USO DE HERRAMIENTAS INFORMÁTICAS PARA LA RECOPILACIÓN, ANÁLISIS E INTERPRETACIÓN DE DATOS DE INTERÉS EN LAS CIENCIAS BIOMÉDICAS

Módulo 3
Estadística Básica con R

Alfredo Rigalli
Maela Lupo
María Eugenia Chulibert
Mercedes Lombarte
Patricia Lupión

Centro Universitario de Estudios Medioambientales
Facultad de Ciencias Médicas
Universidad Nacional de Rosario

ISBN 978-987-86-0205-9

USO DE HERRAMIENTAS INFORMÁTICAS PARA LA RECOPILACIÓN, ANÁLISIS E INTERPRETACIÓN DE DATOS DE INTERÉS EN LAS CIENCIAS BIOMÉDICAS

Estadística Básica con R

Alfredo Rigalli
Maela Lupo
María Eugenia Chulibert
Mercedes Lombarte
Patricia Lupión

Centro Universitario de Estudios Medioambientales
Facultad de Ciencias Médicas
Universidad Nacional de Rosario

Uso de herramientas informáticas para la
recopilación, análisis e interpretación de
datos de interés en las ciencias biomédicas :
estadística básica con R / Alfredo Rigalli ...
[et al.]. - 1a edición para el alumno -
Rosario : Alfredo Rigalli, 2019.
Libro digital, PDF

Archivo Digital: descarga
ISBN 978-987-86-0205-9

1. Bioestadísticas. I. Rigalli, Alfredo.
CDD 610.28

AUTORES

Chulibert, María Eugenia: Licenciada en nutrición. Estudiante del doctorado en Ciencias Biomédicas de la Facultad de Ciencias Médicas de la Universidad Nacional de Rosario. Becaria doctoral del CONICET.

Lombarte, Mercedes: Licenciada en biotecnología y Doctora en Ciencias Biomédicas. Investigadora del Laboratorio de Biología Ósea y del Centro Universitario de Estudios Medioambientales y docente de la cátedra de Química Biológica de la Facultad de Ciencias Médicas de la Universidad Nacional de Rosario.

Lupión, Patricia: Licenciada en biotecnología. Estudiante del doctorado en Ciencias Biomédicas de la Facultad de Ciencias Médicas de la Universidad Nacional de Rosario. Becaria doctoral del CONICET.

Lupo, Maela: Licenciada en biotecnología y Doctora en Ciencias Biomédicas. Investigadora del Laboratorio de Biología Ósea y del Centro Universitario de Estudios Medioambientales y docente de la cátedra de Química Biológica de la Facultad de Ciencias Médicas de la Universidad Nacional de Rosario.

Rigalli, Alfredo: Bioquímico y Doctor en bioquímica. Investigadora del Laboratorio de Biología Ósea y del Centro Universitario de Estudios Medioambientales y docente de la cátedra de Química Biológica de la Facultad de Ciencias Médicas de la Universidad Nacional de Rosario. Investigador independiente del Consejo de Investigaciones de la UNR y del CONICET

Tabla de Contenidos

ORGANIZACIÓN DE LA OBRA

Esta obra está dividida en módulos y clases. Cada módulo agrupa temas diferentes. Brevemente

Módulo 1: introducción al manejo de objetos y funciones en R.

Módulo 2: introducción al uso de bibliotecas gráficas.

Módulo 3: introducción a la estadística básica.

Módulo 4: análisis multivariado de datos numéricos y análisis especiales de datos.

Módulo 5: desarrollo de scripts y programación en R.

Cada módulo se divide en 9 clases, las cuales constan de tablas específicas para cada clase, así como de un vídeo y una ejercitación. Al final de las 9 clases existe un examen final del módulo.

Las clases llevarán el nombre Clase1- seguido de un número de 1-9 si son clases del módulo 1, por ejemplo. Así tendrá clases Clase2-3, Clase4-1, etc según sean la clase 3 del módulo 2 o la clase 1 del módulo 4.

Las planillas de cálculo en formatos ods o xls llevarán la denominación tablaR1-3.ods por ejemplo si es la planilla de cálculo para la clase 3 del módulo 1. En el interior de la planilla hallará tablas con los nombre tablaR131, tablaR132, tablaR133, etc. Todas tablas para el módulo 1 (primer número), de la clase 3 (segundo número) y el tercer número indica el número de tabla. Con estos nombres serán introducidos como objetos en el espacio de trabajo.

Al principio de cada clase hallará un link al vídeo sobre la clase y tendrá un link a la planilla de cálculo con las tablas para el desarrollo de la clase.

1. Clase 3.1

Video: https://youtu.be/LK_Wu-Xe3fY
Tabla de datos: http://hdl.handle.net/2133/11555

En este módulo veremos pruebas estadísticas que nos permitirán contrastar una hipótesis contra otra.

Planteamos algunos ejemplos para comprender qué nos permitirá lo adquirido en este módulo

1- Deseamos comprobar si la media de las glucemias de un grupo de ratas es mayor que la media de las glucemias de otro grupo.

2- Deseamos comprobar si las variancias de las mediciones de las alturas de alumnos de un curso es igual o distinta de otro grupo de alumnos de la misma edad, pero de otra área del país.

3- Deseamos comprobar si la frecuencia de padecer osteoporosis es la misma o diferente entre dos grupos de mujeres que tienen dietas con diferente contenido de calcio.

4- Deseamos comprobar si existe alguna asociación entre los datos de índice de masa corporal de adolescentes de escuela secundaria de Rosario y la cantidad de vasos de gaseosa consumido por semana.

En general se contrasta la hipótesis llamada nula, con la alternativa. La hipótesis nula es que no existe la diferencia que si se busca demostrar con la hipótesis alternativa.

veamos en los ejemplos mencionados.

1- Deseamos comprobar si la media de las glucemias de un grupo de ratas es mayor que la media de las glucemias de otro grupo.

hipótesis nula: no existe diferencia entre las glucemias de ambos grupos de ratas.

Hipótesis alternativa: la media de la glucemia del grupo control es mayor que la del segundo grupo.

2- Deseamos comprobar si existe alguna asociación entre los datos de índice de masa corporal (IMC) de adolescentes de escuela secundaria de Rosario y la cantidad de vasos de gaseosa consumido por semana.

hipótesis nula: no existe asociación entre el IMC y la cantidad de gaseosa consumida.

hipótesis alternativa: existe asociación (positiva o negativa) entre la cantidad semanal de vasos de gaseosa y el IMC.

Luego de la prueba estadística llegaremos a una conclusión, pero la misma no estará libre de error. Sin embargo éste será conocido.

Cuando se contrasta una hipótesis contra la hipótesis nula se cometen dos tipos de errores y podremos conocer la probabilidad de estos errores al aplicar una prueba.

Error tipo 1 o alfa: Probabilidad de rechazar la hipótesis nula cuando en realidad no lo era.

En general fijaremos la probabilidad de este error en un valor de 0,05 o menos. El error de tipo 1 o alfa en R habitualmente aparece como p-value. Así, cuando realicemos una prueba de hipótesis si el p-value<0.05 rechazaremos la hipótesis nula, es decir rechazamos en general la falta de diferencia inclinándonos por la situación en que sí son diferentes. De una manera simplificada podemos interpretar el p-value como la probabilidad de haber optado por la hipótesis alternativa

cuando en realidad era cierta la hipótesis nula. En estas circunstancias, se suele interpretar como un falso positivo, es decir indicamos que algo es diferente cuando en realidad no lo era.

Error tipo 2 o beta: Probabilidad de quedarnos con la hipótesis nula cuando en realidad era correcta la hipótesis alternativa. Se interpreta como como una fasto negativo.

Con el valor de beta podremos calcular la potencia del ensayo, que es la probabilidad de haber aceptado la hipótesis alternativa cuando en realidad era cierta. Este valor lo fijaremos en 0,8 (es decir un 80%). La potencia nos mide la probabilidad de un verdadero positivo.

Importante: Siempre que concluyamos luego de aplicar un test estadísticos, sabemos que tenemos una probabilidad de haber dado la conclusión incorrecta y de haber dado la correcta.

1.1. Revisión

1.1.1. Introducción, auditoría de datos y estadística descriptiva

A continuación veremos las principales estadísticas descriptivas de una muestra y la forma de introducir datos en R y el control del procedimiento.
Supongamos que tenemos una muestra de datos en que se han medido diferentes variables. En este caso (ver tablaR3-1.xls/ods - hoja tablaR311) hemos medido en diferentes semillas diferentes variables: el tratamiento aplicado (tratados: t, controles: c), longitud de la raíz a los 7 días, en mm y si se produjo la germinación (si-no).
En primer lugar introducimos los datos. Tratándose de una tabla pequeña podemos hacerlo a través del portapapeles utilizando el siguiente código

```
> tablaR311<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")
> tablaR311
   numero germinacion tratamiento longitud
1       1          si           c     23.1
2       2          si           c     23.0
3       3          no           c       NA
4       4          si           c     22.9
5       5          si           c     21.0
6       6          si           c     21.8
7       7          si           c     20.0
8       8          no           c       NA
9       9          si           c     19.0
10     10          si           c     21.0
11     11          no           c       NA
12     12          si           c     22.0
13     13          si           c     22.0
14     14          si           c     21.9
15     15          si           c     23.2
16     16          no           c       NA
17     17          si           c     21.4
18     18          si           c     20.2
19     19          no           c       NA
20     20          si           c     20.8
21     21          si           c     20.0
22     22          si           c     20.0
23     23          si           c     20.0
24     24          no           c       NA
25     25          si           t     19.0
```

26	26	si	t	18.7
27	27	no	t	NA
28	28	si	t	13.0
29	29	si	t	13.0
30	30	no	t	NA
31	31	si	t	25.0
32	32	si	t	15.8
33	33	si	t	15.0
34	34	si	t	15.0
35	35	no	t	NA
36	36	no	t	NA
37	37	si	t	12.9
38	38	si	t	15.0
39	39	no	t	NA
40	40	si	t	15.0
41	41	si	t	15.0
42	42	si	t	12.8
43	43	si	t	13.0
44	44	no	t	NA
45	45	no	t	NA
46	46	si	t	14.0
47	47	si	t	14.0
48	48	no	t	NA
49	49	si	t	14.8
50	50	si	t	13.0

Auditoría de datos. Usted puede obviar la auditoría de los datos ingresados, pero tarde o temprano le dará el valor a este procedimiento.

1- Verificamos primero el tipo de objeto con la función str()

> str(tablaR311)

'data.frame': 50 obs. of 4 variables:

$ numero : int 1 2 3 4 5 6 7 8 9 10 ...

$ germinacion: Factor w/ 2 levels "no","si": 2 2 1 2 2 2 2 1 2 2 ...

$ tratamiento: Factor w/ 2 levels "c","t": 1 1 1 1 1 1 1 1 1 1 ...

$ longitud : num 23.1 23 NA 22.9 21 21.8 20 NA 19 21 ...

como podemos ver, nos indica que el objeto es un data.frame con 50 observaciones y 4 variables. Debaho con $numero, $germinación, etc nos indica cada variable si son números enteros (int) o reales (num). En caso que sean variables cualitativas nos indica la cantidad de niveles y sus denominaciones.

Aunque la función str() nos indica el tipo de objeto, la función is.data.frame() nos indica con TRUE si realmente es un data.frame y con FALSE si no lo es

> is.data.frame(tablaR311)

[1] TRUE

2- El tipo de variables: continuas, factores, caracter, numérica puede ser investigada con la función

summary()

```
> summary(tablaR311)
     numero       germinacion  tratamiento    longitud
 Min.   : 1.00   no:14       c:24        Min.   :12.80
 1st Qu.:13.25   si:36       t:26        1st Qu.:14.95
 Median :25.50                           Median :19.50
 Mean   :25.50                           Mean   :18.26
 3rd Qu.:37.75                           3rd Qu.:21.50
 Max.   :50.00                           Max.   :25.00
                                         NA's   :14
```

En aquellas variables como número y longitud que son números nos muestra mínimo, máximo, cuartilos, media y mediana. Si son factores no indica los niveles y la cantidad de unidades por nivel del factor. Analizando los datos anteriores, 24 unidades pertenecen al tratamiento "c" y 26 al tratamiento "t".

3- Dimensiones de la tabla. Podemos conocer el número de filas y columnas con las funciones nrow() y ncol()

```
> nrow(tablaR311)
[1] 50
> ncol(tablaR311)
[1] 4
```

1.1.2. Estadísticas descriptivas

Revisaremos la forma de calcular con R las principales estadísticas descriptivas de una muestra.

1.1.2.1. media

```
> mean(tablaR311$longitud,na.rm=TRUE)
[1] 18.25833
```

recuerde que cuando realiza una operación cualquiera involucrando datos NA, el resultado será NA. Para poder excluir los datos NA, debe incluir entre los argumentos de la función: na.rm=TRUE.

1.1.2.2. desvío estándar o desviación típica

```
> sd(tablaR311$longitud,na.rm=TRUE)
[1] 3.840266
```

1.1.2.3. variancia

```
> var(tablaR311$longitud,na.rm=TRUE)
[1] 14.74764
```

1.1.2.4. mediana

```
> median(tablaR311$longitud,na.rm=TRUE)
[1] 19.5
```

1.1.2.5. rango

```
> range(tablaR311$longitud,na.rm=TRUE)
```

[1] 12.8 25.0

1.1.2.6. percentilos

> quantile(tablaR311$longitud,na.rm=TRUE)

0% 25% 50% 75% 100%

12.80 14.95 19.50 21.50 25.00

si deseara un percentilo en particular, por ejemplo aquel valor de la variable que acumula el 30 % de valores menores a él, es decir el percentilo 30, aplicaremos

> quantile(tablaR311$longitud,probs=0.30,na.rm=TRUE)

30%

15

1.1.2.7. modo

Recuerde que R no tiene función que calcule el modo, es decir el valor que más veces se presenta. No confunda la función mode() con el cálculo del modo o moda.

Para calcular el modo, puede transformar los datos a caracter y factor y luego aplicar la función summary(). Allí le indica la cantidad de veces que se presenta cada valor.

Haremos el ejemplo con la columna longitud de la tablaR311. En primer lugar ordenamos los datos en forma creciente con la función sort(), luego los transformamos en caracter con la función as.character(). Luego los transformamos en factor con la función as.factor() y finalmente aplicamos la función summary() para obtener el resultado. Podemos hacer paso por paso o bien podemos aplicar todas las funciones de manera anidada.

> summary(data.frame(as.factor(as.character(sort(tablaR311$longitud)))))

nos mostrará cada valor con la cantidad de veces que se repite, ordenándolos según la cantidad de veces que se repite el valor

> summary(modo)
15 : 5
13 : 4
20 : 4
14 : 2
19 : 2
21 : 2
(Other):17

podemos ver entonces que la moda de los valores de longitud es 15 mm. Es decir que dentro de las plantas medidas 15 mm es la longitud que más veces se presenta, en este caso 5 veces. La longitud 13 le sigue en cantidad de veces que repite en la tablaR311

1.1.2.8. max

La función max() nos permite hallar el máximo de los valores dentro de un set de datos. Si aplicamos esta funcion a la columna longitud de la tablaR311

> max(tablaR311$longitud,na.rm=T)

[1] 25

nos indica que la longitud máxima entre todas las unidades experimentales es 25.

1.1.2.9. min

La función min() nos permite hallar el mínimo valor de la longitud

```
> min(tablaR311$longitud,na.rm=T)

[1] 12.8
```

1.1.2.10. intervalo intercuartilos

El intervalo intercuartilo es la diferencia de longitud entre el cuartilo 25 y el 75%

```
> IQR(tablaR311$longitud,na.rm=T)

[1] 6.55
```

Es una medida de dispersión y nos indica que el 50% de los datos que ocupa una zona central de la variable difieren en 6.55 entre el mínimo y máximo de la variable.

1.1.2.11. mediana de las desviaciones absolutas

Es una medida robusta de la dispersión de datos. Calcula la mediana de las desviaciones absolutas de cada valor respecto de un valor (center) No es tan influenciada como el desvío estándar por valores extremos. Tampoco es tan utilizada y en general en las ciencias biomédicas es más común utilizar el desvío estándar o el rango dependiendo la distribución de probabilidad de los datos. En el caso siguiente utilizamos como center la mediana

```
> mad(tablaR311$longitud,center=median(tablaR311$longitud,na.rm=TRUE),na.rm=T)

[1] 5.26323
```

y en el siguiente, la media

```
> mad(tablaR311$longitud,center=mean(tablaR211$longitud,na.rm=TRUE),na.rm=T)

[1] 4.979065
```

1.1.2.12. medias con peso

En algunos casos al calcular la media puede darse diferente peso a cada dato, es decir que cada dato no se promediará directamente sino teniendo en cuenta este peso. Esto se puede realizar en R con la función weighted.mean(). Aquellos valores para los cuales el peso es menor tienen menos influencia en el valor de la estadística calculado. Esto puede ser útil cuando las mediciones no son realizadas todas con el mismo error o hay valores muy dispersos. Deberá analizarse cuidadosamente la aplicación de este tipo de cálculo. Ante la duda es preferible inclinarse por la función mean() o median() que dan las medias o la mediana, respectivamente.

Veamos el ejemplo de la tabla tablaR312 de la planilla de cálculo tablaR3-1.xls/ods, en la cual para cada valor de la medición se le asignó un peso (no es competencia de este módulo conocer como se asignó dicho valor). Cuanto menor es el número de la columna peso, menos influirá dicho dato sobre la media

```
> tablaR312<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")
> tablaR312
  medicion       peso
1      0.1 0.004347826
2      0.2 0.008695652
3      0.2 0.008695652
4      0.2 0.008695652
```

5	0.4	0.017391304
6	0.5	0.021739130
7	0.5	0.021739130
8	23.0	1.000000000
9	21.0	0.913043478
10	20.0	0.869565217
11	20.0	0.869565217
12	19.0	0.826086956
13	15.0	0.652173913
14	12.0	0.521739130
15	9.0	0.391304348
16	5.0	0.217391304

Calculemos primero la media de los datos de la columna medición

```
> mean(tablaR312$medicion)
```

```
[1] 9.13125
```

calculemos ahora la media con pesos.

```
> weighted.mean(tablaR312$medicion,tablaR312$peso)
```

```
[1] 17.84251
```

vemos que la media calculada con esta función es más alta que con la función mean(), ya que los pesos asigna valores menores de peso a los valores menores de la medición. También podrían ser valores de peso al revés o cualquier otro modelo que se debiera aplicar. El peso que le asigne a cada valor saldrá de un análisis de cada situación experimental.

1.1.3. frecuencias absoluta

Supongamos que queremos conocer cuantas unidades experimentales de la tablaR311 tienen tratamiento (t) y cuantos (c), la función table() nos da esta información

```
> table(tablaR311$tratamiento)
c  t
24 26
```

Si deseáramos conocer el número de c y t, pero solo para aquellos en que se ha podido medir la longitud, el código siguiente nos permite excluir las unidades sin medición de longitud.

```
> table(tablaR311$tratamiento[tablaR311$longitud!="NA"])
 c  t
18 18
```

en este caso podemos decir que dentro de las unidades en que se midió la longitud 18 fueron de tratamiento "c" y 18 de tratamiento "t".

1.1.4. frecuencia relativa

La frecuencia relativa es el número de unidades experimentales que cumple con una condición dividido el número de unidades experimentales totales. Este dato se obtiene con la función prop.table()

```
> prop.table(table(tablaR311$tratamiento))
  c    t
0.48 0.52
```

El resultado nos indica la fracción de la unidad de cada tratamiento. Más claro es el porcentaje que

vemos a continuación.

1.1.5. porcentaje

Cuantas unidades experimentales sobre un total de 100, cumplen determinada condición se puede obtener también con la función prop.table(), pero multiplicada por 100.

```
> prop.table(table(tablaR311$tratamiento))*100
c  t
48 52
```

48 % de las unidades experimentales han sido controles (c) y 52 % tratadas (t)

1.1.6. tapply()

Esta función le permite calcular estadísticas por grupos. Veamos un ejemplo con datos de la tablaR311.

Supongamos que deseamos conocer la media de la longitud pero clasificando a las unidades por tratamiento (c o t). En la función tapply() indicaremos en el primer argumento la variable sobre la que queremos calcular la media, en este caso la longitud. En el segundo argumento incluimos el criterio de división de las unidades experimentales, en este caso el tratamiento. Además excluimos los valores NA para que pueda realizarse el cálculo

```
> tapply(tablaR311$longitud,factor(tablaR311$tratamiento),mean,na.rm=TRUE)
       c        t
21.29444 15.22222
```

así obtuvimos la media de las longitudes de los controles y los tratados. Si deseáramos calcular el desvío estándar aplicamos la siguiente estructura

```
> tapply(tablaR311$longitud,factor(tablaR311$tratamiento),sd,na.rm=TRUE)
       c        t
1.273934 3.036359
```

y si deseamos el rango, la siguiente línea de comandos.

```
> tapply(tablaR311$longitud,factor(tablaR311$tratamiento),range,na.rm=TRUE)

$c

[1] 19.0 23.2

$t

[1] 12.8 25.0
```

El resultado hallado nos indica que el rango para el grupo "c" es 19,0-23,2, mientras que para el grupo "t" los valores se hallan dentro del intervalo que tiene límites: 12,8 y 25,0

1.2. Revisión gráficas más comunes

En el módulo 2 se desarrollaron numerosas gráficas que permiten visualizar los datos y extraer conclusiones parciales o bien tener una idea más clara para el análisis de los datos. Si bien las gráficas y el tipo de análisis estadísticos de los datos queda fijado a la hora del diseño del experimento, la gráfica permitirán formar una idea más clara de los resultados del mismo.

Las gráficas más utilizadas a la hora de hacer un análisis de datos son:

scatterplot

boxplot

sectores

barras

histograma

No olvide toda la variedad de gráficas posibles que permitirán tener visiones parciales o totales de experimentos complicados.

Revisemos brevemente los códigos en su forma más simple de los cinco tipos de gráficas mencionadas.

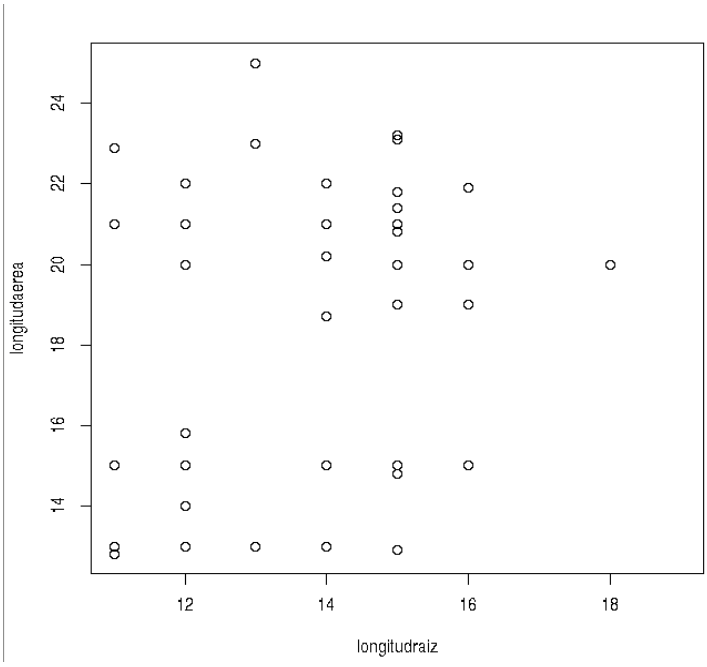
Introduzca los datos de la tabla tablaR313 de la planilla de cálculo de esta clase.

```
> tablaR313<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")
```

1.2.1. Scatterplot

La función scatterplot() nos da un gráfico de dispersión de puntos. Como hemos visto en el módulo 2, este puede ser modificado para darle mejor visibilidad. Acá lo veremos en su forma con mínimos argumentos.

