamos y dibujamos un árbol filogenético
 plot(hclust(MD, method = "single"))
#
#Parte 2:
#Otro diagrama
trw <- bionj(MD)
plot(trw)
#
#Parte 3:
#Amova
#Procedencia  de los 9 individuos
#en el orden en que aparecen en secs
zona <- c("A", "A","A","B","B","A","A","B", "B")
amova(MD ~ zona, nperm = 100)
```

El paquete *pegas* es muy poderoso. Para ver todas sus funciones usamos el comando

```
library(help = pegas)
```

115 *Biodiversidad*

Una medida cuantitativa de la biodiversidad de un ecosistema se llama **índice de biodiversidad**, de los cuales el de *Shannon* es muy famoso. Para calcularlo usamos un paquete especialmente diseñado para agricultura: *agricolae*. Sobre Windows se baja de la *cran* y se instala con el menu de *R*. Sobre Linux, se baja de la *cran* y se instala con el comando:

```
R  CMD INSTALL agricolae_1.0-9.tar.gz
```

Una vez instalado *agricolae*, uno puede correr el siguiente programa que es una adpatación pedagógica de uno que viene con el paquete mencionado.

```
#==============================
#INDICE DE BIODIVERSIDAD DE SHANNON
#Limpia la memoria
rm(list = ls())
#Modo gráfico
par(mfrow=c(3,2), pch=16)
#Parte 1
#Literatura
library(agricolae)
#Teclee q para salir de la ayuda
help(agricolae)
#
#Parte 2
#Lista de funciones
#Teclee q para salir de la ayuda
library(help = agricolae)
#
#Parte 3
#Información sobre el Indice de Biodiversidad de Shannon
```

```
#Teclee q para salir de la ayuda
help(index.bio)
#
#Parte 4
data(paracsho)
paracsho
# date 22-06-05 and treatment CON = application with insecticide
specimens <- paracsho[1:10,6]
#Cálculo con boostrapping: modifique nboot
output1 <- index.bio(specimens,method="Simpson.Div",level=95,nboot=100)
output2 <- index.bio(specimens,method="Shannon",level=95,nboot=200)
rbind(output1, output2)
```

Capítulo 16

Epílogo

La estadística con *R* es una empresa de nunca acabar. Los que tengan mucho afán han escogido sufrir de estrés eterno. Lo mejor es disfrutar el trabajo continuo y permanente. Si se tiene esta perspectiva, se obtendrán muchos, muchísmos momentos felices.