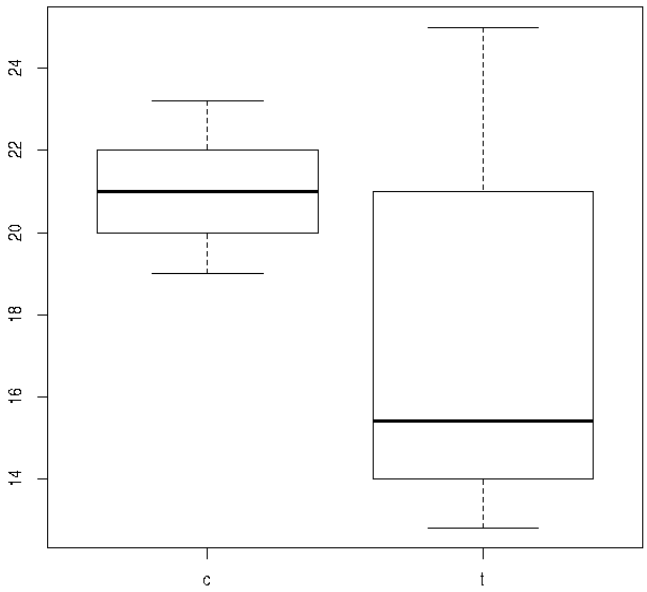
```
> plot(longitudaerea~longitudraiz,data=tablaR313)
```



1.2.2. Boxplot

La función boxplot() nos permite graficar los datos por grupos mostrando rango, cuartilos y mediana.

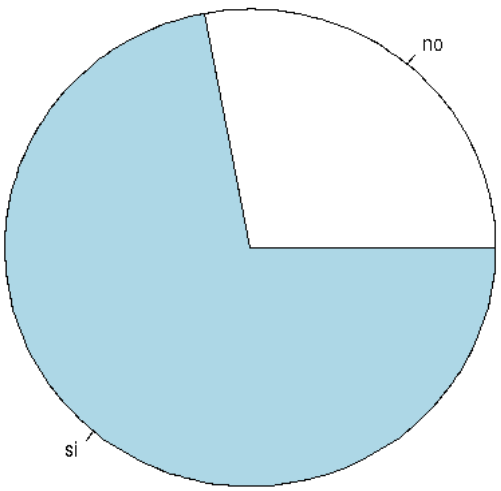
> boxplot(longitudaerea~tratamiento,data=tablaR313)



1.2.3. Sectores

La función pie() nos permite mostrar relación entre el número de individuos en uno y otro grupo.

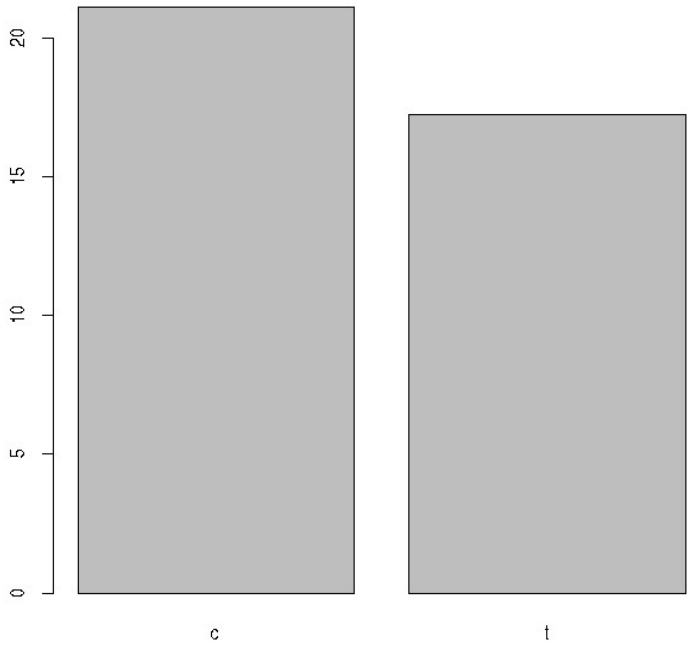
> pie(table(tablaR313$germinacion))



1.2.4. Gráficos de barras

El código siguiente nos muestra con rapidez y facilidad las medias de la variable longitudaerea de plantas agrupadas por tratamiento. La altura de la barra es el valor de la media.

> barplot(tapply(tablaR313$longitudaerea,tablaR313$tratamiento,mean))

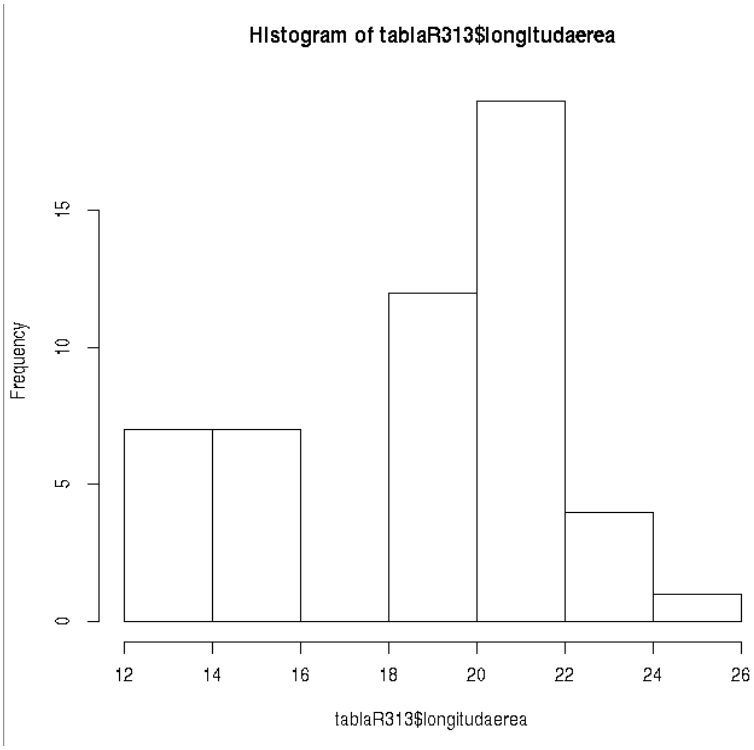


1.2.5. Histograma

La función hist() nos permite visualizar cuantos individuos de nuestra muestra caen dentro de un intervalo de valores de una de las variables medidas. En nuestro caso vemos la cantidad de individuos que caen dentro de intervalos de la función longitudaerea.

> hist(tablaR313$longitudaerea)

podemos ver claramente por ejemplo que hay 7 plantas que tienen longitudaerea entre 14 y 16 cm.



1.2.6. Datos apareados vs datos no apareados

Cuando disponemos de datos experimentales, por el diseño del experimento, estos datos pueden ser independientes (desapareados) o bien dependientes (apareados)

Veamos algunos ejemplos

La tablaR314 de la planilla de cálculo tablaR3-1.xls/ods muestra las glucemias de los animales de dos grupos experimentales: grupo1 y grupo2. Introduzca los datos en su espacio de trabajo. No copie la primer línea que da un detalle de los datos.

```
> tablaR314<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")

> tablaR314
  grupo1 grupo2
1    1.0    0.9
2    1.1    1.0
3    1.2    1.0
4    1.3    1.2
5    1.0    1.3
6    1.0    1.5
```

Cada grupo tiene 6 animales. Las glucemias fueron medidas en cada animal. Por ejemplo el animal 1 del grupo1 tuvo una glucemia de 1,0 mientras que el animal 1 del grupo2 tuvo una glucemia de 0,9. Estos datos son independientes, ya que los valores de glucemia mencionados corresponden a mediciones en diferentes animales.

Veamos ahora otra tabla de la misma planilla de cálculo. La tablaR315 tiene las glucemias de un grupo de animales a los que se les midió la glucemia al inicio de un experimento y al final del

mismo. Introducimos los datos.

```
> tablaR315<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")

> tablaR315
  tiempo0 tiempo30
1     1.0      1.0
2     1.0      1.2
3     1.2      1.3
4     1.0      1.5
5     0.9      1.0
6     0.8      0.9
```

En este caso el animal nº 1 tuvo al principio del experimento un valor de 1,0 y el mismo animal al finalizar el experimento luego de 30 días tuvo una glucemia de 1,0. Estos datos son apareados, ya que la glucemia fue medida en la misma unidad experimental.

Es clave determinar si los datos son independientes (desapareados) o dependientes (apareados) ya que los test estadísticos que se aplicarán serán diferentes en cada caso. Discuta esto hasta que se encuentre bien seguro de la elección del tipo de datos. Como regla práctica general decimos que si una misma variables fue medida dos o más veces sobre la misma unidad experimental los datos son dependientes o apareados. Contrariamente si una misma variable fue medida en unidades experimentales distintas los datos son no apareados o independientes. Pero no maneje mecánicamente estos conceptos, sino que someta siempre a discusión este tema tan básico e importante.

1.2.7. Distribución de probabilidad conocida o desconocida

Como hemos visto en le módulo 2, clase 5 (R2-5) los datos de una muestra pueden tener una distribución de probabilidad con una función conocida o no. Lo más común dentro de las funciones conocidas es la distribución normal. Por lo tanto es común que nos fijemos si una muestra de datos tiene o no distribución normal.

Conocer si una muestra tiene distribución normal o no es importante para la elección del test estadístico a aplicar para contrastar las hipótesis.

Para comprobar dicha distribución tenemos pruebas gráficas o analíticas

Distribución de probabilidad: identificación gráfica.

La forma gráfica de identificar la distribución de probabilidad es útil especialmente si tenemos un número grande de datos (podríamos decir, más de 50). En caso contrario deberemos recurrir a las pruebas analíticas. A continuación revisamos los conceptos discutidos en la clase R2-5

1.2.7.1. Histograma

La forma del histograma puede ayudarnos, aunque no darnos toda la respuesta, a decidir si una distribución es normal o no.

Introduzcamos los datos de la tablaR316 de la planilla de cálculo de esta clase. Esta tabla tiene datos de edad, sexo, peso, altura, ocupación y actividad física de estudiantes.

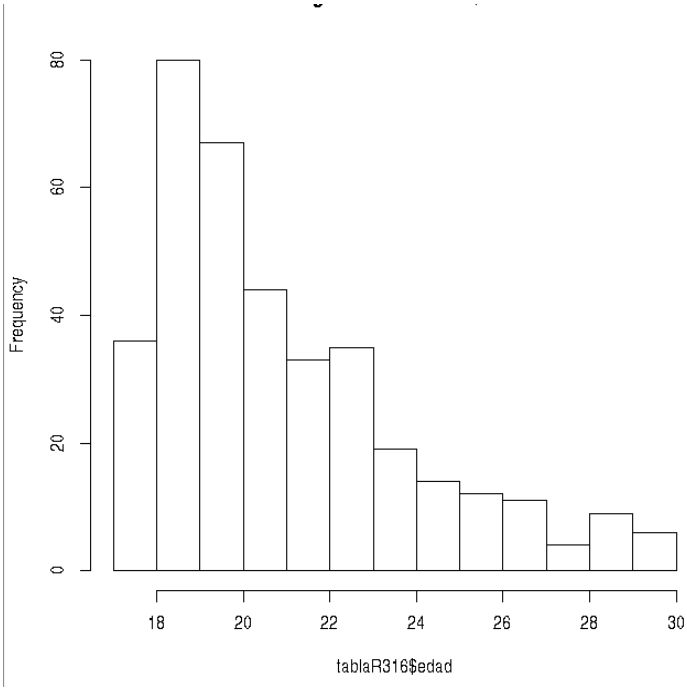
```
> tablaR316<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")

> tablaR316
   edad sexo  peso altura  ocupacion actividadfisica
1    26    f    66   1.74  estudiante              s
```

2	19	m	85	1.76	estudiante	s
3	20	f	55	1.72	estudiante	s
4	25	f	51	1.59	estudiante	s
5	23	m	100	1.84	estudiante	s
6	20	f	62	1.67	estudiante	s
.....						
366	20	f	56	1.64	estudiante	s
367	20	f	50	1.56	estudiante	s
368	19	f	59	1.70	estudiante	s
369	19	f	74	1.60	estudiante	n
370	20	f	64	1.63	estudiante	s

como recordará el código mínimo para obtener un histograma de distribución de los individuos por su edad es

```
> hist(tablaR316$edad)
```

que nos permite obtener la siguiente gráfica



Cuando esta gráfica tiene una distribución simétrica es probable aunque no necesariamente correcto, que la distribución sea normal. En este caso es bastante evidente que no es simétrica y posiblemente la distribución de probabilidad no sea normal.

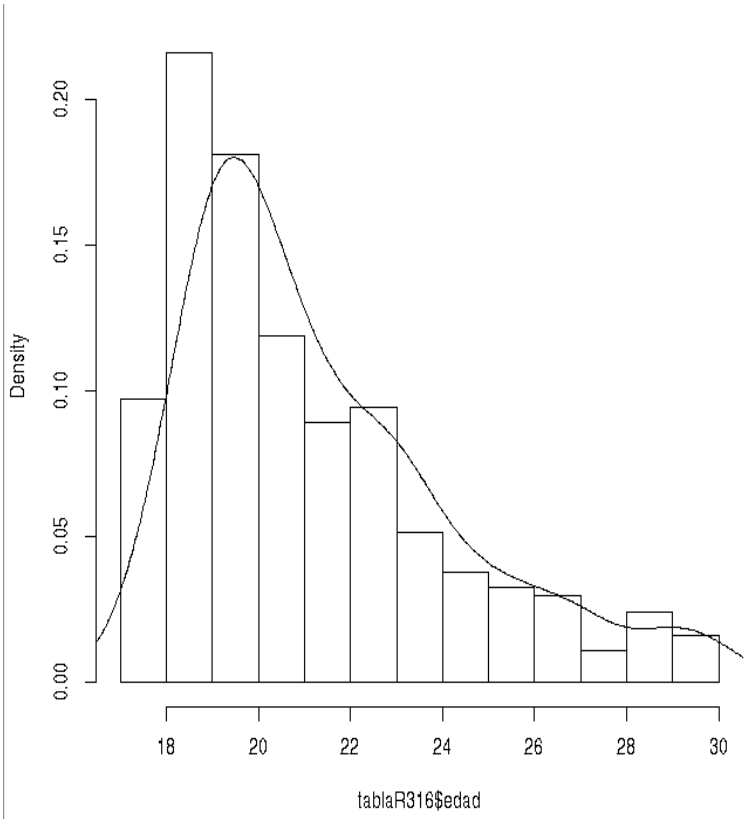
1.2.7.2. Histograma y density

La gráfica del histograma y la superposición de la función density() puede darnos una idea mejor de la distribución de probabilidad. Para ello ejecutamos los códigos

```
> hist(tablaR316$edad,freq=F)
```

```
> lines(density(tablaR316$edad))
```

que nos da la siguiente gráfica



Se observa un claro desplazamiento hacia la izquierda de la densidad de probabilidad, lo que podría estar indicando que en este caso los datos no tienen una distribución normal.

1.3. Steam and leaf

Esta representación también puede darnos información al respecto. Ya la hemos estudiado en el modulo 2 y no es una forma habitual de hallar en presentación y análisis de resultados en las ciencias biomédicas.

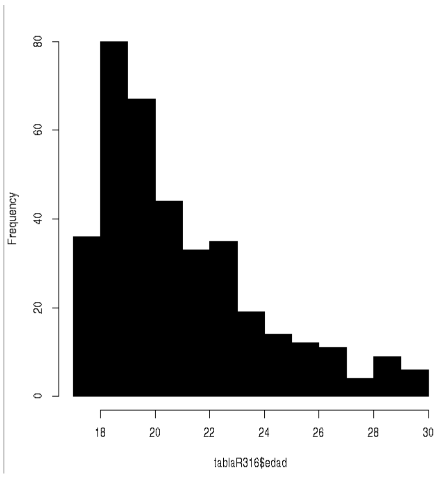
```
> stem(tablaR316$edad,scale=1)
  The decimal point is at the |

  17 | 00000
  18 | 00000000000000000000000000000000
  19 | 000000000000000000000000000000000000000000000000000000000000000000000000
  20 | 00000000000000000000000000000000000000000000000000000000000000000000000000
  21 | 0000000000000000000000000000000000000000000000
  22 | 00000000000000000000000000000000
  23 | 000000000000000000000000000000000000
  24 | 0000000000000000000
  25 | 00000000000000
  26 | 000000000000
  27 | 00000000000
```

28 | 0000
29 | 000000000
30 | 000000

esto nos indica por ejemplo que tenemos 6 alumnos con edad de 30 años y solo 4 con edad de 28 años, por ejemplificar algunos. Si se mira toda la salida en realidad se puede observar la misma distribución que se vería en el histograma. Si a la salida obtenida con R, le cambiamos el tamaño de letra, por ejemplo a 4-5 y lo ponemos en negrita, obtenemos

```
17 | 00000
18 | 0000000000000000000000000000000
19 | 00000000000000000000000000000000000000000000000000000000000000000000
20 | 0000000000000000000000000000000000000000000000000000000000000000000
21 | 00000000000000000000000000000000000000000000
22 | 000000000000000000000000000000000
23 | 00000000000000000000000000000000000
24 | 0000000000000000000
25 | 00000000000000
26 | 000000000000
27 | 00000000000
28 | 0000
29 | 000000000
30 | 000000
```

que podemos ver es igual aunque en formato horizontal que la distribución hallada con la función hist()



podría haber algunas diferencias debido a los argumentos include.lowest y right que no han sido utilizados.

1.3.1.1.   ecdf

La función ecdf() o empirical cummulative distribution hace la distribución acumulada de los datos en función del valor de la variable. Luego se puede dibujar sobre ella la línea que correspondería a una distribución normal acumulada tomando la media y el desvío estándar de los datos. Se evalúa a simple vista por la desviación de los puntos respecto de la línea. Si no coinciden los puntos con la recta, la muestra no tiene o se aleja de la distribución normal. No es una análisis común en las ciencias biomédicas, pero no por ello sin valor a la hora de determinar el tipo de datos.

tomamos la variable edad del data.frame tablaR316, con el que venimos trabajando

> plot(ecdf(tablaR316$edad))

calculamos el rango de edades

> range(tablaR316$edad)

[1] 17 30

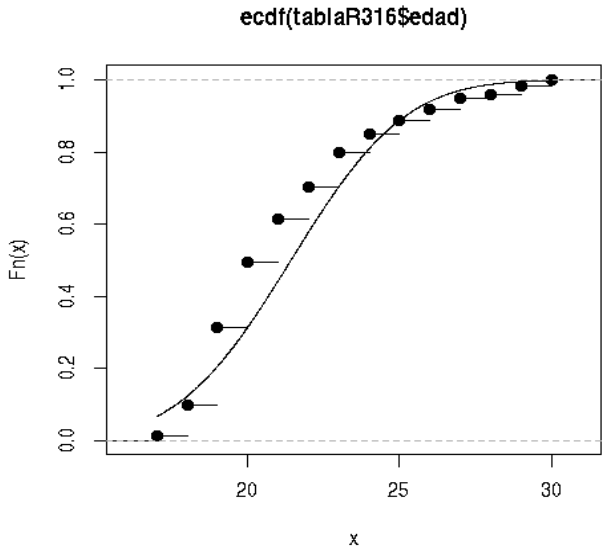
generamos un vector con ese rango y a saltos fijos: 0,01

> x<-seq(17,30,0.01)

luego graficamos la probabilidad acumulada si nuestros datos tuvieran distribución normal, para lo cual utilizamos entre otras la función pnorm()

> lines(x,pnorm(x,mean=mean(tablaR316$edad),sd=sd(tablaR316$edad)))

y obtenemos la siguiente gráfica



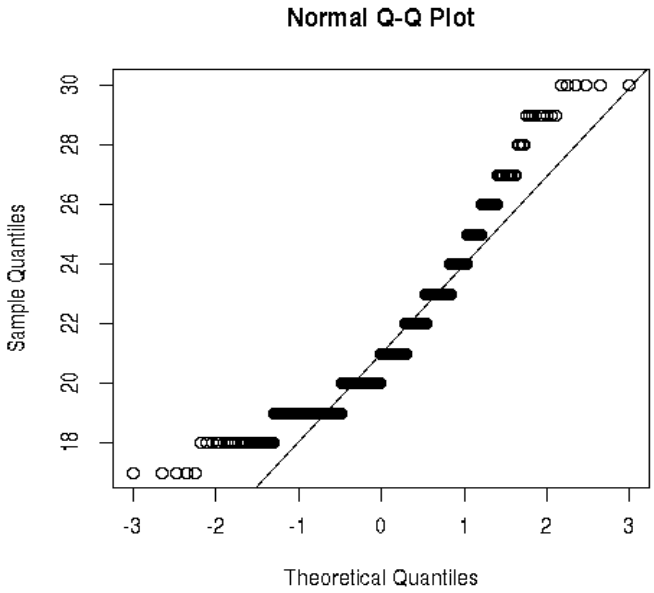
si los puntos coinciden con la línea es muy probable una distribución normal de los datos. En este caso es evidente no hay coincidencia.

1.3.1.2. Q-Q plots

Quantile – quantile plot. Permite también gráficamente ver si una distribución es o no normal. No es habitual ver el uso de estas funciones en nuestras áreas del conocimiento.

> qqnorm(tablaR316$edad)

> qqline(tablaR316$edad)



es evidente también con esta gráfica que la distribución de probabilidad de las edades de estos estudiantes no sigue una distribución normal. En ese caso debería haber una coincidencia entre los puntos y la línea.

2. Clase 3.2

Video: https://youtu.be/LlXPM_ZQkcM
Tabla de datos: http://hdl.handle.net/2133/11558

2.1. Definición y gráfica de funciones

Una función es una relación matemática entre variables. Introduciremos manejos básicos de funciones en R, donde una función se define como un objeto con un nombre y el siguiente código general

nombrefunción<-function(variable){expresión}

donde variable indica la o las variables independientes y expresión es la fórmula matemática que relaciona a las mencionadas variables independientes con la variable dependiente que queda definida con el nombre del objeto nombrefunción.

Por ejemplo si quisiéramos utilizar en R la función y=x+1, el código para introducir la función sería

> y<-function(x){x+1}

si le pedimos la función, vemos que quedó en nuestro espacio de trabajo

> y

function(x){x+1}

la función quedó dentro de nuestro espacio de trabajo como un objeto llamado "y"

Introduzca la hoja tablaR321 de la planilla tablaR3-2.ods/xls.

> tablaR321<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")

Creará el siguiente data.frame, con los códigos que usted ya conoce.

```
> tablaR321
   medicion condicion
1       2.0         o
2       2.1         o
3       2.2         o
4       2.3         o
5       2.4         o
6       2.5         o
7       2.6         o
8       2.7         o
9       2.8         o
10      2.0         b
11      2.1         b
12      2.2         b
13      2.0         b
14      2.1         b
15      2.0         b
16      1.9         b
17      2.0         b
```

veamos la columna medición

> tablaR321$medicion

[1] 2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.0 2.1 2.2 2.0 2.1 2.0 1.9 2.0

ahora apliquémosle la función y(x). que hemos definido. Esto implica a cada valor de la columna medición, sumarle 1. Para ello utilizamos el siguiente código.

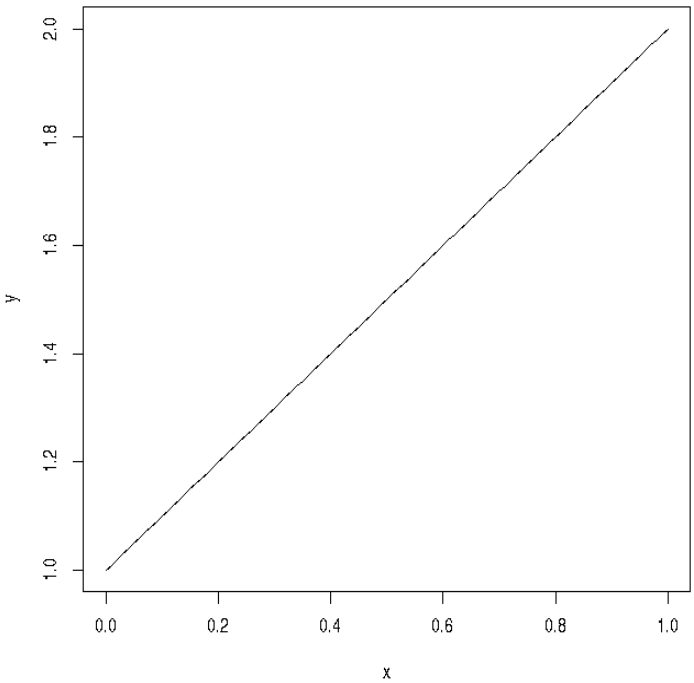
> y(tablaR321$medicion)

[1] 3.0 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.0 3.1 3.2 3.0 3.1 3.0 2.9 3.0

vemos que obtuvimos el resultado deseado. Para el primer valor de tablaR321$medición que tiene el valor 2.0, al aplicarle la función obtuvimos 3.0, y así sucesivamente.

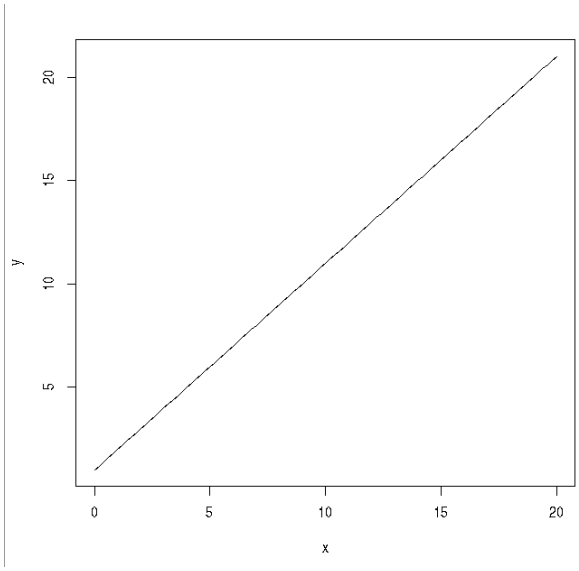
si quisieramos graficar la función y, aplicamos el código

> plot(y)



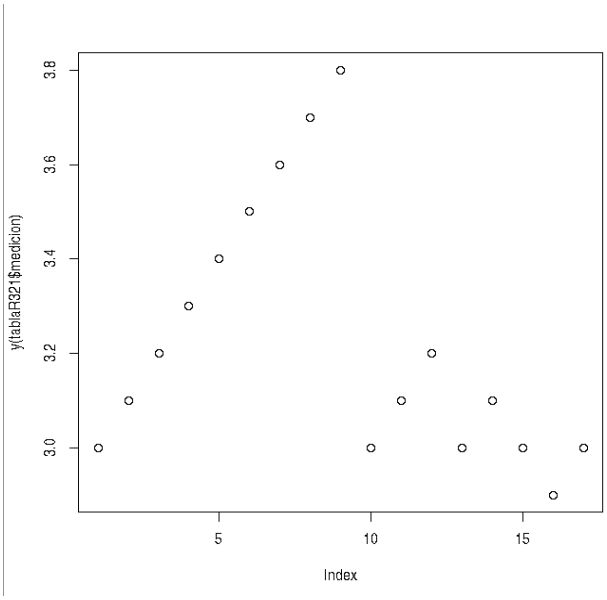
Con el código anterior, R por defecto grafica la función en el intervalo [0,1], pero si quisiéramos un intervalo determinado, se agrega el mismo al código. Por ejemplo si queremos graficar en el intervalo [0,20], utilizamos

> plot(y,0,20)



Si quisíeramos graficar la función pero con los datos de nuestra tabla, utilizamos

> plot(y(tablaR321$medicion))



utilizaremos estos conceptos en el futuro.

2.1.1. La función normal

Como vimos en el ítem anterior, y= x+ 1 es una función. Tiene variables: ***y*** se considera la variable dependiente, ya que su valor dependerá de que valor tome la variable dependiente ***x***. Por su parte x es la variable independiente. Esta denominación es arbitraria, pero en general se considera así. x e y son símbolos para identificar las variables, pero bien podría ser cualquier otro símbolo. Por otra parte aparecen dos números: un "1" sumando a la x y otro "1" que multiplica a la x, que no aparece

en la expresión por razones obvias. Estos dos números son los parámetros, valores fijos que no cambiarán, al menos en este caso. Veremos más adelante que estos parámetros pueden llegar a ser variables que adoptan valores óptimos para cada set de datos.

Pasemos ahora a una función un poco más compleja en su expresión matemática, pero que recibirá el mismo trato que cualquier función: la función normal.

La función normal es una función cuyos parámetros son la media y el desvío estándar (SD en nuestro caso) de un set de datos. Esta función responde a la forma

$$f(x)=\frac{1}{SD*\sqrt{2*\pi}}*e^{-\frac{1}{2}*\left(\frac{x-media}{SD}\right)^2}$$

Introducimos en nuestro espacio de trabajo esta función con el siguiente código

```
f<-function(x){(1 / (SD * sqrt(2 *pi))) * exp (-(1/2)*((x - media) /  SD)^2)}
```

hemos creado un objeto f, que podemos pedir para comprobar

```
> f
function(x){(1 / (SD * sqrt(2 *pi))) * exp (-(1/2)*((x - media) /  SD)^2)}
```

La variable independiente es "x" y los parámetros son "SD" y "media". La variable dependiente es en nuestra expresión f, que bien podríamos haberle puesto "y" o cualquier otro nombre. Se utiliza a menudo f(x) para indicar que el valor de ella depende de x y se lee "f de x = efe de equis".

En este caso utilizamos x como variable independiente, pero bien podría ser escrita con cualquier otros símbolo, por ejemplo (no introduciremos esta forma en nuestro espacio)

```
f<-function(W){(1 / (s * sqrt(2 *pi))) * exp (-(1/2)*((W - u) /  s)^2)}
```

donde la variable independiente la llamamos W, y los parámetros u y pi.

Veamos como graficar una función en R.

grafiquemos ahora la función normal con los siguientes parámetros

media=10

SD= 1

en el intervalo de x [0,20]

en primer lugar debemos crear la variable media y SD

```
> media<-10
> SD<-1
```

podemos generar un vector x, con valores como hemos aprendido en módulos anteriores. Con el código siguiente generamos una serie de números que van entre 0 y 20 y tienen un escalón o diferencia entre ellos de 0,01.

```
> x<-seq(0,20,0.01)
```

El código anterior nos genera un vector de 2000 datos, que podemos ver con

>x

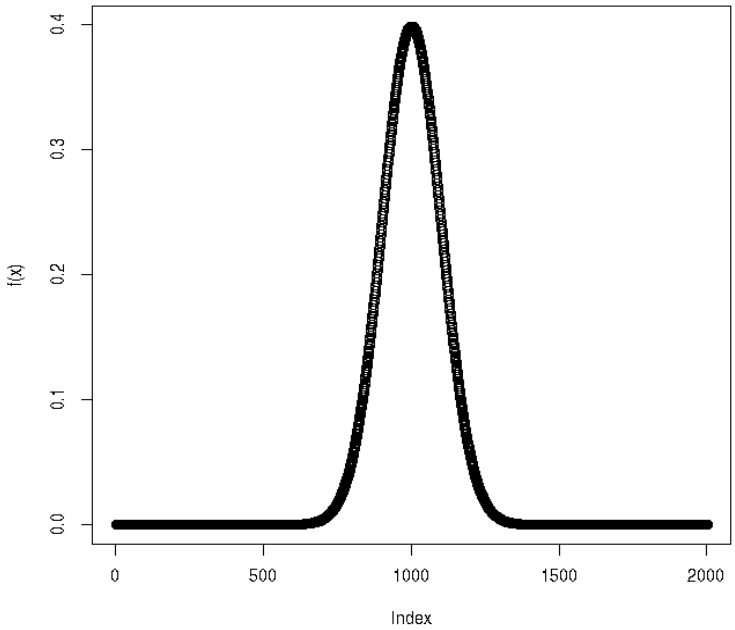
luego aplicamos la función f definida anteriormente al vector x, recientemente definido

> f(x)
  [1] 7.694599e-23 8.503421e-23 9.396325e-23 1.038195e-22 1.146981e-22
  [6] 1.267040e-22 1.399526e-22 1.545710e-22 1.706993e-22 1.884916e-22
 [11] 2.081177e-22 2.297642e-22 2.536369e-22 2.799619e-22 3.089884e-22
 [16] 3.409902e-22 3.762 ..... sigue

para graficar, podemos utilizar estos valores

> plot(x,f(x))

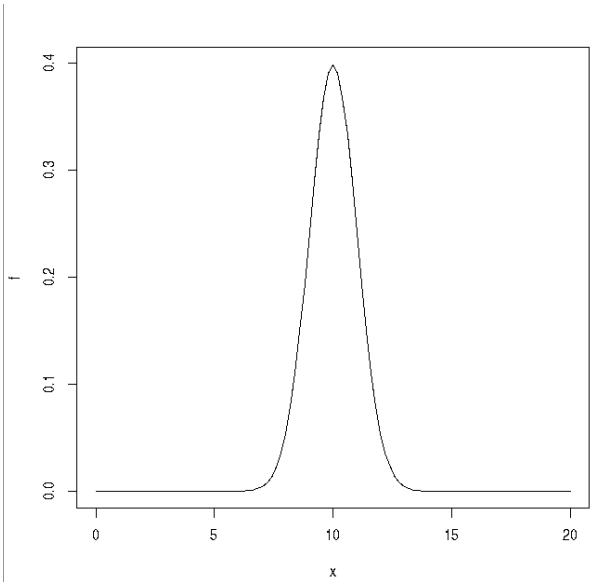
obtendremos



también podríamos hacerlo con

> plot (f,0,20)

obtendríamos



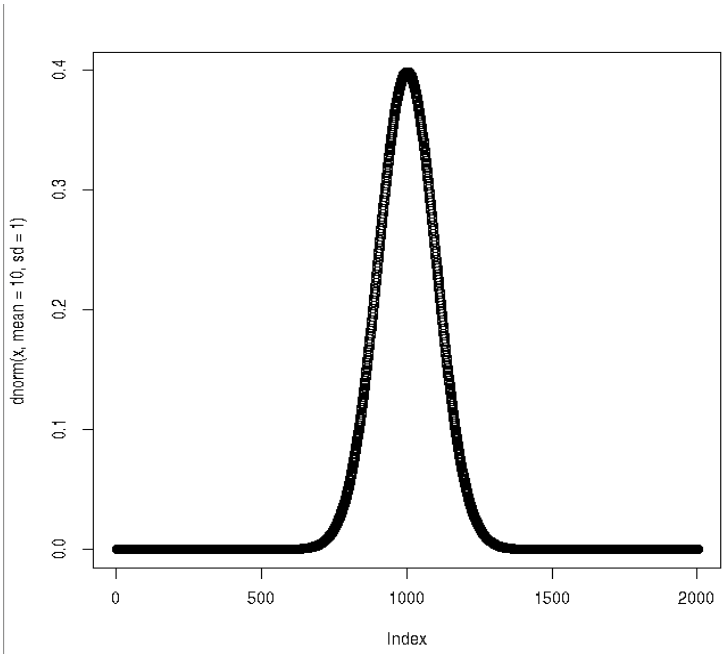
también se puede hacer con la función dnorm(), que es la función normal que R ya trae incorporada con el siguiente formato general

dnorm(x,mean,sd)

veamos para nuestro caso

> plot(dnorm(x,mean=10,sd=1))

obtendremos la gráfica



Pruebe a introducir y graficar las siguientes funciones

$y= 2*x^2 + 2$

$z= 3 + 2*x + 1/x$

Hemos dedicado un tiempo a la función normal dato que gran cantidad de variables que enfrentamos en las ciencias biomédicas tienen esta distribución. Para la aplicación de pruebas que nos permitan hacer una inferencia estadística, en las ciencias biomédicas es común que primero verifiquemos si nuestros datos tienen o no distribución normal. En base a esta prueba haremos la elección de las pruebas estadísticas a aplicar.

2.1.2. Pruebas para evaluar distribución de probabilidad

En la clase 1 de este módulo desarrollamos técnicas gráficas para identificar distribuciones de probabilidad, especialmente para decidir si la distribución es o no normal.
Recuerde que el conocimiento de la función de distribución de probabilidad de una variable condiciona la elección de los test estadísticos a aplicar para realizar el ensayo de hipótesis, es decir seleccionar si nos inclinaremos por quedarnos con la hipótesis nula o la alternativa.

En esta clase veremos pruebas analíticas que nos permiten decidir si nuestra muestra tiene o no distribución normal. Estas pruebas analíticas nos liberan de la subjetividad que puede tener una decisión tomada en base a la observación de una gráfica.
Generaremos primero dos grupos de datos. Uno con distribución normal y otro sin ella. Para ello utilizaremos funciones conocidas
El siguiente código generará una set de 100 datos con distribución normal de media = 5 y desvío estándar = 0,1. La función rnorm() utilizada a continuación es una función de R que le permite generar aleatoriamente datos con distribución normal con un dado valor de media y de sd,

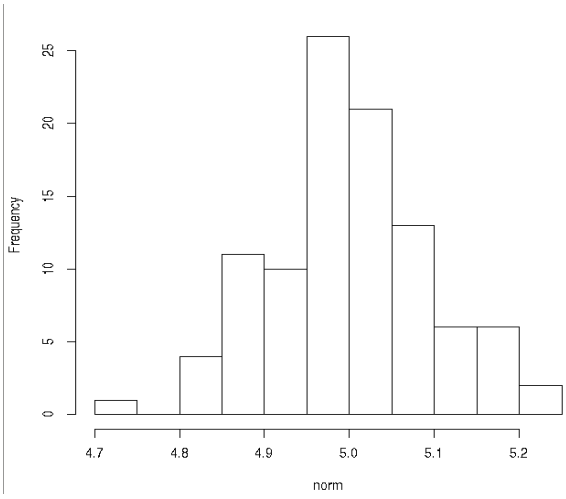
```
> norm<-rnorm(100, mean = 5, sd = 0.1)
```

por lo tanto en el objeto norm, tendremos ahora 100 datos aleatorios pero que tienen distribución normal. Contrariamente la función runif() genera datos aleatorios con distribución uniforme
El siguiente código generará un set de 100 datos aleatorios comprendidos entre 1 y 50

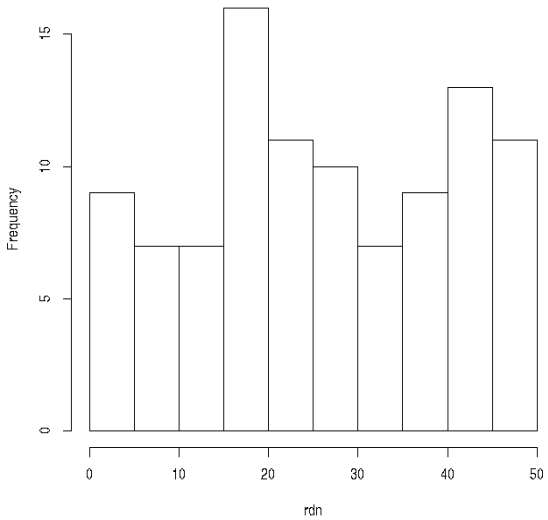
```
> rdn<-runif(100,1,50)
```

así tenemos dos objetos con datos: norm y rdn. El primero con distribución normal y el otro no. No se olvide de ellos ya que los utilizaremos en adelante en esta clase.
Por la forma de generarlos tenemos clara la distribución, norm debería distribuirse con forma de campana característico de una distribución normal y rdn de manera uniforme. Veamos los histogramas que nos permitirán en parte verificar esa conclusión

```
> hist(norm)
```



> hist(rdn)



si ajustamos a cada histograma una función normal con la media y sd de cada muestra de datos veremos claramente que se asemeja mucho más en el primer caso.

La función normal fue definida como

f<-function(x){(1 / (SD * sqrt(2 *pi))) * exp (-(1/2)*((x - media) / SD)^2)}

donde x es la variable independiente

media y SD los parámetros

generamos un vector x en el rango [0,50], que corresponde al rango de nuestro objetos norm y rdn

> x<-seq(0,50,0.1)

calculamos los parámetros con los datos del objeto rdn

> media<-mean(rdn)

> SD<-sd(rdn)

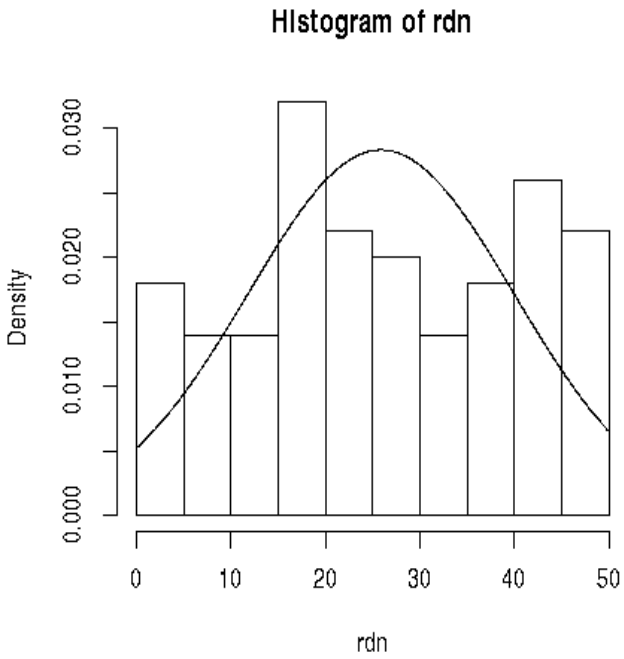
graficamos la distribución de probabilidad con los datos

> hist(rdn,freq=F)

graficamos la función normal con los parámetros obtenidos

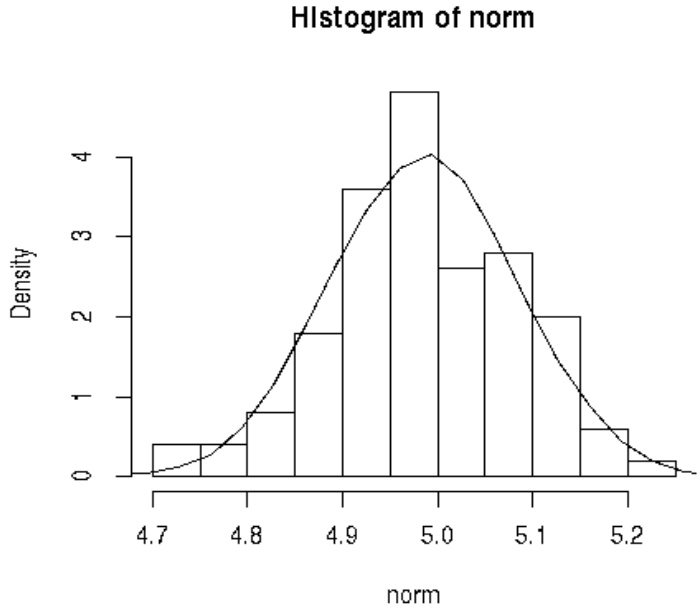
> lines(spline(x,f(x)))

obtenemos



veamos el otro set de datos, contenidos en el objeto norm.

```
> hist(norm,freq=F)
> media<-mean(norm)
> SD<-sd(norm)
> lines(spline(x,f(x)))
```



claramente los datos del objeto norm se ajustan mejor a la distribución normal que los rdn.

Veamos ahora como analizar esto sin necesidad de nuestra subjetividad, sino a través de un ensayo de hipótesis.

2.1.3. Pruebas analíticas de distribución normal

2.1.3.1. Prueba de Shapiro Wilk

Analizaremos si los datos norm y rdn obtenidos anteriormente se pueden considerar con distribución normal o no. La prueba de Shapiro Wilk nos permite tomar una decisión sobre su distribución de probabilidad y es recomendada para variables continuas con gran número de datos. Si el número de datos es menor a 50, esta prueba es recomendada por sobre la prueba de Kolmogorov Smirnov que veremos a continuación.

El código para aplicar esta prueba es

```
> shapiro.test(norm)
        Shapiro-Wilk normality test
data:  norm
W = 0.98787, p-value = 0.4986
```

La hipótesis nula es que la distribución analizada tiene distribución normal. Si p-value fuese menor a 0.01, no tendría distribución normal. Como p-value=0.4986, la probabilidad que tenga distribución normal bajo la hipótesis nula es de aproximadamente el 50%, siendo tan alta la probabilidad aceptamos que es normal. El valor de p-value puede cambiar respecto del mostrado en este texto, ya que la función rnorm() utilizada anteriormente genera datos aleatorios que seguramente no serán los mismos en el momento que usted los realice comparados con los realizados en este texto.

En síntesis: cuando p-value es mayor a 0.01 aceptamos que es normal. Podría fijar otro valor de corte para p-value. En las ciencias biomédicas es común utilizar 0,01 o 0.05 dependiendo la situación, tema que discutiremos más adelante. Si el p-value luego de aplicar el test es menor a 0.01, entonces decimos que la muestra no tiene distribución normal.

Aplicamos ahora la prueba de normalidad al set de datos rdn

```
> shapiro.test(rdn)
        Shapiro-Wilk normality test
data:  rdn
W = 0.95451, p-value = 0.001665
```

La hipótesis nula es que la distribución analizada tiene distribución normal. Si p-value fuese menor a 0.01, no tendría distribución normal. Como p-value=0.001665, la probabilidad que tenga distribución normal bajo la hipótesis nula es de aproximadamente el 0,1%, siendo tan baja rechazamos la hipótesis nula y aceptamos que la distribución de probabilidad no es normal.

En síntesis: como p-value es menor a 0.01 rechazamos la hipótesis que la distribución sea normal, diciendo que nuestros datos no tienen distribución normal.

2.1.3.2. Prueba de Kolmogorov Smirnov

Esta prueba es de utilidad cuando se desea comparar si la distribución de probabilidad de dos muestras discrepa o no (Two sample test) o bien para comprobar si una muestra tiene o no distribución normal (one sample test). Este último no es recomendado, y es preferible utilizar Shapiro test. Esta prueba es adecuada si las variables son continuas y el número de datos supera a 50.

Se puede utilizar también para comprobar normalidad u otras formas de distribución de

probabilidad menos comunes como la distribución gama.

2.1.3.3. Comparación de dos muestras

Deseo comparar si las muestras de datos generadas anteriormente: norm y rdn provienen de la misma distribución de probabilidad. Utilizamos ahora el test de Kolmogorov-Smirnov, que se hace con la función ks.test()

```
> ks.test(norm,rdn)
        Two-sample Kolmogorov-Smirnov test

data:  norm and rdn
D = 0.91, p-value < 2.2e-16
alternative hypothesis: two-sided
```

La hipótesis nula es que las muestras norm y rdn, sí provienen de la misma distribución de probabilidad. Bajo esta hipótesis como p-value es mucho menor que 0,01, puedo concluir que es muy baja la probabilidad que ambas provengan de la misma distribución. En conclusión tiene distribuciones de probabilidad diferentes.

2.1.3.4. Comprobar si una muestra proviene de distribución normal

Para utilizar ks.test() para probar si una muestra de datos, por ejemplo norm, tiene distribución normal aplicamos el código que se muestra a continuación. Para esta comprobación se utiliza el objeto que tiene los datos, en este caso norm. La distribución de probabilidad que quiero probar si se ajusta (en este caso normal, con "pnorm") y los parámetros de la distribución normal, en este caso mean y sd del set de datos a estudiar.

```
> ks.test(norm,"pnorm",mean(norm),sd(norm))
        One-sample Kolmogorov-Smirnov test
data:  norm
D = 0.059251, p-value = 0.874
alternative hypothesis: two-sided
```

Como el valor de p-value > 0.01 entonces acepto que la muestra de datos norm tiene distribución normal.

Si ahora aplico la prueba a los datos rdn, que gráficamente había comprobado que distaba de tener distribución normal

```
> ks.test(rdn,"pnorm",mean(rdn),sd(rdn))
        One-sample Kolmogorov-Smirnov test
data:  rdn
D = 0.090957, p-value = 0.3797
alternative hypothesis: two-sided
```

p-value >0,01 acepto que tiene distribución normal.

Según este test, ambas muestras tendrían distribución que no discrepa de ser normal. Conclusión a la que no arribé utilizando Shapiro.test.

El diseñador del test no recomienda el uso de ks.test para una sola muestra, con media y sd calculados a partir de la muestra de datos. En conclusión si desea verificar si una muestra proviene de una población normal utilice Shapiro.test() y reserve ks.test() para comparar si dos muestras de datos provienen o no de una misma distribución de probabilidad.

3. Clase 3.3

Video: https://youtu.be/Tf7_Sacj_lQ
Tabla de datos: http://hdl.handle.net/2133/11559

3.1. Prueba de aleatoriedad

Al tomar una serie datos a partir de un grupo de unidades experimentales, estos pueden ser tomados aleatoriamente o bien no cumplir con este criterio. La aleatoriedad es habitualmente una condición indispensable para llegar a conclusiones adecuadas en muchos caso. Por supuesto que no siempre debe ser aleatorio y podrán existir situaciones en que no sea un requisito, pero la misma deberá estar muy fundamentada.

Pongamos algunos ejemplos

Supongamos que desea conocer la media o mediana (dependerá de la distribución de probabilidad, cual de las dos estadísticas me convenga) de la edad de los alumnos de la carrera de medicina de Rosario, UNR. La facultad de Medicina de Rosario (UNR) tiene concentrado a segundo año de la carrera en la planta baja de un edificio, el cual en el segundo y en el tercer piso tiene a primer año y tercer año, respectivamente. Si considero que con 100 alumnos tendré una muestra representativa y cuando hago la encuesta llego a ese número de alumnos solo habiendo estado en la planta baja, mi muestra no es aleatoria para la carrera de medicina y por supuesto no será representativa.

Otro ejemplo: medimos el voltaje generado por un electrodo sobre una misma muestra a lo largo del tiempo. La medición es continua, podría ocurrir que cada medición esté condicionada por la medida anterior, por lo que las mediciones no serían aleatorias.

Existen diferentes pruebas que podrán orientarnos cuando tengamos dudas respecto de la aleatoriedad de los datos o de la relación que puede existir entre ellos. Veremos las pruebas de autocorrelación y el test de rachas

3.1.1. Autocorrelación

Supongamos que tenemos valores medidos con un sensor en presencia de la sustancia: condición= "o" y en presencia de otra sustancia, condición = "b", como muestra la tablaR333 de la planilla de cálculo tablaR3-3.xls/ods. ¿Los datos obtenidos son completamente aleatorios en su distribución o siguen alguna tendencia? Cada dato parece o no estar condicionado o relacionado al anterior.

Es de utilidad en estos casos la autocorrelación de Ljung-Box. Este test prueba autocorrelación de primer orden (si analiza el dato x_n con el dato x_{n+1}.) será de segundo orden si (si analiza el dato x_n con el dato x_{n+2}.) y así sucesivamente. En el ejemplo analizaremos la autocorrelación de primer orden.

Analicemos en primer lugar los datos de las mediciones con la condición "o"

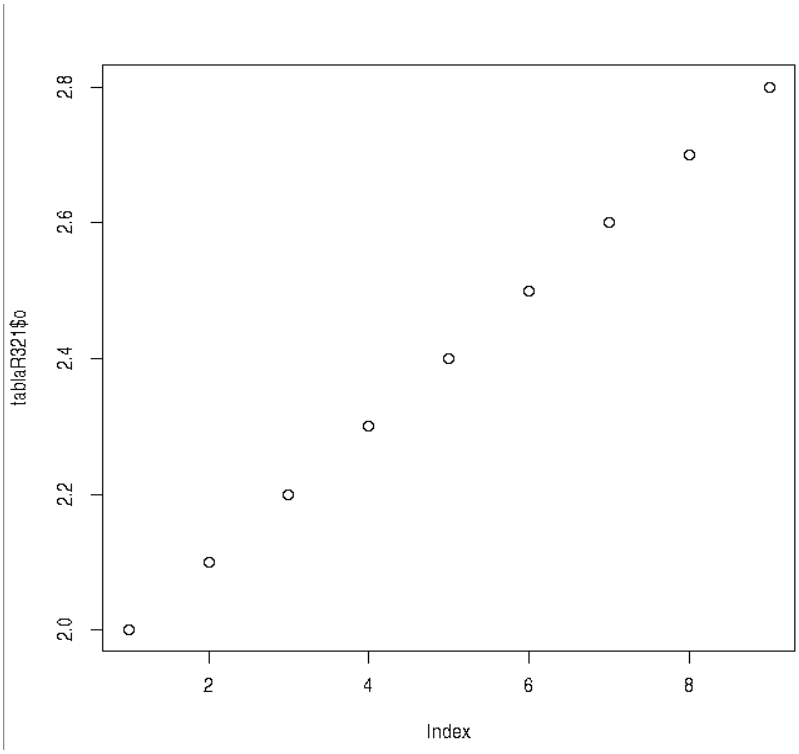
Introducimos la tabla como ya conocemos

```
tablaR333<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")
```

Haga la auditoría de los datos para comprobar que el ingreso fue correcto.

Si graficamos lo valores en función de su orden en la tabla

```
> plot(tablaR333$medicion[tablaR333$condicion=="o"])
```



vemos una clara asociación del dato con el orden en que fue tomado. No siempre es tan evidente, especialmente cuando el número de datos es muy grande.

La prueba analítica nos muestra los mismo, pero prescindiendo de la imagen.

```
> Box.test(tablaR333$medicion[tablaR333$condicion=="o"],lag=1,type=c("Ljung-Box"))
        Box-Ljung test
data:  tablaR333$medicion[tablaR333$condicion == "o"]
X-squared = 5.5, df = 1, p-value = 0.01902
```

En esta prueba contrastamos la hipótesis nula (llamémosla Ho)

Ho: no hay autocorrelación entre los datos

con la hipótesis alternativa (llamémosla H1)

H1: existe autocorrelación entre los datos.

Luego de realizar la prueba observamos que el valor p-value= 0.01902

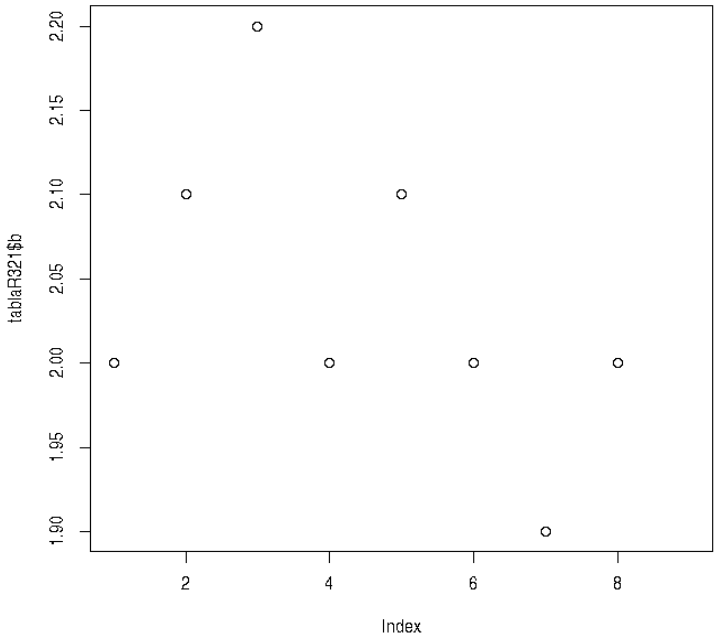
¿Qué significa?

Que si no existiera autocorrelación entre los datos, la probabilidad de hallar este grado de autocorrelación es de 0,01902. En otras palabras bajo la hipótesis que no existiera autocorrelación, hallar una autocorrelación como la hallada entre los datos o mayor sería de 1,9%. Siendo tan baja la probabilidad de hallar este valor, no existiendo correlación, concluimos que si hay autocorrelación.

Simplificando la conclusión: si p-value<0,01 nos indica que hay autocorrelación entre los datos. Es decir cada valor está de alguna manera condicionado o relacionado al anterior y al siguiente.

Veamos ahora los datos con la condición "b" de la misma hoja.

> plot(tablaR333$medicion[tablaR333$condicion=="b"])



la gráfica no muestra una tendencia o dependencia como en el caso anterior, o al menos no tan marcada. A simple vista sería difícil predecir autocorrelación entre los datos. Vemos a continuación el análisis con la función Box.test()

> Box.test(tablaR333$medicion[tablaR333$condicion=="b"],lag=1,type=c("Ljung-Box"))

Box-Ljung test

data: tablaR333$medicion[tablaR333$condicion == "b"]

X-squared = 0.17857, df = 1, p-value = 0.6726

Conclusión: Bajo la hipótesis que no existe autocorrelación, la probabilidad de hallar un valor como el hallado con nuestros datos es de 0,67 (67%). Siendo una probabilidad tan alta, concluimos que no existe autocorrelación.

Simplificando: p-value>0.01, no existe autocorrelación.

3.1.2. test de rachas

Este test consiste en medir el número de rachas, entendiéndose por racha un grupo de valores de igual signo interrumpido por uno de signo contrario. Si no son de signos contrarios se puede establecer que este por encima o debajo de la mediana, por ejemplo.

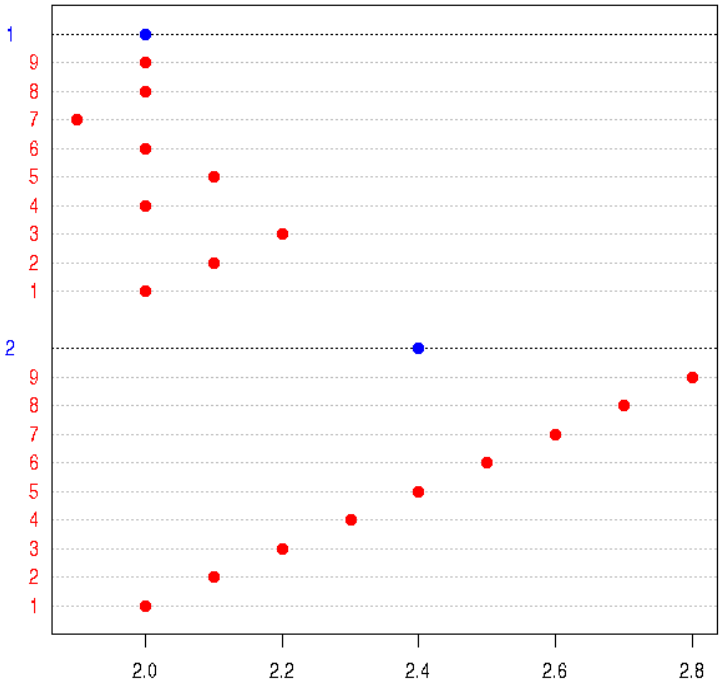
Se requiere la biblioteca tseries y la función runs.test()

library(“tseries”)

Aplicamos el test de rachas a las medidas de la columna o y b de la hoja tablaR333 y tomaremos la mediana como referencia

Veamos primero la gráfica en la que además de los datos mostramos el valor de la mediana. Es útil para este caso la gráfica del tipo dotchar, aprendida en el módulo 2 de este curso. En esta gráfica mostraremos los valores de ambas columnas

```
>
dotchart(cbind(tablaR333$medicion[tablaR333$condicion=="b"],tablaR333$medicion[tablaR333$condicion=="o"]),gdata=c(tapply(tablaR333$medicion,tablaR333$condicion,median)),pch=19,gpch=19,color="red",gcolor="blue")
```



La gráfica es evidente que en el panel superior que representa la condición b, no existen rachas de valores o si la hay es poco evidente. En cambio en el panel inferior, se observa una mediana de 2,4 y los primeros 4 valores fueron menores a ella y los cuatro últimos superiores a ella.

En el panel superior hay dos rachas consecutivas con valores superiores a la mediana y uno con valores menores a la mediana. Muchos valores coinciden con la mediana. En cambio en el panel inferior hay claramente una racha de valores menores y otra de valores mayores.

ejecutamos entonces la función runs.test() para el test de rachas de la condición "o"

```
>
runs.test(as.factor(tablaR333$medicion[tablaR333$condicion=="o"]>median(tablaR333$medicion[tablaR333$condicion=="o"])))
```

Runs Test

data: as.factor(tablaR333$medicion[tablaR333$condicion == "o"] > median(tablaR333$medicion[tablaR333$condicion == "o"]))

Standard Normal = -2.49, p-value = 0.01278

alternative hypothesis: two.sided

Concluimos que para la medición en condición "o" el p-value=0,01278. Es decir que bajo la hipótesis nula, que no hay rachas evidentes de datos, la probabilidad de hallar esta distribución de rachas es muy poco probable= 1,2% de probabilidad.

Por lo tanto concluimos que existen rachas de datos evidentes con p-value= 0,01278.

Simplificando: como p-value<0,01, decimos que los datos se autocorrelacionan y se distribuyen por rachas alrededor de la mediana. Indicando que no habría aleatoriedad en la toma de los datos.

Veamos ahora la condición "b"

>
runs.test(as.factor(tablaR333$medicion[tablaR333$condicion=="b"]>median(tablaR333$medicion[tablaR333$condicion=="b"])))

Runs Test

data: as.factor(tablaR333$medicion[tablaR333$condicion == "b"] > median(tablaR333$medicion[tablaR333$condicion == "b"]))
Standard Normal = 0.20597, p-value = 0.8368
alternative hypothesis: two.sided
Conclusión: como p-value>0,01 podemos concluir que no existe una distribución por rachas y por lo tanto hay una distribución aleatoria.

3.2. Pruebas de homogeneidad de variancias

La variancia es un parámetro de las distribuciones normales. Muchos test estadísticos tienen como supuesto que las muestras que se comparan tienen variancias homogéneas, es decir, suponen que las variancias, que mide la dispersión de los datos, no discrepa entre los grupos.

Es claro que la prueba de homogenidad de variancias las haremos si hemos demostrado que la muestra tiene distribución normal.

Si bien esto puede observarse a través de gráficos, no siempre es tan sencillo y los test analíticos nos permite aceptar o rtabechazar la hipótesis que las variancias de las muestras son homogéneas.

Introduzca en su espacio de trabajo los datos de la hoja tablaR331 de la planilla de cálculo tablaR3-3.ods/xls

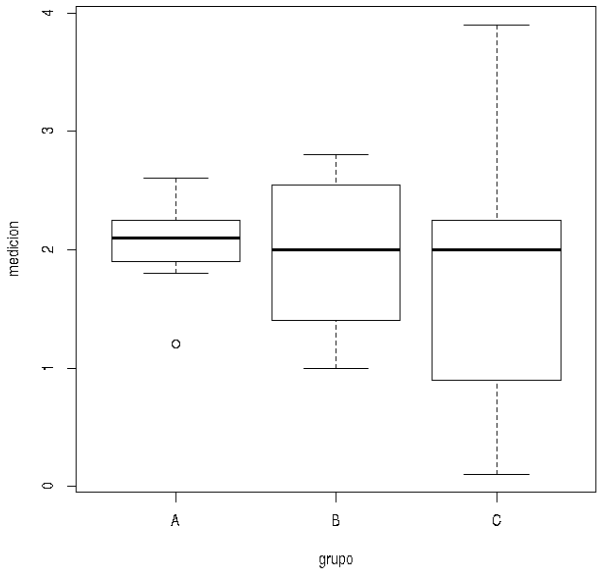
> tablaR331<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR331
  medicion grupo
1   2.1   A
2   2.2   A
3   2.3   A
4   2.1   A
5   1.9   A
6   1.2   A

.....
21 2.7 B
22 1.3 B
23 0.2 C
24 2.0 C
25 3.9 C
26 1.9 C
27 0.1 C
28 2.0 C
29 2.5 C
30 1.5 C
31 0.3 C
32 2.0 C
33 2.8 C

grafiquemos estos datos

```
> plot(medicion~grupo,data=tablaR331)
```



La gráfica nos muestra un distribución de los datos bastante diferentes entre los grupos, evidenciado por rangos y quartilos. Es evidente una dispersión de los datos que determinará variancias bastante diferentes para grupo.

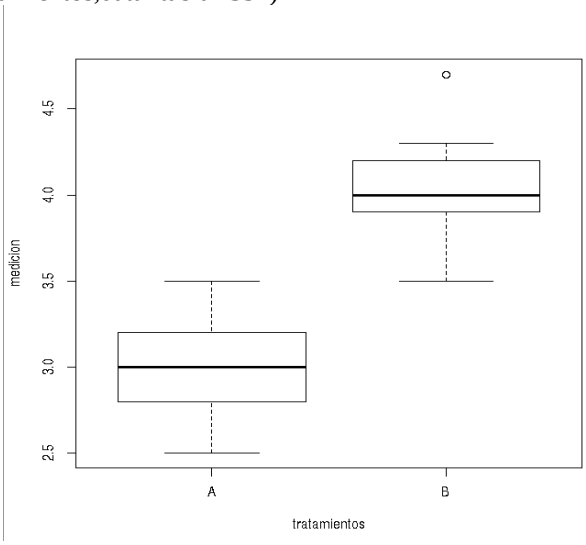
Veamos los valores con la función tapply()

```
> tapply(tablaR331$medicion,tablaR331$grupo,var)
        A         B         C
0.1427273 0.4445455 1.3787273
```

Vemos que la variancia del grupo C es 10 veces mayor que la del grupo A. Sin embargo, no podemos con estos valores ni con el gráfico asegurar que las variancias son homogéneas o no. Menos aun podemos afirmar sobre nuestra conclusión con una probabilidad de error.

Veamos otro ejemplo. Introduzcamos los datos de la tablaR332 de la misma planilla de cálculo.

```
>tablaR332<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR332
   medicion tratamientos
1       3.0           A
2       3.1           A
3       3.2           A
4       2.8           A
5       2.9           A
6       3.0           A
7       3.5           A
8       2.5           A
9       2.8           A
10      3.2           A
11      4.0           B
12      4.1           B
13      4.3           B
14      3.9           B
15      3.9           B
16      4.0           B
17      4.7           B
18      3.5           B
19      3.5           B
20      4.2           B
> plot(medicion~tratamientos,data=tablaR332)
```



la gráfica nos muestra una dispersión de los puntos bastante parecida, tanto en rango como en quartilos. Calculemos las variancias

```
> tapply(tablaR332$medicion,tablaR332$tratamientos,var)
         A         B
0.07555556 0.12766667
```

Vemos que las variancias son semejantes, en consonancia con la gráfica obtenida recientemente.

A continuación veremos test estadísticos que nos permiten realizar estas comprobaciones y aceptar o rechazar la hipótesis nula con una probabilidad de error de tipo I conocida.

Veremos las siguientes pruebas para evaluar la homogenidad de variancias

Test de Bartlett

Var test

Fligner test

3.2.1. Test de Bartlett

Esta prueba sirve para evaluar la homogenidad de variancias entre dos o más muestras.

El código general es

bartlett.test(list(x,y,....)

donde x, y, ... son grupos de datos

Veamos la aplicación a la tablaR331, que muestra valores de una variable medicion, para tres grupos de datos: A, B y C

```
>
bartlett.test(list(tablaR331$medicion[tablaR331$grupo=="A"],tablaR331$medicion[tablaR331$gr
upo=="B"],tablaR331$medicion[tablaR331$grupo=="C"]))
        Bartlett test of homogeneity of variances
data:  list(tablaR331$medicion[tablaR331$grupo == "A"], tablaR331$medicion[tablaR331$grupo
==     "B"], tablaR331$medicion[tablaR331$grupo == "C"])
Bartlett's K-squared = 11.188, df = 2, p-value = 0.00372
```

Como p-value<0,01 podemos concluir que las variancias no son homogéneas. Vemos que la conclusión coincide con la observación realizada de las gráficas.

Veamos ahora para la tablaR332, donde la variable medicion se registró para dos grupos de datos: A y B

```
>
bartlett.test(list(tablaR332$medicion[tablaR332$tratamientos=="A"],tablaR332$medicion[tablaR3
32$tratamientos=="B"]))
        Bartlett test of homogeneity of variances
data:  list(tablaR332$medicion[tablaR332$tratamientos ==  "A"],
tablaR332$medicion[tablaR332$tratamientos ==     "B"])
Bartlett's K-squared = 0.57992, df = 1, p-value = 0.4463
```

Como p-value>0,01 concluimos que las variancias de los datos de los tratamientos A y B son homogéneas.

3.2.2. Var test

Var test es una prueba de homogeneidad de variancias que se halla instalada en R, sin necesidad de paquetes extra ya que se halla en el package:stats, que se instala simultáneamente con el programa.

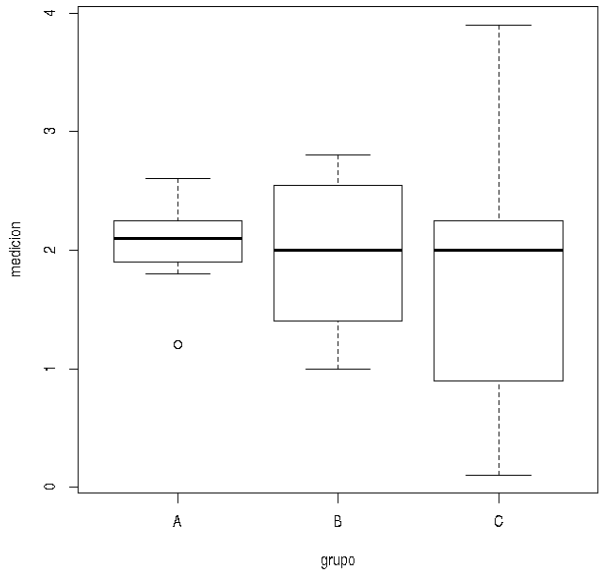
Es una prueba que sirve para comparar **dos variancias** de dos muestras con distribución normal

El código general es

var.test(x, y.)

donde x e y son los datos de cada grupo a evaluar.

Apliquemos el test a comparar las variancias de los grupos A y B de la tablaR331, cuyos datos dieron el siguiente gráfico para los grupos A, B y C




```
>
var.test(tablaR331$medicion[tablaR331$grupo=="A"],tablaR331$medicion[tablaR331$grupo=="
B"])
        F test to compare two variances
data:  tablaR331$medicion[tablaR331$grupo == "A"] and tablaR331$medicion[tablaR331$grupo
== "B"]
F = 0.32106, num df = 10, denom df = 10, p-value = 0.08741
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.08638186 1.19332581
sample estimates:
ratio of variances
      0.3210634
```

como p-value>0.01 concluimos que las variancias no son diferentes entre el grupo A y B

Probemos ahora el test entre los grupos A y C

```
>
var.test(tablaR331$medicion[tablaR331$grupo=="A"],tablaR331$medicion[tablaR331$grupo=="
C"])

        F test to compare two variances
```

data: tablaR331$medicion[tablaR331$grupo == "A"] and tablaR331$medicion[tablaR331$grupo == "C"]
F = 0.10352, num df = 10, denom df = 10, p-value = 0.001324
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.02785225 0.38476614
sample estimates:
ratio of variances
 0.103521

Como p-value es menor que 0,01 concluimos que las variancias de A y C son diferentes.

Obviamente var.test no servirá para comparar las tres muestras simultáneamente. En el caso de los datos de tablaR331, el test de Bartlett parece más adecuado, por ser tres muestras diferentes.

3.2.3. Fligner Test

La función fligner.test() permite comparar variancias de muestras y se halla por defecto entre los paquetes de R en el package:stats.

El test de Fligner-Killeen es una test de homogenidad de variancias donde la hipótesis nula es que las variancias de los grupos es la misma. Se puede aplicar a más de dos muestras de datos y no supone distribución normal de los datos. Utiliza como centro de distribución las medianas de cada muestra.

El código general es

fligner.test(list(x,y,...))

donde x, y, ... son vectores de datos de cada muestra o grupos de datos de un data.frame.

Veamos su aplicación con los grupos A, B y C de la tablaR331

```
>
fligner.test(list(tablaR331$medicion[tablaR331$grupo=="A"],tablaR331$medicion[tablaR331$grupo=="B"],tablaR331$medicion[tablaR331$grupo=="C"]))
        Fligner-Killeen test of homogeneity of variances
data:  list(tablaR331$medicion[tablaR331$grupo == "A"], tablaR331$medicion[tablaR331$grupo ==  "B"], tablaR331$medicion[tablaR331$grupo == "C"])
Fligner-Killeen:med chi-squared = 4.5715, df = 2, p-value = 0.1017
```

Según el test de Fligner, las variancias de los grupos no son diferentes ya que p-value>0,05.

Si el mismo test se hace con el test de Bartlett, obtenemos un resultado diferente.

```
>
bartlett.test(list(tablaR331$medicion[tablaR331$grupo=="A"],tablaR331$medicion[tablaR331$grupo=="B"],tablaR331$medicion[tablaR331$grupo=="C"]))
        Bartlett test of homogeneity of variances
data:  list(tablaR331$medicion[tablaR331$grupo == "A"], tablaR331$medicion[tablaR331$grupo ==  "B"], tablaR331$medicion[tablaR331$grupo == "C"])
Bartlett's K-squared = 11.188, df = 2, p-value = 0.00372
```

Como p-value<0,05 podemos concluir que las variancias no son homogéneas.

¿Cómo es esto? ¿Puedo tener dos conclusiones distintas sobre el mismo análisis?. La respuesta es si. Dependerá del test aplicado. Evidentemente el test de Fligner, es un test más conservador y la posibilidad de falsos positivos (es decir de cometer un error de tipo I = rechazar la hipótesis nula

cuando es realmente verdadera = decir que las variancias son distintas cuando en realidad no lo eran) es menor.

Acá pueden aparecer cosas confusas.

Si comparamos

test de Bartlett p-value = 0.00372

test de Fligner p-value = 0.1017

Por el valor del p-value del test de Bartlett me inclinaré por decir que las variancias son diferentes. Con los mismos datos, con el test de Fligner por el p-value me inclinaré por decir que no los son. Es decir que si fueran iguales, en el test de Bartlett tengo más chance de estar dando una conclusión errónea. Sin embargo por el otro lado se modifica el error de tipo II, es decir, rechazar la hipótesis alternativa cuando la hipótesis nula es cierta. En el test de Fligner se aumenta el error de tipo II y por lo tanto baja la potencia de nuestro ensayo. Volveremos sistemáticamente sobre este tema

4. Clase 3.4

Video: https://youtu.be/USh0Fi3PAzE
Tabla de datos: http://hdl.handle.net/2133/11560

4.1. Comparación de dos muestras

Es habitual en la investigación tener dos muestras de datos de una misma variable y deseamos conocer si las mismas discrepan o no. Por ejemplo, hemos medido la concentración de glucosa en un grupo de personas normales y en otro grupo de personas con una determinada característica, diferente de los normales y deseamos comprobar si los valores de la concentración de glucosa difieren o no entre los grupos.
Cuando tenemos dos muestras de datos cuantitativos a comparar tenemos varias preguntas que hacernos para encaminar nuestro análisis:
1- ¿Los datos tienen distribución normal?
Las respuesta puede ser si o no. Conclusión a la que llegaremos con el test de Shapiro Wilk o bien con la prueba de Kolmogorv Smirnov.
2- Si las muestras tuvieron distribución normal nos preguntamos ¿Las variancias son homogéneas?
Las repuestas posibles son si o no. Conclusión a la que llegaremos con los test de Bartlett, var.test o Fligner test.
3- Si las muestras tuvieron distribución normal y las variancias son homogéneas nos preguntamos si las muestras son independientes (no apareadas) o dependientes (apararеadas).
La respuesta será si o no y para ello no tenemos test sino que tendremos que analizar el diseño de nuestro experimento.
Si la respuesta es : independientes, aplicamos la prueba t de Student para datos independientes, que se logra con la función t.test(....,paired=FALSE)
Si la respuesta es: apareados, aplicamos la prueba t de Student para datos apareados, que se logra con la función t.test(...,paired=TRUE)
los puntos 1-3 quedan reflejados en la Figura 4.1.

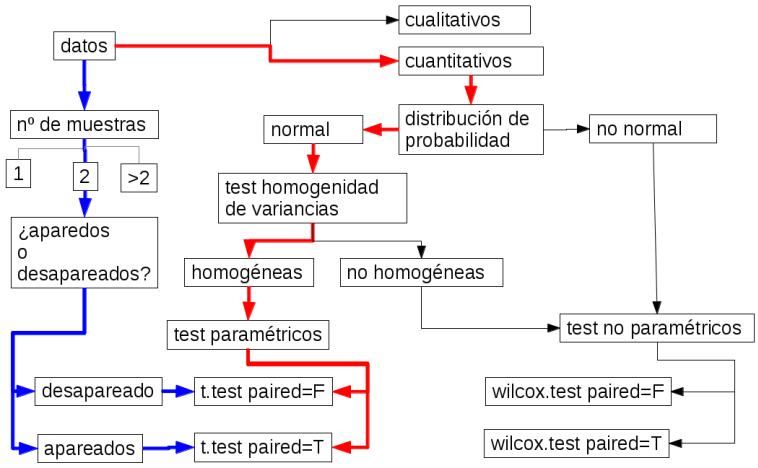


Figura 4.1. Decisiones a tomar con dos muestras de datos cuantitativos que cumplen los principios de normalidad e igualdad de variancias, pudiendo ser apareados/no apareados

4- Si las muestras no hubieran tenido distribución normal o no hubieran tenido las variancias homogéneas, aplicamos pruebas no paramétricas. Nos preguntaremos: ¿Los datos son independientes o apareados?

Si los datos son independientes aplicamos la prueba de Mann Whitney, que se ejecuta con la función wilcox.test(..... paired=FALSE).

Si los datos hubieran sido apareados, aplicamos la prueba de Wilcoxon, que se ejecuta con la función wilcox.test(.....paired=TRUE).

La Figura 4.2 muestra lo expuesto 1, 2 y 4.

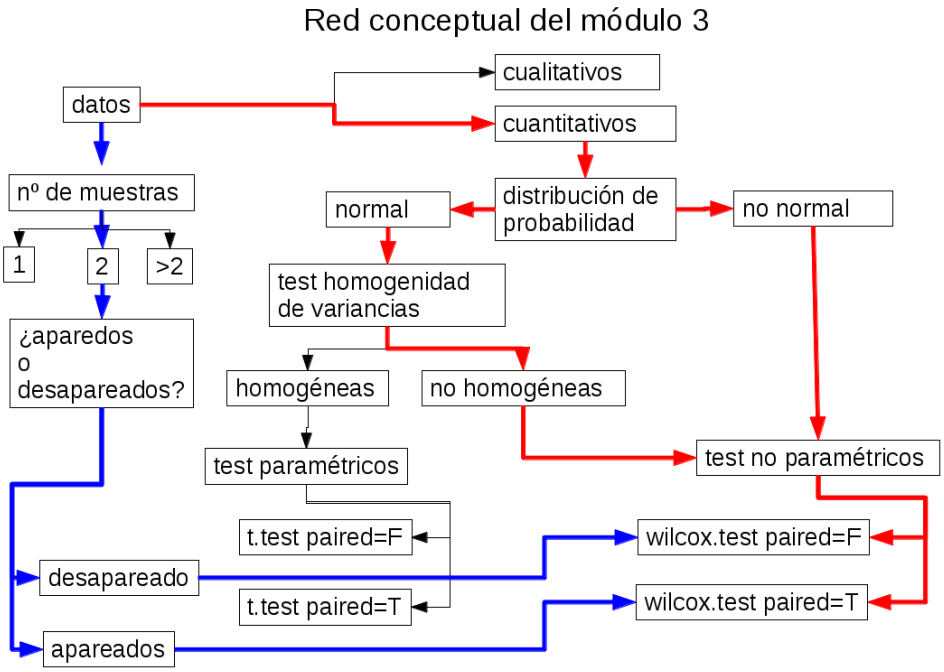


Figura 4.2. Decisiones a tomar con dos muestras de datos cuantitativos que no cumplen uno o más principios: normalidad, igualdad de variancias, apareados/no apareados

Pero podemos tener otras preguntas que hacernos. Supongamos que sospechamos fuertemente una diferencia entre los grupos, por datos exploratorios nuestros o de la bibliografía. Entonces deseamos fijar un número de datos para nuestras muestras que nos permitan verificar dicha diferencia.

5- La pregunta que podríamos hacernos sería ¿Cuál debería ser el número de datos de cada muestra para demostrar dicha diferencia con un error de tipo I del 0.05 y una potencia de 0.8. Recordemos que en ensayos biológicos un error de tipo I de 0.05 o menor suele ser aceptable. De cualquier forma es un tema siempre a discutir a la hora del diseño de los experimentos y el análisis de los datos. Por el otro lado la potencia del 0.8 es aceptable, es decir, si aceptamos que existe la diferencia, la probabilidad que esto sea así es del 80%. Por supuesto pueden fijarse otros valores.

El cálculo del número de datos necesarios para probar nuestra hipótesis se realiza con funciones del paquete pwr de R. Descárguelo de los repositorios que utiliza habitualmente.

Los test de este paquete tienen cuatro variables

a- número de datos (n)

b- magnitud del efecto (z): es un valor que nos indica el efecto que sobre la variable en estudio tiene el tratamiento que diferencia ambos grupos en estudio. Este valor puede obtenerse en base a datos de experimentos piloto, datos de otros investigadores, datos de la bibliografía o bien por aproximación con una función provista por el paquete. Lo recomendable es obtenerlo por experimentos piloto, con un número reducido de unidades experimentales.

c- Probabilidad de error tipo I (alfa)

d- Potencia (Pot)

fijados z, Pot y alfa, se obtiene el valor de n para cada grupo. La Figura 4.3 esquematiza lo expuesto

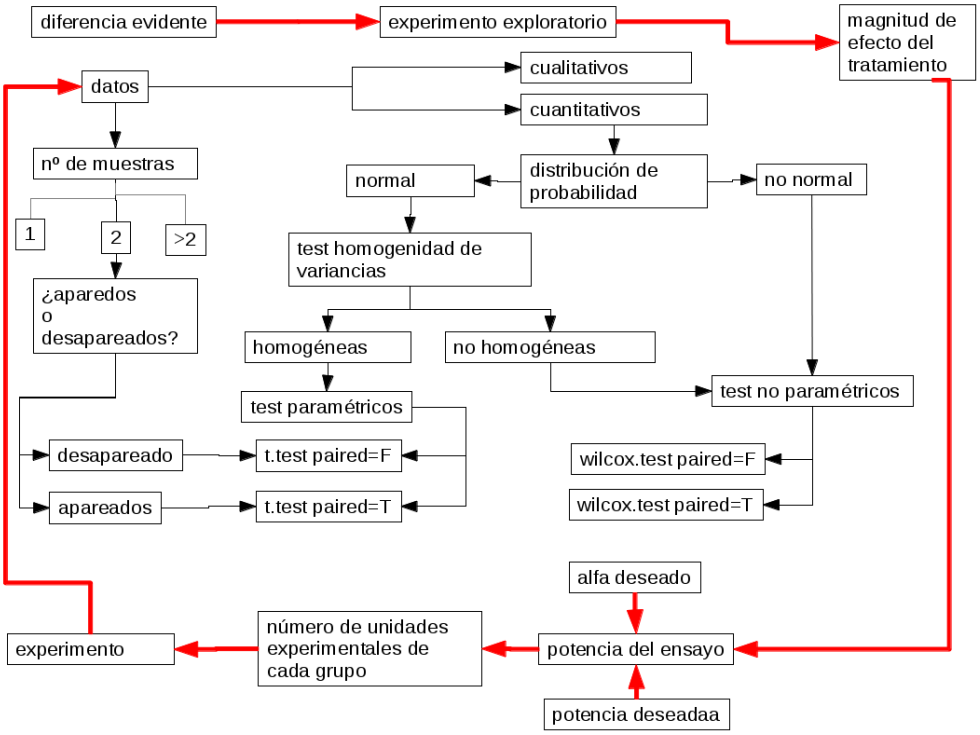


Figura 4.3.

6- Si ya hemos realizado el experimento, demostramos la diferencia con un error de tipo I igual o menor a 0.05 y con un cierto número de datos podríamos preguntarnos ¿Qué potencia tiene el ensayo?. Es decir que probabilidad tiene mi conclusión de ser cierta. Esto también se hace con el test de potencia del paquete pwr. La Figura 4.4 esquematiza lo expuesto

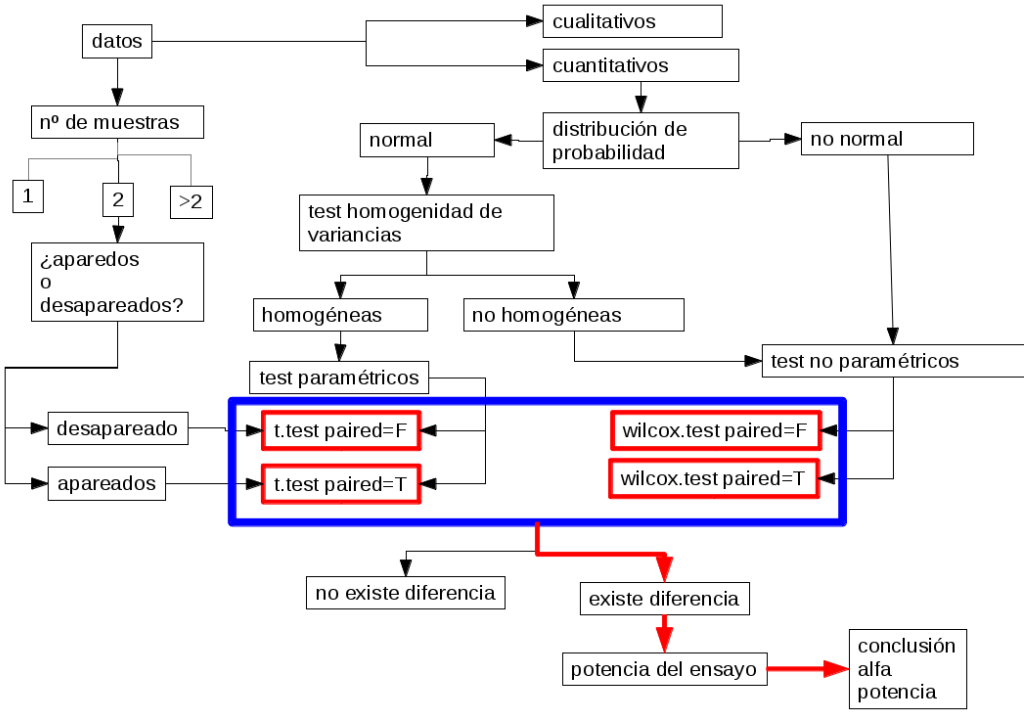


Figura 4.4

Veremos algunos casos ilustrativos.

4.1.1. Caso 1 (2 muestras datos independientes)

Hemos realizado un experimento y tenemos dos grupos de datos, representados por la tablaR341 de la planilla de cálculo tablaR3-4.ods/xls. No conocemos más que los datos obtenidos, el diseño del experimento y la probabilidad de error de tipo I que fijaremos en 0.05. Es decir que aceptaremos diferencias entre los grupos cuando p-value<0.05. Los datos corresponden a calcemias medidas en dos grupos de animales donde un grupo recibió un tratamiento (nivel factor: tratado) y otro es un grupo sin tratamiento (nivel factor: control).

Introduzcamos los datos en el espacio de trabajo

```
> tablaR341<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")
```

```
> tablaR341
  calcemia tratamiento
1     10.0     control
2     10.1     control
3     11.0     control
4     10.5     control
5     11.1     control
6     12.3     control
7      9.8     control
8     10.5     control
9      9.9     control
```

10 7.8 tratado
11 8.1 tratado
12 9.0 tratado
13 10.0 tratado
14 11.0 tratado
15 7.9 tratado
16 8.8 tratado
17 9.0 tratado
18 7.7 tratado

Seguiremos los mismos pasos expuestos en la introducción de esta clase

1- ¿Tienen los datos distribución normal?

Aplicamos test de Shapiro Wilk al grupo de animales controles, utilizando el siguiente código

```
> shapiro.test(tablaR341$calcemia[tablaR341$tratamiento=="control"])
        Shapiro-Wilk normality test
data:  tablaR341$calcemia[tablaR341$tratamiento == "control"]
W = 0.86982, p-value = 0.1224
```

y luego aplicamos el mismo test al grupo de animales tratados

```
> shapiro.test(tablaR341$calcemia[tablaR341$tratamiento=="tratado"])
        Shapiro-Wilk normality test
data:  tablaR341$calcemia[tablaR341$tratamiento == "tratado"]
W = 0.88756, p-value = 0.1883
```

En función de los p-values obtenidos de aplicar el test a ambas muestras, aceptamos que ambas muestras tienen distribución normal. Hubiéramos rechazado el supuesto de normalidad si p-value hubiera sido al menos para uno de los grupos menor a 0.05.

2- Probaremos si las muestras tienen variancias homogéneas. Siendo dos muestra podemos aplicar cualquier test. Aplicaremos test de Bartlett

```
>
bartlett.test(list(tablaR341$calcemia[tablaR341$tratamiento=="control"],tablaR341$calcemia[tablaR341$tratamiento=="tratado"]))
        Bartlett test of homogeneity of variances
data: list(tablaR341$calcemia[tablaR341$tratamiento == "control"], tablaR341$calcemia[tablaR341$tratamiento == "tratado"])
Bartlett's K-squared = 0.82697, df = 1, p-value = 0.3632
```

Como p-value>0.05, aceptamos que las variancias son homogéneas, es decir nos quedamos con la hipótesis nula.

3- Nos preguntamos si los datos son independientes o apareados. Por el diseño experimental se trata de datos independientes. Los animales del grupo control no son los mismos del grupo tratado.

Por lo tanto aplicaremos la función t.test() con el argumento paired=FALSE, indicándole a la función que no hay apareamiento o dependencia de datos. En la función t.test() colocamos primero los datos de calcemia del grupo control y luego las calcemias del grupo tratado. Identifique en la función escrita a continuación, cada uno de los datos mencionados.

```
>
```

t.test(tablaR341$calcemia[tablaR341$tratamiento=="control"],tablaR341$calcemia[tablaR341$tratamiento=="tratado"],paired=FALSE)

```
        Welch Two Sample t-test
data: tablaR341$calcemia[tablaR341$tratamiento == "control"] and tablaR341$calcemia[tablaR341$tratamiento == "tratado"]
t = 3.8804, df = 14.493, p-value = 0.001569
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.793283 2.740050
sample estimates:
mean of x mean of y
10.577778  8.811111
```

como p-value<0.05, concluimos que los valores de calcemia del grupo tratado son diferentes de la calcemia del grupo control. Algunos otros datos interesantes a tener en cuenta son el ·95 percent confidence interval", si hay diferencias entre los grupos este intervalo no debe contener el valor 0. Además el análisis nos da los valores de las medias del grupo control y tratado, donde vemos que además de diferentes, el grupo tratado tiene calcemia menor que el control.

4- no corresponde a este análisis

5- no corresponde a este análisis

6- Nos cabe preguntarnos ¿Cuál es la potencia de nuestro ensayo?

Recordemos que para la potencia utilizaremos el paquete pwr. El test de potencia como dijimos utiliza cuatro datos: alfa, n, potencia y magnitud de efecto (EZ).

Deseamos conocer la potencia, por lo tanto los otros valores los debemos tener: alfa=0.05, n= 9 (nueve datos por grupo), magnitud de efecto (se calcula con fórmula especial para cada ensayo)

Para comparaciones de dos muestras la fórmula es

$$EZ = \frac{\left|(media\ grupo\ 1 - media\ grupo\ 2)\right| * 2}{(desvío\ estandar\ grupo\ 1 + desvío\ estándar\ grupo\ 2)}$$

Ecuación 4.1.

Calculemos EZ para nuestros datos. Para ello necesitamos las medias y desvíos estándar, con los códigos siguientes hacemos dichos cálculos y los asignamos a objetos: mediacontrol, mediatratado, sdcontrol, sdtratato

```
> mediacontrol<-mean(tablaR341$calcemia[tablaR341$tratamiento=="control"])
> mediatratado<-mean(tablaR341$calcemia[tablaR341$tratamiento=="tratado"])
> sdcontrol<-sd(tablaR341$calcemia[tablaR341$tratamiento=="control"])
> sdtratado<-sd(tablaR341$calcemia[tablaR341$tratamiento=="tratado"])
> EZ<-abs(mediacontrol-mediatratado)*2/(sdcontrol+sdtratado)
> EZ
[1] 1.854155
```

ahora reemplazamos en la función pwr, dejando que la función calcule la potencia (power), que para ello, a power le asignamos NULL. Elegimos además algunos parámetros como type:

two.samples, porque son dos muestras y alternative: two.sided, ya que estamos trabajando a dos colas, sin cuestionarrnos si una muestra es mayor o no que la otra, sino solamente si son diferentes o no.

```
> pwr.t.test(n = 9, d = EZ, sig.level = 0.05, power = NULL, type="two.sample", alternative="two.sided")
    Two-sample t test power calculation
             n = 9
             d = 1.854155
             sig.level = 0.05
             power = 0.9579999
   alternative = two.sided
NOTE: n is number in *each* group
```

Conclusión

En base al p-value del t.test() concluimos que la calcemia de los animales tratados es diferente de la calcemia de los animales controles ($p<0.05$). Y en base a la función pwr.t.test() concluimos que la potencia del ensayo fue de 0.96.

En otras palabras, la probabilidad de error al haber concluido que las calcemias son diferentes, si en realidad no lo fueran es menor a 0.05 (tendríamos una probabilidad de error del 5% de estar diciendo que son diferentes, cuando en realidad no fueran. Muy baja lo que hace pensar que nuestra conclusión es correcta). Por otro lado, la probabilidad de estar acertando en nuestra conclusión de decir que las calcemias son diferentes, siendo esto lo correcto es de 96%, valor muy alto, lo que induce a pensar aun más que estamos en lo cierto con nuestra conclusión.

4.1.2. Caso 2 (2 muestras datos dependientes)

Hemos realizado un experimento y tenemos dos grupos de datos, representados por la tablaR342 de la planilla de cálculo tablaR3-4.ods/xls. No conocemos más que los datos obtenidos, el diseño del experimento y la probabilidad de error de tipo I que fijaremos en 0.05. Es decir que aceptaremos diferencias entre los grupos cuando p-value<0.05. Los datos corresponden a las calcemias medidas en un mismo grupo de animales a dos tiempos consecutivos. La primer medición se halla indicada como "basal" y la medición al final del experimento, está identificada como "final". Debe notar con respecto al Caso 1, que en esta situación tenemos un solo grupo de unidades experimentales y sobre cada una de ellas hacemos las dos mediciones. Es importante que las unidades experimentales estén ordenadas siguiendo en ambos tiempo el mismo orden, como lo indica la tercer columna de la tablaR432

Introduzcamos los datos en el espacio de trabajo

```
>  tablaR342<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR342
  calcemia tiempo unidadexperimental
1     10.0  basal                  1
2     10.1  basal                  2
3     11.0  basal                  3
4     10.5  basal                  4
5     11.1  basal                  5
```

```
6     12.3  basal            6
7      9.8  basal            7
8     10.5  basal            8
9      9.9  basal            9
10     7.8  final            1
11     8.1  final            2
12     9.0  final            3
13    10.0  final            4
14    11.0  final            5
15     7.9  final            6
16     8.8  final            7
17     9.0  final            8
18     7.7  final            9
```

Seguiremos los mismos pasos expuestos en la introducción

1- ¿Tienen los datos distribución normal?

Aplicamos test de Shapiro Wilk para los valores "basal"

```
> shapiro.test(tablaR342$calcemia[tablaR342$tiempo=="basal"])
        Shapiro-Wilk normality test
data:  tablaR342$calcemia[tablaR342$tiempo == "basal"]
W = 0.86982, p-value = 0.1224
```

y luego para los valores "final"

```
> shapiro.test(tablaR342$calcemia[tablaR342$tiempo=="final"])
        Shapiro-Wilk normality test
data:  tablaR342$calcemia[tablaR342$tiempo == "final"]
W = 0.88756, p-value = 0.1883
```

En función de los p-values obtenidos de aplicar el test a ambas muestras, aceptamos que ambas muestras tienen distribución normal. Podríamos decir que las muestras no tienen distribución normal, pero en el caso de la muestra "basal" tendríamos una probabilidad de estar cometiendo un error del 12% y en el caso de la muestra "final" del 18%. Podrán parecerles pequeñas las probabilidades de equivocarnos en la toma de la decisión, pero no lo es tanto. Piense esto. Suponga que tiene un plato con deliciosos hongos preparados de una manera exquisita. Cuando está a punto de comerlos, le dice el chef: "Que disfrute la comida de estos hongos que consideramos saludables y sabrosos. Solo le avisamos que la probabilidad que sean venenosos y no saludables es del 18%". ¿Los comería?.

2- Como hemos aceptado que ambos grupos de datos tienen distribución normal, probaremos si las muestras tienen variancias homogéneas. Siendo dos muestra podemos aplicar cualquier test. Aplicaremos test de Bartlett

```
> bartlett.test(list(tablaR342$calcemia[tablaR342$tiempo=="basal"],tablaR342$calcemia[tablaR342$tiempo=="final"]))
        Bartlett test of homogeneity of variances
data:  list(tablaR342$calcemia[tablaR342$tiempo == "basal"], tablaR342$calcemia[tablaR342$tiempo == "final"])
Bartlett's K-squared = 0.82697, df = 1, p-value = 0.3632
```

Como p-value>0.05, aceptamos que las variancias son homogéneas, es decir nos quedamos con la hipótesis nula.

3- Nos preguntamos si los datos son independientes o apareados. Por el diseño del experimento, son datos aparedos o dependientes. Los animales son los mismos y sobre ellos se realizaron dos mediciones en diferente tiempo. El valor del segundo tiempo estará siempre condicionado por el valor del primero.

Por lo tanto aplicaremos la función t.test() con el argumento paired=TRUE.

```
> t.test(tablaR342$calcemia[tablaR342$tiempo=="basal"],tablaR342$calcemia[tablaR342$tiempo=="final"],paired=TRUE)
	Paired t-test
data: tablaR342$calcemia[tablaR342$tiempo == "basal"] and tablaR342$calcemia[tablaR342$tiempo == "final"]
t = 4.2468, df = 8, p-value = 0.002811
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.8073702 2.7259632
sample estimates:
mean of the differences
	1.766667
```

como p-value<0.05, concluimos que la calcemia final de los animales es diferente de la calcemia basal.

4- no corresponde a este análisis

5- no corresponde a este análisis

6- Nos cabe preguntarnos ¿Cuál es la potencia de nuestro ensayo?

Recordemos que para la potencia utilizaremos el paquete pwr. El test de potencia como dijimos utiliza cuatro datos: alfa, n, potencia (power) y magnitud de efecto (EZ).

Como hemos concluido que las calcemias basales difieren de las finales, deseamos conocer la potencia, es decir la probabilidad de certeza de dicha conclusión. Por lo tanto los otros valores los debemos tener: alfa=0.05, n= 9 (nueve datos por grupo), magnitud de efecto (se calcula con fórmula especial para cada ensayo). Para comparaciones de dos muestras, la fórmula es la misma que utilizamos en caso de datos independientes

$$EZ=\frac{|(media\ grupo\ 1-media\ grupo\ 2)|*2}{(desvío\ estandar\ grupo\ 1+desvío\ estándar\ grupo\ 2)}$$

Ecuación 4.2.

Calculemos EZ para nuestros datos. Para ello necesitamos las medias y desvíos estándar

```
> mediabasal<-mean(tablaR342$calcemia[tablaR342$tiempo=="basal"])

> mediafinal<-mean(tablaR342$calcemia[tablaR342$tiempo=="final"])

> sdbasal<-sd(tablaR342$calcemia[tablaR342$tiempo=="basal"])

> sdfinal<-sd(tablaR342$calcemia[tablaR342$tiempo=="final"])


>  EZ<-abs(mediafinal-mediabasal)*2/(sdbasal+sdfinal)
```

> EZ

[1] 1.854155

y al aplicar la función pwr.t.test() debemos prestar atención que type debe tomar el valor "paired", por tratarse de datos apareados o dependientes.

> pwr.t.test(n = 9, d = EZ, sig.level = 0.05, power = NULL, type="paired", alternative="two.sided")
Paired t test power calculation
n = 9
d = 1.854155
sig.level = 0.05
power = 0.9978591
alternative = two.sided
NOTE: n is number of *pairs*
Conclusión

La calcemia final es diferente de la basal ($p<0.05$). La potencia del ensayo fue de 0,998. En otras palabras, la probabilidad de haber concluido que las calcemias son diferentes si en realidad no lo fueran es menor a 0.05 (tendríamos una probabilidad menor del 5% de estar diciendo que son diferentes, cuando en realidad no lo son. Muy baja lo que hace pensar que nuestra conclusión es correcta). Por otro lado, la probabilidad de decir que las calcemias son diferentes, siendo esto lo real es de 99,8%, valor muy alto, lo que induce a pensar aun más que estamos en lo cierto con nuestra conclusión.

4.1.3. Caso 3 (2 muestras datos independientes con variancias no homogéneas)

Hemos realizado un experimento y tenemos dos grupos de datos, representados por la tablaR343 de la planilla de cálculo tablaR3-4.ods/xls. No conocemos más que los datos obtenidos, el diseño del experimento y la probabilidad de error de tipo I que fijaremos en 0.05. Es decir que aceptaremos diferencias entre los grupos cuando p-value<0.05. Los datos corresponden a las calcemias medidas en dos grupos de animales simultáneamente: "control" y "tratado" . Es un diseño similar al caso 1

Introduzcamos los datos en el espacio de trabajo

> tablaR343<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR343
calcemia tratamiento
1 10.0 control
2 10.1 control
3 11.0 control
4 10.5 control
5 11.1 control
6 12.3 control
7 9.8 control
8 10.5 control
9 9.9 control
10 7.3 tratado
11 8.1 tratado
12 3.5 tratado
13 10.0 tratado
14 11.0 tratado

15 7.9 tratado
16 8.8 tratado
17 15.1 tratado
18 7.7 tratado

Seguiremos los mismos pasos expuestos en la introducción

1- ¿Tienen los datos distribución normal? primero probamos el supuestos para el grupo control

Aplicamos test de Shapiro Wilk

```
> shapiro.test(tablaR343$calcemia[tablaR343$tratamiento=="control"])

        Shapiro-Wilk normality test

data:  tablaR343$calcemia[tablaR343$tratamiento == "control"]
W = 0.86982, p-value = 0.1224
```

Aceptamos que el grupo control tiene distribución normal, ya que p>0,05. Probamos el mismo supuestos ahora para el grupo tratado.

```
> shapiro.test(tablaR343$calcemia[tablaR343$tratamiento=="tratado"])

        Shapiro-Wilk normality test

data:  tablaR343$calcemia[tablaR343$tratamiento == "tratado"]
W = 0.9318, p-value = 0.4987
```

En función de los p-values obtenidos de aplicar el test a ambas muestras, aceptamos que ambas muestras tienen distribución normal.

2- Probaremos si las muestras tienen variancias homogéneas. Siendo dos muestra podemos aplicar cualquier test. Aplicaremos test de Bartlett

```
>
bartlett.test(list(tablaR343$calcemia[tablaR343$tratamiento=="control"],tablaR343$calcemia[tablaR343$tratamiento=="tratado"]))

        Bartlett test of homogeneity of variances

data: list(tablaR343$calcemia[tablaR343$tratamiento == "control"], tablaR343$calcemia[tablaR343$tratamiento == "tratado"])
Bartlett's K-squared = 11.179, df = 1, p-value = 0.0008274
```

Como p-value<0.05, aceptamos que las variancias son diferentes, lo que nos obliga a resolver nuestro problema con un test no paramétrico. No podremos utilizar la función t.test() y nos inclinaremos por la funcion wilcox.test() que aplicamos cuando las muestras no tienen distribución normal y/o variancias homogéneas. .

3- Nos preguntamos si los datos son independientes o apareados. Por el diseño del experimento se trata de datos independientes

4- Como las muestras no tuvieron variancias homogéneas, a pesar de tener ambas muestras distribución normal, corresponde hacer una ensayo no paramétrico. Para este caso realizaremos la prueba de Mann Whitney

Por lo tanto aplicaremos la función wilcoxt.test() con el argumento paired=FALSE, ya que los datos por diseño son independientes o no se hallan apareados.

```
>
wilcox.test(tablaR343$calcemia[tablaR343$tratamiento=="control"],tablaR343$calcemia[tablaR343$tratamiento=="tratado"],paired=FALSE)
        Wilcoxon rank sum test with continuity correction
data: tablaR343$calcemia[tablaR343$tratamiento == "control"] and tablaR343$calcemia[tablaR343$tratamiento == "tratado"]
W = 63, p-value = 0.0517
```

como p-value>0.05, concluimos que la calcemia de los controles no difiere de los tratados.

5- Dado que el valor de p-value fue muy cercano al límite que tomamos para aceptar o rechazar la hipótesis nula. Realizaremos el cálculo del número de animales que deberíamos tener para demostrar que realmente existe diferencia con una potencia del 80%. Utilizamos para esto el mismo test del paquete pwr, salvo que los datos conocidos son EZ, power y alfa (sig.level)

calculamos EZ

```
> mediacontrol<-mean(tablaR343$calcemia[tablaR343$tratamiento=="control"])

>mediatratado<-mean(tablaR343$calcemia[tablaR343$tratamiento=="tratado"])

> sdcontrol<-sd(tablaR343$calcemia[tablaR343$tratamiento=="control"])

> sdtratado<-sd(tablaR343$calcemia[tablaR343$tratamiento=="tratado"])

>  EZ<-abs(mediacontrol-mediatratado)*2/(sdcontrol+sdtratado)

> EZ

[1] 0.8925653
```

ahora n será nuestra incógnita, a la que le asignamos el valor NULL

```
>  pwr.t.test(n = NULL, d = EZ, sig.level = 0.05, power = 0.8, type="two.sample", alternative="two.sided")
    Two-sample t test power calculation
             n = 20.70978
             d = 0.8925653
     sig.level = 0.05
         power = 0.8
   alternative = two.sided
NOTE: n is number in *each* group
```

6- No corresponde en este caso ya que no hallamos diferencia en el ensayo

Conclusión

Con los datos de nuestro experimento concluimos con un nivel de significación igual a 0.05 que la calcemia de los controles no difiere de la calcemia de los tratados. Sin embargo el p-value cercano al límite nos anima a evaluar el número de datos que requeriríamos para evidenciar la diferencia. Luego de aplicar la función pwr.t.test(), vemos que cada grupo debería tener 20 animales para demostrar la diferencia con una potencia del 80%. En este caso valdría la pena hacer el experimento.

Supongamos que hubiéramos obtenidos n=7200. En este caso el test de potencia no sirve más aun para ratificar que no existe diferencia entre los grupos.

Los ensayos realizados se hicieron de las maneras estándar, sin profundizar o hacer hincapié en parámetros de los ensayos que podrían aumentar la potencia de los mismos. En la clase siguiente se profundizará sobre ellos.

5. Clase 3. 5

Video: https://youtu.be/thxGzZIUNRM
Tabla de datos: http://hdl.handle.net/2133/11561

5.1. Comparación de una estadística de una muestras contra un valor

Cuando tenemos una muestra de datos cuantitativos y deseamos comparar alguna de sus estadísticas contra un valor dado (que puede provenir de la bibliografía o de algún experimento previo) tenemos varias preguntas que hacernos para encaminar nuestro análisis:

1- ¿Los datos tienen distribución normal?

La respuesta a la pregunta obviamente podrá ser SI o NO, dependiendo del resultado del test de normalidad, por ejemplo con la función shapiro.test()

Si la respuesta es si:

aplicaremos la función t.test(..... mu=numero)

El esquema siguiente muestra de manera esquemática los pasos, que luego realizaremos prácticamente.

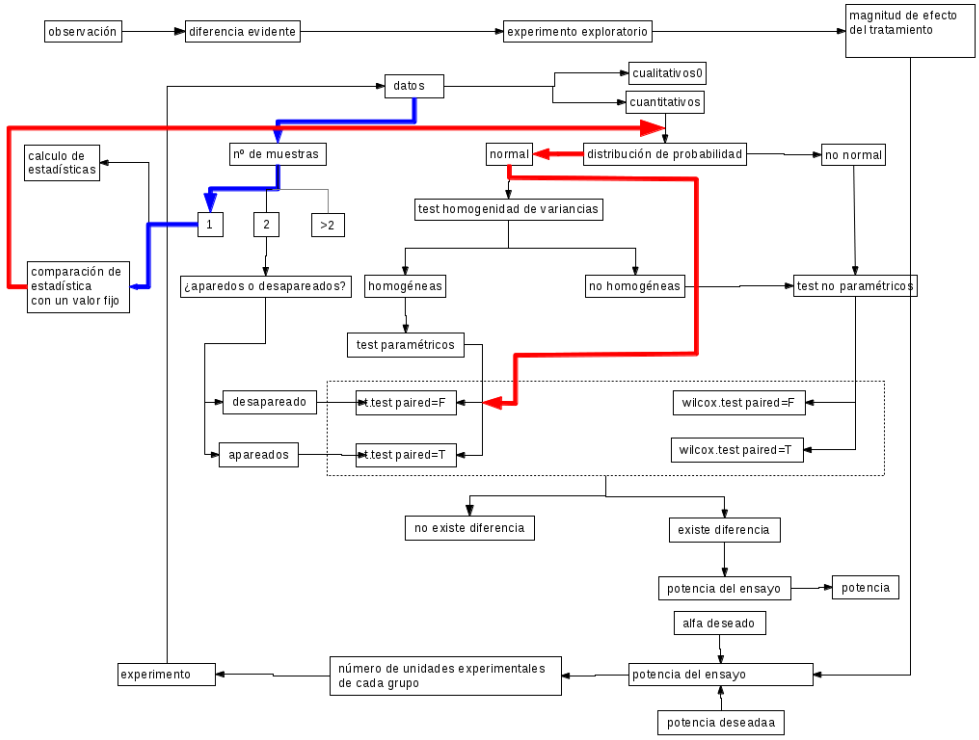


Figura 5.1
Si la respuesta fue no:

aplicaremos la función wilcox.test(.....mu=número)

La figura siguiente muestra los pasos a seguir

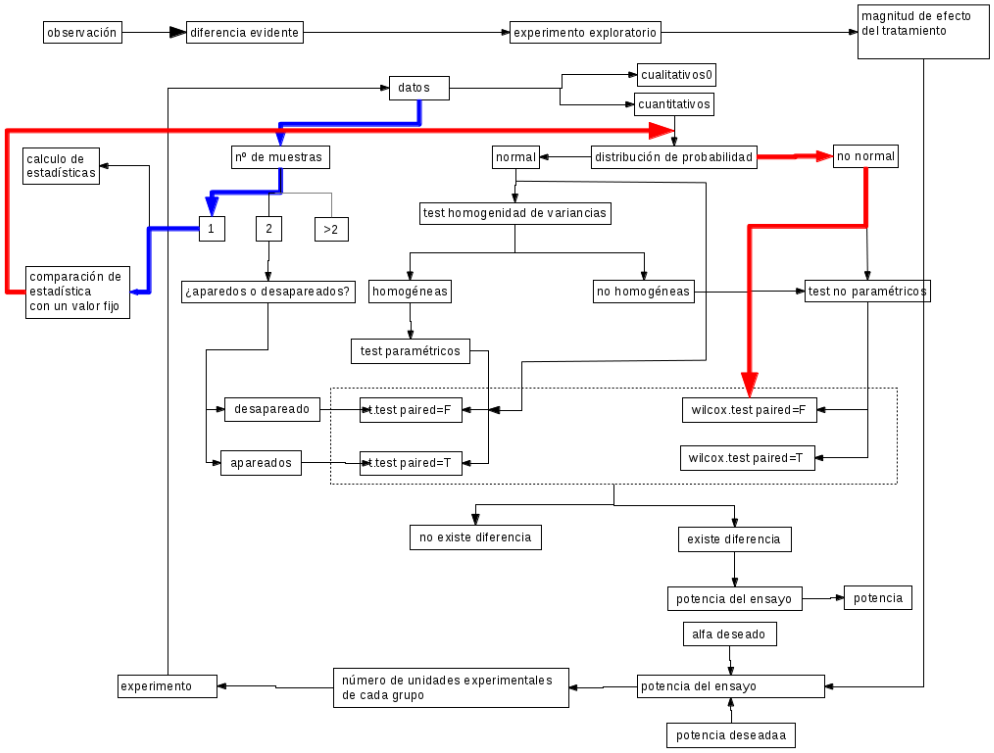


Figura 5.2

A

nalicemos una serie de casos para evaluar este ensayo.

5.1.1. Caso 1

Hemos realizado una serie de mediciones de fosfato en agua de diversos lugares, datos que están reflejados en los valores de la tablaR351 de la planilla de cálculo tablaR3-5.ods/xls.

Introduzcamos los datos en el espacio de trabajo

```
> tablaR351<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR351
   fosfato
1      1.0
2      0.9
3      1.2
4      0.8
5      0.9
6      1.0
7      1.1
8      1.0
9      1.2
10     0.8
11     0.6
```

12 1.0
13 0.9
14 0.6
15 0.6
16 1.0
17 0.7
18 0.9
19 1.0
20 1.1

Deseamos conocer si los datos son coincidentes con una media de 1.2, citada en la bibliografía.

En primer lugar probamos el supuesto de normalidad

```
> shapiro.test(tablaR351$fosfato)
        Shapiro-Wilk normality test
data:  tablaR351$fosfato
W = 0.92227, p-value = 0.1096
```

No podemos rechazar el supuesto de normalidad. Suponemos entonces distribución normal.

La pregunta que nos hacemos ahora es: ¿La media de este grupo difiere de la media = 1.2, citada en la bibliografía?

Para ello aplicamos la función t.test ()

En nuestro caso queremos comparar los valores de nuestra muestra con el valor 1.2.

Si observamos la media de nuestra muestra

```
> mean(tablaR351$fosfato)
```

[1] 0.915

Este valor parece ser bastante menor que el valor de comparación = 1.2

Es más, el valor mencionado coincide con el límite superior del rango

```
> range(tablaR351$fosfato)
```

[1] 0.6 1.2

Aunque a simple vista puede parecernos que la media de nuestra muestra será diferente del valor 1.2 de la bibliografía, la decisión final la tomaremos con un ensayo estadístico, utilizando la función t.test(). Sin embargo partiremos de la hipótesis nula que la media de nuestra muestra no difiere de 1.2 y como alternativa utilizaremos la hipótesis que nuestra media es menor a la de la bibliografía. Para tener en cuenta esto, modificamos el argumento alternative. En esta función el argumento alternative que puede tomar valores c("two.sided","less","greater"), podríamos modificarlo. Si no tuviéramos idea si realmente podría ser mayor o menor utilizamos two.sided, si nuestra media parece ser menor que el valor de referencia utilizaremos "less", que es nuestro caso. Si nuestra media pareciera mayor que el valor de referencia utilizaríamos "greater". La correcta elección de este argumento aumenta la potencia de nuestro ensayo. Sería absurdo para este caso proponer que nuestra media podría ser mayor que el valor de referencia, siendo este el límite superior de nuestro rango.

Entonces aplicamos la función t.test(). Como esta función se utiliza para comparar dos muestras, asignamaos a y=NULL (es decir no hay segunda muestra). El argumento alternative lo fijamos como se explicó anteriormente en "less" y mu es el valor de referencia, 1.2 en este caso. Como se verá en la función podemos explicitar también el error de tipo I. En casos anteriores no se

explicitó ya que el valor por defecto es 0.05. En la función t.test, como en otras funciones de R, se suele fijar el nivel de confianza (conf.level) que toma el valor 1-error de tipo I. Por esta razón si el error de tipo I lo fijamos en 0.05, conf.lev será igual a 0.95

```
> t.test(tablaR351$fosfato, y = NULL,alternative = c("less"), mu = 1.2, conf.level = 0.95)
        One Sample t-test
data:  tablaR351$fosfato
t = -6.9149, df = 19, p-value = 6.798e-07
alternative hypothesis: true mean is less than 1.2
95 percent confidence interval:
     -Inf 0.9862664
sample estimates:
mean of x
   0.915
```

Concluimos que nuestra muestra tiene una media menor a 1.2 con un p-value de 0.05.

Veamos ahora la potencia de nuestro ensayo, es decir la probabilidad de estar acertados en nuestra conclusión

para ello utilizaremos la función pwr.t.test() del paquete pwr. Cargamos la biblioteca mencionada

```
library(pwr)
```

Tomaremos como n el número de datos de nuestra muestra, que calcularemos con la función nrow(). Como magnitud del efecto (effect size) la diferencia entre nuestra media y el valor propuesto, dividido por el desvío estándar de la muestra.

$$EZ = \frac{\left|(media\ grupo - valor\ de\ referencia)\right|}{desvío\ estandar\ muestra}$$

Ecuación 5.1.

```
> ez=abs((mean(tablaR351$fosfato)-1.2)/sd(tablaR351$fosfato))
> ez
[1] 1.546228
```

Además fijamos sig.level=0.05, type="one.sample", alternative="two.sided")

```
> pwr.t.test(n = nrow(tablaR351), d = ez, sig.level = 0.05, power = NULL, type = c("one.sample"),alternative="two.sided")
    One-sample t test power calculation
             n = 20
             d = 1.546228
     sig.level = 0.05
         power = 0.9999976
   alternative = two.sided
```

Concluimos entonces que la media de nuestra muestra es significativamente menor que 1.2 (p-value<0.05) con una potencia del ensayo de 0,9999976, que es prácticamente 1. Es decir es muy poco probable que la media de nuestra muestra sea igual a la media de la bibliografía y por otro

lado es muy probable que sea realmente diferente.

5.1.2. Caso 2

Hemos realizado una serie de mediciones de fosfato en agua de diversos lugares, datos que están reflejados en los valores de la tablaR352 de la planilla de cálculo tablaR3-5.ods/xls.

Introduzcamos los datos en el espacio de trabajo

```
> tablaR352<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR352
   fosfato
1     0.50
2     0.15
3     0.25
4     0.75
5     1.00
6     1.20
7     0.25
8     0.29
9     0.18
10    0.45
11    0.62
12    0.52
13    0.87
14    0.13
15    0.15
16    1.90
17    0.85
18    0.28
> mean(tablaR352$fosfato)

[1] 0.5744444
```

Deseamos conocer si los datos son coincidentes en una media de 1.2, citada en la bibliografía.

En primer lugar probamos el supuesto de normalidad

```
> shapiro.test(tablaR352$fosfato)

        Shapiro-Wilk normality test

data:  tablaR352$fosfato

W = 0.84965, p-value = 0.008361
```

Rechazamos la hipótesis nula, que la muestra tiene distribución normal. Por lo tanto nos inclinaremos por un test no paramétrico.

La pregunta que nos hacemos es: ¿la media de este grupo difiere de la media = 1.2, citada en la bibliografía?.

Si analizamos el rango de nuestras mediciones

```
> range(tablaR352$fosfato)

[1] 0.13 1.90
```

vemos que el valor de comparación (1.2) está incluido en el rango. Por lo tanto es más razonable preguntarnos si este valor difiere o no de la media de nuestra media y no si es mayor o menor.

Aplicaremos entonces la función wilcox.test (.....)[1], por tratarse de una muestra sin distribución normal.

En nuestro caso queremos comparar el valor de la mediana de nuestra muestra con el valor 1.2.

Si observamos la mediana de nuestra muestra

> median(tablaR352$fosfato)

[1] 0.475

El argumento alternative se fijará en "two.sided" por razones expuestas más arriba.

Entonces aplicamos la función wilcox.test(). Como esta función la utilizaremos para una sola muestra el fijaremos el argumento y=NULL (es decir no hay segunda muestra).

El error de tipo I lo fijaremos en 0.05, por lo que el argumento conf.level toma el valor 0.95. Nótese que expresa el nivel de confianza que es 1-error de tipo I.

> wilcox.test(tablaR352$fosfato,y=NULL, alternative = c("two.sided"), mu = 1.2, paired = FALSE, conf.int = FALSE, conf.level = 0.95)

Wilcoxon signed rank test with continuity correction

data: tablaR352$fosfato

V = 7.5, p-value = 0.001179

alternative hypothesis: true location is not equal to 1.2

Concluimos que nuestra muestra tiene una mediana diferente de 1.2 con un p-value de 0.05

Veamos ahora la potencia de nuestro ensayo

Se presenta un problema a la hora del test de potencia si los datos no tienen distribución normal. Una alternativa sería aplicar pwr.t.test(), asumiendo la situación. Calculamos ez

> ez=abs(median(tablaR352$fosfato-1.2))/sd(tablaR352$fosfato)

> ez

[1] 1.566526

ahora aplicamos la función pwr.t.test(), asignando a **n** el número de filas de la tablaR352, al argumento **d** le asignamos el valor hallado para ez, sig.level será igual a 0.95, type= one sample y alternative= two.sided.

> pwr.t.test(n = nrow(tablaR352), d = ez, sig.level = 0.05, power = NULL, type = c("one.sample"),alternative="two.sided")

---

1 La función wilcox.test(....) ya fue utilizada con algunos argumentos. Los argumentos de la función wilcox.test son:

wilcox.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), mu = 0, paired = FALSE, conf.int = FALSE, conf.level = 0.95)

al comparar dos muestras, se especificará x e y. Si es una sola muestra contra un valor fijo, y=NULL. El argumento alternative tiene la interpretación dada en ensayos previos. Si no podemos decidir con seguridad si nuestro valor a comparar es mayor o menor que el de referencia, utilizamos two.sided. El argumento mu es el valor de referencia con el cual queremos comparar la mediana de nuestra muestra.

```
     One-sample t test power calculation
              n = 18
              d = 1.566526
      sig.level = 0.05
          power = 0.9999906
    alternative = two.sided
```

Concluiríamos que la mediana de nuestra muestra es significativamente diferente de 1.2 ($p<0.05$) con una potencia de aproximadamente 1.

5.2. Comparación de más de dos muestras

La comparación de más de dos grupos es habitual en los trabajos experimentales. En estos estudios en primer lugar se investiga si hay diferencias entre los grupos y esto es lo que desarrollaremos en esta clase. Luego de demostrar que hay diferencias entre los grupos, se busca entre qué grupos existe la diferencia. En este caso se aplican los test conocidos como "de comparaciones múltiples".

5.2.1. Caso 1: Datos independientes con distribución normal y variancias homogéneas

Supongamos que hemos medido el peso de animales de experimentación con tres tratamiento distintos: tablaR353 de la planilla de cálculo tablaR3-5.ods/xls.

Debemos dar primero dos respuestas

1- ¿Son datos dependientes o independientes?

2- ¿Tienen los datos distribución normal o no?

Introducimos los datos

```
> tablaR353<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR353
   numero tratamiento peso
1       1          a  127
2       2          a  130
3       3          a  125
4       4          a  126
5       5          a  121
6       6          a  115
7       7          a  120
8       8          a  121
9       9          a  126
10     10          a  114
11     11          b  154
12     12          b  154
13     13          b  154
14     14          b  155
15     15          b  146
16     16          b  151
17     17          b  156
18     18          b  154
19     19          b  145
20     20          b  151
21     21          c  126
```

```
22    22        c  127
23    23        c   125
24    24        c   127
25    25        c   132
26    26        c   124
27    27        c  124
28    28        c  129
29    29        c  128
30    30        c   127
```

Se trata de datos son independientes, porque son animales diferentes en los que se realizaron las mediciones. Pasamos ahora a realizar el test de normalidad para cada grupo. En primer lugar para el tratamiento **a**

```
> shapiro.test(tablaR353$peso[tablaR353$tratamiento=="a"])
        Shapiro-Wilk normality test
data:  tablaR353$peso[tablaR353$tratamiento == "a"]
W = 0.93574, p-value = 0.5066
```

luego el tratamiento **b**

```
> shapiro.test(tablaR353$peso[tablaR353$tratamiento=="b"])
        Shapiro-Wilk normality test
data:  tablaR353$peso[tablaR353$tratamiento == "b"]
W = 0.8689, p-value = 0.09705
```

y finalmente el tratamiento **c**

```
> shapiro.test(tablaR353$peso[tablaR353$tratamiento=="c"])
        Shapiro-Wilk normality test
data:  tablaR353$peso[tablaR353$tratamiento == "c"]
W = 0.92638, p-value = 0.4133
```

Los test aplicados a cada tratamiento nos permiten afirmar que las distribuciones de probabilidad son normales, ya que en ninguno de los tres casos el p-value fue menor a 0.05.

Ahora probamos homogenidad de variancias con el test de Bartlett, que permite comparar más de dos muestras.

```
>
bartlett.test(list(tablaR353$peso[tablaR353$tratamiento=="a"],tablaR353$peso[tablaR353$tratamiento=="b"],tablaR353$peso[tablaR353$tratamiento=="c"]))
        Bartlett test of homogeneity of variances
data:  list(tablaR353$peso[tablaR353$tratamiento == "a"], tablaR353$peso[tablaR353$tratamiento ==  "b"], tablaR353$peso[tablaR353$tratamiento == "c"])
Bartlett's K-squared = 4.7468, df = 2, p-value = 0.09317
```

El valor de p-value mayor a 0.05 nos permite afirmar que las variancias no son diferentes, es decir que son homogéneas.

Con distribuciones normales y variancias homogéneas la comparación de los grupos se realizará con la función aov(), investigando el efecto que sobre el peso tiene el tratamiento. Esta función permite hacer el análisis que normalmente se conoce como ANOVA a un criterio o análisis de la variancia a un criterio. Crearemos un objeto para recibir el análisis realizado por la función aov(), al que llamamos aovtablaR353

```
> aovtablaR353<-aov(peso~tratamiento,data=tablaR353)
```

veremos el contenido del objeto

```
> summary(aovtablaR353)
             Df Sum Sq Mean Sq F value   Pr(>F)
tratamiento  2   4993  2496.4   159.3 1.12e-15 ***
Residuals   27    423    15.7
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Un valor de p-value(en este caso expresado como Pr(>F) muy inferior a 0.05 nos indica que podemos inclinarnos por la hipótesis alternativa, es decir que los tratamientos tuvieron un efecto sobre el peso, haciendo que entre los tratamiento haya diferencia de peso. En la próxima clase veremos como investigar entre que grupos existen diferencias.

5.2.2. Caso 2: Datos dependientes con distribución normal y variancias homogéneas

Tomemos los datos de la tablaR354. Estos datos corresponden a pesos de 10 ratas medidos a los 30, 35 y 40 días. Como podemos ver en la columna número se repiten tres veces los valores de 1 a 10, indicando que cada rata fue medida tres veces. Indudablemente se trata de datos dependientes, ya que son mediciones realizadas a diferente tiempo sobre una misma unidad experimental.

Introducimos los datos, hacemos test de normalidad y de homogeneidad de variancias

```
 > tablaR354<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")
```

```
> tablaR354
   numero edad peso
1       1   30       127
2       2   30       130
3       3   30       125
4       4   30       126
5       5   30       121
6       6   30       115
7       7   30       120
8       8   30       121
9       9   30       126
10     10   30       114
11      1   35       134
12      2   35       137
13      3   35       132
14      4   35       133
15      5   35       128
16      6   35       122
17      7   35       127
18      8   35       128
19      9   35       133
20     10   35       121
21      1   40       152
22      2   40       154
23      3   40       154
24      4   40       155
25      5   40       146
26      6   40       151
27      7   40       156
```

```
28     8  40        154
29     9  40        145
30    10  40         151
```

Hacemos los test de normalidad para los valores de peso clasificados por cada edad

```
> shapiro.test(tablaR354$peso[tablaR354$edad==30])
        Shapiro-Wilk normality test
data:  tablaR354$peso[tablaR354$edad == 30]
W = 0.93574, p-value = 0.5066
```

```
> shapiro.test(tablaR354$peso[tablaR354$edad==35])
        Shapiro-Wilk normality test
data:  tablaR354$peso[tablaR354$edad == 35]
W = 0.93574, p-value = 0.5066
```

```
> shapiro.test(tablaR354$peso[tablaR354$edad==40])
        Shapiro-Wilk normality test
data:  tablaR354$peso[tablaR354$edad == 40]
W = 0.8689, p-value = 0.09705
```

Concluimos que los valores de peso tienen distribución normal. Probamos ahora si las variancias de los pesos para cada edad difieren o no con el test de Bartlett

```
>
bartlett.test(list(tablaR354$peso[tablaR354$edad==30],tablaR354$peso[tablaR354$edad==35],tab
laR354$peso[tablaR354$edad==40]))
        Bartlett test of homogeneity of variances
data:  list(tablaR354$peso[tablaR354$edad == 30], tablaR354$peso[tablaR354$edad ==      35],
tablaR354$peso[tablaR354$edad == 40])
Bartlett's K-squared = 1.244, df = 2, p-value = 0.5369
```

Los valores resaltados en amarillos nos permiten afirmar que las muestras tienen distribución normal y que las variancias de los grupos son homogéneas. Sin embargo son datos dependientes.

Se utiliza la función aov() con modificaciones. Cuando los datos están apareados, en este caso el peso está apareado por número. Es decir por ejemplo, a la rata 1 se le midió el peso a tres tiempos. Se utiliza el siguiente código, que es similar al utilizado en el ejemplo anterior, pero incluye un término Error

```
> aovtablaR354<-aov(peso~edad+Error(numero/peso),data=tablaR354)
```

se suma un término Error(variable que establece la dependencia/variable medida)

```
> summary(aovtablaR354)
Error: numero
          Df Sum Sq  Mean Sq  F value Pr(>F)
Residuals  1  1 97.3    197.3
Error: peso:numero
          Df   Sum Sq  Mean Sq
edad       1   4206    4206
Error: Within
            Df Sum Sq Mean Sq  F value     Pr(>F)
edad         1  535.0   535.0     38.5  1.45e-06 ***
```

```
Residuals      26  361.3       13.9
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

El valor de p-value (Pr)>F) nos permite afirmar que las medias de los pesos de los animales a diferentes tiempos son diferentes.

5.2.3. Caso 3: datos independientes sin distribución normal y/o sin homocedasticidad

Introducimos los datos de la tablaR355 y realizamos test de normalidad

```
> tablaR355<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")
> tablaR355
   numero tratamiento peso
1       1          a  127
2       2          a  130
3       3          a  125
4       4          a  126
5       5          a  121
6       6          a  115
7       7          a  120
8       8          a  121
9       9          a  126
10     10          a  114
11     11          b  156
12     12          b  154
13     13          b  154
14     14          b  155
15     15          b  146
16     16          b  151
17     17          b  156
18     18          b  154
19     19          b  145
20     20          b  151
21     21          c  126
22     22          c  127
23     23          c  125
24     24          c  127
25     25          c  132
26     26          c  124
27     27          c  124
28     28          c  129
29     29          c  128
30     30          c  127

> shapiro.test(tablaR355$peso[tablaR355$tratamiento=="a"])
        Shapiro-Wilk normality test
data:  tablaR355$peso[tablaR355$tratamiento == "a"]
W = 0.93574, p-value = 0.5066

> shapiro.test(tablaR355$peso[tablaR355$tratamiento=="b"])
        Shapiro-Wilk normality test
```

data: tablaR355$peso[tablaR355$tratamiento == "b"]
W = 0.83961, p-value = 0.04365

> shapiro.test(tablaR355$peso[tablaR355$tratamiento=="c"])
Shapiro-Wilk normality test
data: tablaR355$peso[tablaR355$tratamiento == "c"]
W = 0.92638, p-value = 0.4133
Como podemos ver el tratamiento **b** no tiene distribución normal con p<0.05

Entonces debemos utilizar un test no paramétrico. Utilizaremos el test de Kruskal Wallis, que sería el equivalente a ANOVA, pero se utiliza cuando no se cumple el supuesto de normalidad y/o homogeneidad de variancias. La función que se utiliza es kruskal.test()

> kruskal.test(peso~tratamiento,data= tablaR355)
Kruskal-Wallis rank sum test
data: peso by tratamiento
Kruskal-Wallis chi-squared = 21.208, df = 2, p-value = 2.482e-05
Conclusión: el peso de las ratas difiere entre los tratamiento aplicados, con un valor p<0.05. En la próxima clase veremos dicha diferencia entre que tratamientos se establece

5.2.4. Caso 4: datos dependientes sin distribución normal y/o sin homocedasticidad.

Los datos de la tablaR356 representan el peso de 10 ratas medido a tres tiempos consecutivos. Evidentemente se trata de datos apareados o dependientes.

Introducimos los datos y hacemos la prueba de normalidad

> tablaR356<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")

> tablaR356
numero edad peso
1 1 30 127
2 2 30 130
3 3 30 125
4 4 30 126
5 5 30 121
6 6 30 115
7 7 30 120
8 8 30 121
9 9 30 126
10 10 30 114
11 1 35 134
12 2 35 137
13 3 35 132
14 4 35 133
15 5 35 128
16 6 35 122
17 7 35 127
18 8 35 128
19 9 35 133
20 10 35 121
21 1 40 155
22 2 40 154
23 3 40 154

```
24     4   40  155
25     5   40  143
26     6   40  155
27     7   40  156
28     8   40  154
29     9   40  145
30    10   40  151
```

realizamos los test de normalidad para cada grupo

```
> shapiro.test(tablaR356$peso[tablaR356$edad==30])
        Shapiro-Wilk normality test
data:  tablaR356$peso[tablaR356$edad == 30]
W = 0.93574, p-value = 0.5066

> shapiro.test(tablaR356$peso[tablaR356$edad==35])
        Shapiro-Wilk normality test
data:  tablaR356$peso[tablaR356$edad == 35]
W = 0.93574, p-value = 0.5066

> shapiro.test(tablaR356$peso[tablaR356$edad==40])
        Shapiro-Wilk normality test
data:  tablaR356$peso[tablaR356$edad == 40]
W = 0.74111, p-value = 0.002782
```

Vemos que el grupo de mediciones a los 40 días no tiene distribución normal. Entonces aplicamos un test no paramétrico para datos dependientes: el test de Friedman o el test de Quade, son adecuados en esta circunstancia. El test de Friedman o el de Quade son equivalentes a ANOVA y Kruskal Wallis, salvo que es para casos en que no tiene distribución normal y/o homogeneidad de variancias y los datos son dependientes. Se utiliza la función friedman.test() o quade.test(), según sea nuestra elección

Con la función friedman.test() procedemos con el siguiente código

groups: lleva el nombre de la variable que se mide a más de un nivel en una misma unidad experimental.

blocks: lleva el nombre de la variable sobre la cual se realizaron las medidas repetidas.

```
> friedman.test(tablaR356$peso,groups=tablaR356$edad,blocks=tablaR356$numero)
        Friedman rank sum test
data:  tablaR356$peso, tablaR356$edad and tablaR356$numero
Friedman chi-squared = 20, df = 2, p-value = 4.54e-05
```

también se puede resolver con el siguiente código:

```
> friedman.test(peso~edad|numero, tablaR356)
        Friedman rank sum test
data:  peso and edad and numero
Friedman chi-squared = 20, df = 2, p-value = 4.54e-05
```

Para el test de Quade utilizamos el código siguiente

```
> quade.test(tablaR356$peso,groups=tablaR356$edad,blocks=tablaR356$numero)
        Quade test
data:  tablaR356$peso, tablaR356$edad and tablaR356$numero
Quade F = 33.201, num df = 2, denom df = 18, p-value = 9.126e-07
```

tambien se puede resolver con el código

```
> quade.test(peso~edad|numero, tablaR356)
        Quade test
data:  peso and edad and numero
Quade F = 33.201, num df = 2, denom df = 18, p-value = 9.126e-07
```

En ambos casos groups, hace referencia a los grupos experimentales, es decir los diferentes días de medición. El argumento blocks hace referencia a cada unidad experimental donde se realizaron las mediciones sucesivas.

No puede haber valores NA, si los hay el bloque es eliminado. Que se entiende por bloque es un grupo de mediciones realizadas sobre una misma unidad experimental. En este caso cada animal identificado por un número.
El test de Quade tiene mayor potencia cuando los tratamientos (groups) son pocos. Contrariamente la potencia es mayor para el test de Friedman si son muchos los tratamientos (groups)

6. Clase 3.6

Video: https://youtu.be/baZ605q-2qw
Tabla de datos:

6.1. Comparación de más de dos muestras (continuación)

La comparación de más de dos grupos es habitual en los trabajos experimentales. El análisis de la variancia a un criterio de clasificación completamente aleatorizado es un modelo que permite el análisis de estos casos. En este análisis se tienen grupos experimentales y a cada unidad experimental de cada grupo se les aplica un nivel de un tratamiento asignado al azar.

En estos estudios en primer lugar se investiga si hay diferencias entre los grupos. De existir dicha diferencia tenemos dos tareas por delante:

1- Evaluar la potencia del ensayo

2- Realizar las comparaciones múltiples

En esta clase veremos ambos temas aplicados a datos con distribución normal y variancias homogéneas, orientándonos al uso de las herramientas más versátiles de R en ese campo. Esta decisión se fundamenta en que se dispone también de herramientas que permiten transformar los datos sin distribución normal en datos con distribución normal.

Veamos el proceso completo!

Supongamos que hemos medido el peso de animales de experimentación con tres tratamiento distintos (a, b,c), datos que hallamos en la tablaR361 de la planilla de cálculo tablaR3-6.ods/xls.

Debemos ejecutar diferentes acciones y dar varias respuestas. Suponiendo que ya hemos realizado el experimento y tenemos los datos debemos

1- Introducir los datos en el espacio de trabajo.

2- Determinar si los datos son dependientes o independientes.

3- Determinar si los datos tienen distribución normal o no.

4- Evaluar si sus variancias son homogéneas o no.

5- Luego confrontaremos la hipótesis que "no existe diferencias entre los tratamientos" contra la hipótesis alternativa que "sí existen diferencias entre los tratamientos". Es decir realizaremos el análisis de la variancia a un criterio de clasificación (ANOVA a 1 criterio)

6- De comprobarse dicha diferencia, calcularemos la potencia del ensayo.

7- Realizaremos las comparaciones múltiples entre los tratamientos, intentando determinar entre qué tratamientos existe diferencia.

A continuación resolveremos dos casos completos que podríamos enmarcar dentro de los análisis paramétricos y no paramétricos

6.2. Resolución de un caso paramétrico completo

Dentro de este título desarrollaremos como subtítulos cada uno de los pasos a seguir para realizar un análisis complete. Anticipamos que en este caso nos hallaremos con muestras con distribución normal, variancias homogéneas y datos independientes. Comenzamos con:

6.2.1. Introducción de datos

```
> tablaR361<-read.table("clipboard",header=T, sep="\t",dec=",",encoding="latin1")
```

```
> tablaR361
   numero tratamiento peso
1       1           a  127
2       2           a  130
3       3           a  125
4       4           a  126
5       5           a  121
6       6           a  115
7       7           a  120
8       8           a  121
9       9           a  126
10     10           a  114
11     11           b  152
12     12           b  154
13     13           b  154
14     14           b  155
15     15           b  146
16     16           b  151
17     17           b  156
18     18           b  154
19     19           b  145
20     20           b  151
21     21           c  126
22     22           c  127
23     23           c  125
24     24           c  127
25     25           c  132
26     26           c  124
27     27           c  124
28     28           c  129
29     29           c  128
30     30           c  127
```

6.2.2. Determinar tipo de datos (dependientes o independientes)

1- En este caso los datos son independientes. Tenemos treinta ratas a las que se dividieron en tres grupos y a cada grupo se aplicó un tratamiento distinto: a, b, c.

Corresponde a un modelo completamente aleatorizado, donde cada unidad experimental recibió un tratamiento asignado al azar. Hay tres tratamientos diferentes y la resolución se realizará con un análisis de la variancia a un criterio, conocido como ANOVA A UN CRITERIO.

6.2.3. Tipo de distribución de probabilidad (normal o no)

Para determinar si los valores de peso tienen distribución normal aplicamos el test de Shapiro Wilk individualmente para cada tratamiento.

```
> shapiro.test(tablaR361$peso[tablaR361$tratamiento=="a"])
        Shapiro-Wilk normality test
data:  tablaR361$peso[tablaR361$tratamiento == "a"]
W = 0.93574, p-value = 0.5066 ⇒ la muestra tiene distribución normal
```

```
> shapiro.test(tablaR361$peso[tablaR361$tratamiento=="b"])
        Shapiro-Wilk normality test
data:  tablaR361$peso[tablaR361$tratamiento == "b"]
W = 0.8689, p-value = 0.09705
```
⇒ la muestra tiene distribución normal

```
> shapiro.test(tablaR361$peso[tablaR361$tratamiento=="c"])
        Shapiro-Wilk normality test
data:  tablaR361$peso[tablaR361$tratamiento == "c"]
W = 0.92638, p-value = 0.4133
```
⇒ la muestra tiene distribución normal

6.2.4. Test de homogeneidad de variancias

Para la prueba de homogeneidad de variancias podemos aplicar varios test: bartlett.test, var.test o fligner.test. Como tenemos tres tratamientos para comparar las variancias de los tres grupos nos convendrá el test de Bartlett.

```
>
bartlett.test(list(tablaR361$peso[tablaR361$tratamiento=="a"],tablaR361$peso[tablaR361$tratamiento=="b"],tablaR361$peso[tablaR361$tratamiento=="c"]))
        Bartlett test of homogeneity of variances
data:  list(tablaR361$peso[tablaR361$tratamiento == "a"], tablaR361$peso[tablaR361$tratamiento ==
    "b"], tablaR361$peso[tablaR361$tratamiento == "c"])
Bartlett's K-squared = 4.7468, df = 2, p-value = 0.09317
```
⇒ la variancias son homogéneas

6.2.5. Contraste de hipótesis: ANOVA a 1 criterio

Por el diseño acetamos que los valores son independientes y por los análisis previos de normalidad y homogeneidad de variancias aceptamos que los datos tienen distribución normal y las variancias son homogéneas. Por lo tanto podemos aplicar un test paramétrico para comparar las muestras y utilizaremos la funcion aov() que es útil para esta situación. Aplicaremos la función y su resultado lo asignamos al objeto aovtablaR361, que nos permitirá análisis posteriores. Esta función analiza el peso en función (~) de los tratamientos

```
> aovtablaR361<-aov(peso~tratamiento,data=tablaR361)
```

```
> summary(aovtablaR361)
             Df Sum Sq Mean Sq F value   Pr(>F)
tratamiento   2   4993  2496.4   159.3 1.12e-15 ***
Residuals    27    423    15.7
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```
⇒ hay efecto de los tratamientos

El valor de p-value que en esta tabla figura como Pr(>F) es menor de 0.05 por lo que podemos afirmar que existe diferencia de peso entre los tratamientos aplicados. Es decir descartamos la hipótesis nula y nos quedamos con la alternativa. Por supuesto que al haber tomado esta decisión podemos estar cometiendo un error de tipo II, es decir haber elegido la situación que son diferentes cuando en realidad los pesos no diferían. Entonces calcularemos la potencia del ensayo, es decir la probabilidad que tenemos de haber elegido la situación que hay diferencia, cuando en realidad esta diferencia existe. Dicho en otras palabras calcularemos la probabilidad de estar en lo cierto.

6.2.6. Cálculo de la potencia del ensayo.

El paquete pwr tiene dos funciones que permiten calcular la potencia de una análisis de la variancia a un criterio balanceado. Es decir que sirve para cuando tenemos un factor en estudio, en

nuestro caso el tratamiento y el número de animales por grupo es igual en todos los grupos. Si no lo fueran el test no es el más adecuado, pero se puede lograr una aproximación tomando como n, el promedio de las unidades experimentales de cada grupo

Las funciones son power.anova.test() y pwr.anova.test()

Dan el mismo resultado, pero llevan datos distintos.

La forma de escritura de ambas funciones son

power.anova.test(groups = NULL, n = NULL, between.var = NULL, within.var = NULL, sig.level = 0.05, power = NULL)

Argumentos: de los cinco argumentos indicados con valor NULL, cuatro deben ser asignados y uno quedará para calcular, en nuestro caso dejaremos power con el valor NULL, ya que deseamos calcular la potencia del ensayo.

Analicemos cada argumento

groups: es el número de tratamientos: 3 en nuestro caso.

n: el el número de individuos por grupo: 10 en nuestro caso.

between.var: variancia entre grupos. Este valor en la tabla de anova dada más arriba es el valor que figura en la columna "Mean Sq" y fila "tratamientos": en nuestro caso 2496.4

within.var: es la variancia residual, en la tabla de anova figura en la columna "Mean Sq" fila "Residuals": en nuestro caso 15.7

sig.level: probabilidad de error tipo I: en nuestro caso 0.05

power: potencia del ensayo

```
>
power.anova.test(groups=3,n=10,between.var=2496.4,within.var=15.7,sig.level=0.05,power=NUL
L)
     Balanced one-way analysis of variance power calculation
    groups = 3
    n = 10
     between.var = 2496.4
     within.var = 15.7
     sig.level = 0.05
     power = 1
NOTE: n is number in each group
```

La potencia de nuestro ensayo es 1. Es decir que rechazamos la hipótesis nula con una probabilidad de error de 5%, aceptando que los tratamientos tienen efecto diferente sobre el peso de los animales. Además tenemos un 100% de probabilidad de estar en lo cierto al decir que son diferentes. Rara vez ocurrirá nuevamente esto que la potencia del ensayo sea 1, en general es menor que 1 y aceptamos que es un buen ensayo cuando la potencia es mayor a 0.8.

Para utilizar la otra función, debemos conocer la magnitud del efecto, cuya fórmula para anova balanceado es

$$f=\sqrt{\frac{\sum n_i/N*(\overline{x}_i-\overline{x})^2}{\sigma^2}}$$

donde

ni: número de observaciones por grupo

N: número total de observaciones

xi = media del grupo i

x= media general (gran media)

sigma2 = error variance within groups

Calculemos entonces la magnitud del efecto en nuestro caso y para ellos comencemos calculando la media de cada tratamiento: xi

> mean(tablaR361$peso[tablaR361$tratamiento=="a"])

[1] 122.5

> mean(tablaR361$peso[tablaR361$tratamiento=="b"])

[1] 151.8

>

[1] 126.9 mean(tablaR361$peso[tablaR361$tratamiento=="c"])

y calculemos la media general, es decir la media de los pesos sin separar por tratamiento

> mean(tablaR361$peso)

[1] 133.7333

Con estos datos calculamos la magnitud del efecto o effect size

$$ez=\sqrt{\frac{10/30*(122,5-133,7333)^2+10/30*(151,8-133,7333)^2+10/30*(126,9-133,7333)^2}{15,7}}$$

reemplazamos los datos en la fórmula anterior

>ez=sqrt((10/30*(122.5-133.7333)^2+10/30*(151.8-133.73333)^2+10/30*(126.9-133.7333)^2)/15.7)

y vemos el valor que adquirió ez

> ez

[1] 3.255848

con este valor y los otros conocidos aplicaremos la otra función para calcular la potencia del ensayo.

pwr.anova.test(k = NULL, n = NULL, f = NULL, sig.level = 0.05, power = NULL)

describamos los argumentos

k= número de grupos, en nuestro caso 3.

n= número de observaciones por grupo, en nuestro caso toma el valor 10.

f= magnitud del efecto o effect size, que en nuestro caso se halla en la variable ez que acabamos de calcular.

sig.level = probabilidad de error de tipo I o p-value que fijamos al aplicar la función aov(). En nuestro caso tomamos p=0.05 para aceptar o rechazar la hipótesis nula.

power: potencia del ensayo.

```
> pwr.anova.test(k = 3, n = 10, f = ez, sig.level = 0.05, power = NULL)
   Balanced one-way analysis of variance power calculation
   k = 3
     n = 10
    f = 3.255848
     sig.level = 0.05
    power = 1
NOTE: n is number in each group
```

Arribamos al mismo resultado que con la función power.anova.test(). Esta fórmula es más útil cuando se desea calcular el número de datos necesarios para el experimento, por ejemplo en la situación que se conozca la magnitud del efecto por datos de la bibliografía o de algún experimento piloto.

6.2.7. Comparaciones múltiples.

La función aov() nos indicó que existen diferencias entre los grupos, pero no nos indicó entre qué grupos existe dicha diferencia. Para dar respuesta a este interrogante debemos realizar un test de comparaciones múltiples. Existen diferentes formas de realizar comparaciones múltiples, siendo para este caso la más completa en lo que respecta a su respuesta, la función LSD.test() de la biblioteca agricolae (instálela si aun no la tiene). Si la tiene instalada entonces cárguela en su espacio de trabajo

```
> library(agricolae)
```

Esta función trabaja con el objeto generado a partir de la función aov(), por lo tanto en principio será aplicable solo a aquellas comparaciones en que hay distribución normal y variancias homogéneas.

Veremos como extender esto luego al resto de los casos

Dado que el análisis con la función aov() ha quedado más arriba, volveremos a repetirlo acá

```
>aovtablaR361<-aov(peso~tratamiento,data=tablaR361)

> summary(aovtablaR361)
             Df  S um Sq  Mean Sq  F value  Pr(>F)
tratamiento   2    4993   2496.4    159.3  1.12e-15 ***
Residuals    27     423     15.7
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Como vimos, el valor Pr(>F) que representa el p-value nos permite afirmar que el tratamiento afecta el peso. ¿Pero son los pesos de los tratamientos todos diferentes entre sí o solo algunos los

tratamientos difieren entre sí?

Para dar la respuesta a este interrogante aplicamos la función LSD.test(), cuyo valor lo almacenaremos en un objeto para su posterior análisis. Como podemos ver con LSD.test() analizamos el resultado de haber aplicado la función aov() a la tablaR361 y lo que queremos es ver es entre qué "tratamiento" existe diferencia. Es decir dentro de LSD.test() indicaremos la tabla obtenida luego de aplicar aov() y como segundo argumento indicamos la columna que contiene el factor que estamos analizando como fuente de variación de la variable en estudio.

```
> lsdaovtablaR361<-LSD.test(aovtablaR361,"tratamiento")
```

En este caso para un anova a un criterio solo se debe especificar el objeto obtenido luego del aov() y la variable que deseamos ver que efecto tiene sobre la variable investigada, en este caso el peso

Para ver qué contiene el objeto obtenido con la función LSD.test() ejecutamos

```
> lsdaovtablaR361
$statistics
   Mean       CV        MSerror      LSD
 133.7333 2.959706 15.66667 3.631991

$parameters
  Df  ntr  t.value       alpha       test              name.t
  27   3     2.051831  0.05       Fisher-LSD   tratamiento

$means
   peso      std          r      LCL          UCL          Min Max
a 122.5   5.233439  10    119.9318    125.0682    114 130
b 151.8   3.705851  10    149.2318    154.3682    145 156
c 126.9   2.424413  10    124.3318    129.4682    124 132

$comparison
NULL

$groups
    trt   means   M
1   b     151.8    a
2   c      126.9   b
3   a      122.5   c
```

El análisis nos muestra varios resultados, pero la tabla $groups nos muestra los resultados más interesantes del análisis. La tabla muestras las medias de los pesos en la columna 3: means. En la columna 2 nos muestra los tratamientos: trt). En la cuarta columna (M) se observa una letra. ¿Cómo se interpreta esta letra?. Al menos una letra igual entre dos filas indica que no hay diferencias significativas entre las medias de los tratamientos de dichas filas. En cambio si todas las letras son distintas entre dos filas indica que existe diferencia significativa entre las medias de los tratamientos comparados. En este caso los tres tratamientos tienen medias diferentes, ya que cada fila tiene una letra distinta en la columna M. Como consecuencia concluimos en base al hallazgo con la función aov() que existen diferencias entre los pesos de los animales con distinto tratamiento. En base a la función LSD.test() concluimos que las medias de los tres grupos son diferentes.

6.2.8. Cálculo de número de unidades por grupo para una ANOVA balanceado

Si ya hemos aplicado la función aov() hallando diferencias entre los grupos, luego aplicamos la función LSD.test() para determinar entre que grupos existe dicha diferencia. Si se dió ese caso, este ítem no es aplicable. Sin embargo sí puede ser útil si no hubieramos hallado diferencias significativas con aov() o bien aun no hubieramos realizado el experimento y no sabemos con cuantas unidades experimentales trabajar. El cálculo de unidades experimentales que debemos utilizar a la hora de probar una dada hipótesis es siempre un problema latente en la investigación. El paquete pwr, que hemos utilizado suele darnos grandes respuestas a este interrogante. Muchas veces hemos realizado un experimento piloto para ver si tratamientos aplicados pueden producir cambios en una dada variable. El paquete pwr nos da las herramientas para calcular el número de unidades que deberíamos tener en cada grupo para demostrar con un error de tipo I y potencia adecuada que tres o más tratamientos producen modificaciones de la variables estudiadas.

Veamos un ejemplo.

Hemos medido la glucemia en tres animales por grupo (experimento piloto) bajo tres tratamiento I, II y III. Nuestra hipótesis es que los tratamientos deberían modificar los valores de glucemia, generando valores diferentes en cada grupo. En otras palabras nuestra hipótesis es que los animales de grupos distintos deberían tener diferentes valores de la glucemia.

Veamos la forma de calcular el número de datos que necesitaríamos para realizar una prueba de hipótesis utilizando los datos del experimento piloto que se hallan en la tablaR362. Con ellos realizamos una ANOVA a un criterio con la función aov()

Introduzcamos primero los datos

```
> tablaR362<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")
```

Realicemos el análisis de la variancia, intendo determinar si con los datos de este experimento piloto podemos ver diferencias en la glucemia en función del tratamiento

```
> aovtablaR362<-aov(glucemia~tratamiento,data=tablaR362)
```

Pidamos un informe del objeto creado

```
> summary(aovtablaR362)
             Df Sum Sq  Mean Sq  F value  Pr(>F)
tratamiento   2  0.1480  0.07401    4.945  0.0538 .⇒ el efecto de los tratamientos no es significativo
Residuals     6  0.0898  0.01497
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

si bien por el valor de p-value el tratamiento no es significativo, vemos que está muy cerca de un valor 0.05. Seguramente con mayor número de datos nos daría significativo. Más adelante veremos como calcular el número de unidades que tendría que tener cada grupo para que la potencia de nuestro ensayo fuera del 80% o más.

Antes analicemos cual sería la potencia del ensayo si rechazáramos la hipótesis nula y nos quedáramos con que los tratamientos afectan la glucemia. El error de tipo I de nuestro ensayo sería muy cercano a 0.05. Recuerde que en general rechazamos la hipótesis nula cuando $p<0.05$. Acá nos hubiéramos tomado el atrevimiento de rechazar la hipótesis nula con un error de tipo I un poco mayor. Es decir ya no estaríamos tan seguros de haber rechazado la hipótesis que los tratamientos no afectan la glucemia, por lo tanto seguramente tampoco estaremos tan seguros que los tratamientos sí afectan la glucemia. Veremos esto con la potencia del ensayo

Para ello calcularemos una serie de estadísticas necesarias para obtener la magnitud del efecto.

Primero la media general

```
> mean(tablaR362$glucemia)
[1] 0.9844444
```

luego la media por tratamientos

```
> mean(tablaR362$glucemia[tablaR362$tratamiento=="I"])
[1] 1.066667
> mean(tablaR362$glucemia[tablaR362$tratamiento=="II"])
[1] 0.8033333
> mean(tablaR362$glucemia[tablaR362$tratamiento=="III"])
[1] 1.083333
```

finalmente la magnitud del efecto

```
> ez=sqrt((3/9*(1.06-0.98)^2+3/9*(0.803-0.98)^2+3/9*(1.08-0.98)^2)/0.01497)
> ez
[1] 1.030907
```

y con este valor de ez, calculamos la potencia de nuestro ensayo.

k que es el número de tratamientos toma el valor 3.

n que es el número de observaciones dentro de cada tratamiento, también toma el valor 3

fijamos sig.level en 0.05,que es el p-value del aov() realizado.

```
> pwr.anova.test(k=3,n=3,f=ez,sig.level=0.05,power=NULL)
    Balanced one-way analysis of variance power calculation
             k = 3
             n = 3
             f = 1.030907
     sig.level = 0.05
       power = 0.5600897
NOTE: n is number in each group
```

La potencia de nuestro ensayo es de 0.56, es decir del 56%. Es decir, si rechazamos la hipótesis que los tratamientos no afectan la glucemia, inclinándonos por que si hay diferencias en la glucemia entre los tratamientos aplicado, esta afirmación la podemos sostener con un 56% de probabilidad de estar en lo cierto. Un poco bajo para sostener una hipótesis.

Entonces calcularemos cuantos animales tendríamos que tener por grupo para que siendo el alfa=0.05 la potencia fuera de 0.8. Aplicamos la función pwr.anova.test(), pero ahora dejamos como dato variable n, es decir el número de unidades experimentales por tratamiento

```
> pwr.anova.test(k=3,n=NULL,f=ez,sig.level=0.05,power=0.8)
   Balanced one-way analysis of variance power calculation
            k = 3
              n = 4.199828
              f = 1.030907
     sig.level = 0.05
```

power = 0.8
NOTE: n is number in each group
Vemos que si repetimos el experimento con 4 y mejor con 5 animales (ya que el ensayo nos dio n= 4.199828) por grupo seguramente rechazaremos la hipótesis nula (que no hay diferencia de glucemia entre tratamientos) con una probabilidad de equivocarnos del 5% y por otro lado podremos afirmar que son diferentes las glucemias entre los tratamientos, con una probabilidad de estar en lo cierto del 80%.

6.3. Resolución de un caso no paramétrico completo

En este caso resolveremos un caso de comparación de más de dos grupos de datos independientes, pero que no cumplen el supuesto de normalidad y/o de homogeneidad de variancias. Para ello introducimos en nuestro espacio de trabajo la tablaR364, que corresponden a pesos medidos en tres grupos de animales donde cada uno recibió uno de los tres tratamientos: a, b, c.

```
> tablaR364<-read.table("clipboard",header=T, sep="\t",dec=",",encoding="latin1")
```

observamos los datos

```
> head(tablaR364)
  numero tratamiento peso
1      1          a  127
2      2          a  130
3      3          a  125
4      4          a  126
5      5          a  121
6      6          a  115
```

realizamos los análisis de normalidad de cada tratamiento

```
> shapiro.test(tablaR364$peso[tablaR364$tratamiento=='a'] )
        Shapiro-Wilk normality test
data:  tablaR364$peso[tablaR364$tratamiento == "a"]
W = 0.93574, p-value = 0.5066
```

p>0.05 aceptamos distribución normal

```
> shapiro.test(tablaR364$peso[tablaR364$tratamiento=='b'])
        Shapiro-Wilk normality test
data:  tablaR364$peso[tablaR364$tratamiento == "b"]
W = 0.83961, p-value = 0.04365
```

p<0.05 rechazamos la hipótesis de distribución normal

```
> shapiro.test(tablaR364$peso[tablaR364$tratamiento=='c'])
        Shapiro-Wilk normality test
data:  tablaR364$peso[tablaR364$tratamiento == "c"]
W = 0.92638, p-value = 0.4133
```

p>0.05 aceptamos distribución normal

No tiene sentido evaluar las variancias, ya que no cumplen todos los grupos con los criterios de normalidad. Por lo tanto para evaluar si existe diferencia entre los grupos utilizaremos la función kruskal.test()

```
>  kruskal.test(peso~tratamiento,data= tablaR364)
        Kruskal-Wallis rank sum test
data:  peso by tratamiento
```

Kruskal-Wallis chi-squared = 21.208, df = 2, p-value = 2.482e-05

por el valor de p-value<0.05 aceptamos que existen diferencias entre los grupos. Por lo cual utilizaremos ahora un test de comparaciones múltiples que nos permita decir entre que grupos existe esta diferencia. Para ello utilizamos la función pairwise.wilcox.test() que es un test de comparaciones múltiples para cuando los datos no tienen distribución normal.

```
> pairwise.wilcox.test(tablaR364$peso,tablaR364$tratamiento,paired=FALSE)
        Pairwise comparisons using Wilcoxon rank sum test
data:  tablaR364$peso and tablaR364$tratamiento
        a          b
b 0.00052 -
c 0.05237 0.00052
```

Para la conclusión miramos la intersección de filas y columnas. En las columnas tenemos los tratamientos a y b mientras que en las filas tenemos b y c. En las intersecciones tenemos los p-value de cada comparación, si es menor de 0.05 existe diferencias entre los grupos

Conclusión:

el grupo a difiere de b, ya que p-value<0.05, en este caso vale 0.00052

el grupo a no difiere de de c, ya que p-value>0.05, en este caso vale 0.05237

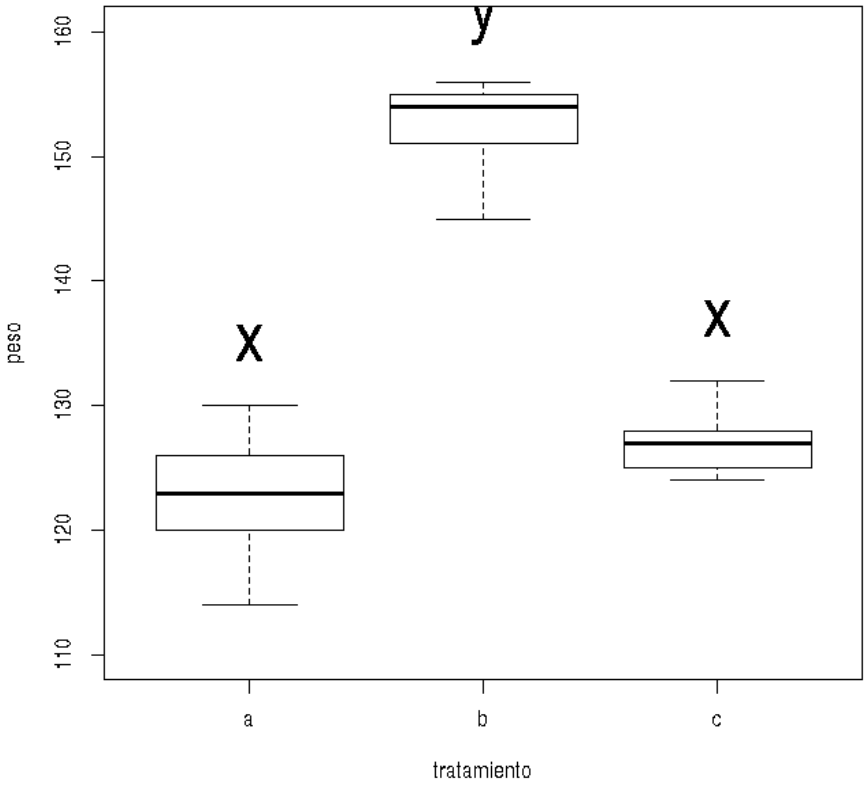
el grupo b difiere de de c, ya que p-value<0.05, en este caso vale 0.00052

Para representar esto podemos asignar letras distintas a los grupos diferentes y así construir una gráfica que nos permita mostrar los valores. Por ejemplo basándonos en la tabla anterior

```
        a          b
b 0.00052 -
c 0.05237 0.00052
```

como a difiere de b, al grupo a le damos la letra x y al grupo b, la letra y
por otra parte como a no difiere de c, al grupo c le damos la letra x
Todas las letras distintas entre dos grupos nos indicarán diferencias significativas y al menos una letra igual nos indicará que no hay diferencia.
Entonces graficamos con los siguientes códigos

```
> plot(tablaR364$peso~tablaR364$tratamiento,ylab='peso',xlab='tratamiento',ylim=c(110,160))
> text(1,max(tablaR364$peso[tablaR364$tratamiento=='a']+5),'x',cex=3)
> text(2,max(tablaR364$peso[tablaR364$tratamiento=='b']+5),'y',cex=3)
> text(3,max(tablaR364$peso[tablaR364$tratamiento=='c']+5),'x',cex=3)
```



Claramente la gráfica nos indica que los tratamientos a y c no difieren mientras que b difiere de los otros dos.

7. Clase 3.7

Video: https://youtu.be/LbvbYW5e3gs
Tabla de datos:

7.1. Análisis de la variancia a dos factores

Hasta ahora hemos visto resultados de experimentos en los que una variable fue afectada por un factor, que se podía presentar a dos niveles o a más de dos niveles. Por ejemplo "la variable" glicemia fue medida en unidades experimentales donde un grupo de ellas recibió el "tratamiento a" y otro grupo el "tratamiento b". Los tratamientos a y b serían los dos niveles del factor Tratamiento.

En el caso que el valor de la variable era medido en unidades experimentales en presencia de un factor a dos niveles diferentes, las comparaciones entre los valores las resolvimos con las funciones: t.test() o wilcox.test() según los datos tuvieran o no distribución normal y/o variancias homogéneas. Por otra parte en ambos caso fijamos el argumento paired en el valor TRUE o FALSE, dependiendo que los datos fueran apareados o no.

Hablando mal y pronto, en estas situaciones simplificamos diciendo que la comparación se hizo con una t de Student, aunque ya sabemos que pueden ser otros test según la característica de los datos.

Cuando el valor de una variable fue medido en unidades experimentales donde cada una de ellas estuvo sometida a un nivel de un factor, y este factor se presenta en el experimento a 3 o más niveles posibles, la comparación de los valores de las variables entre los grupos con cada nivel del factor se realizó utilizando las funciones aov(), kruskal.test(), friedman.test() o quade.test(), dependiendo que los datos tuvieran: distribución normal o no y que fueran aparedos o no.

Hablando nuevamente mal y pronto, en este caso decimos que la comparación se realizó con un ANOVA a un criterio, aunque sabemos que pueden ser otros varios test diferentes que utilizaremos según el caso.

En esta clase veremos la medición de una variable en unidades experimentales, donde cada una de ellas está sometida a un nivel de dos o más factores. Por ejemplo, estamos estudiando el peso corporal (la variable) y tenemos entre nuestras unidades experimentales hombres y mujeres (dos niveles del factor sexo) y además tenemos entre los individuos en estudio aquellos que tienen peso normal, bajo peso, sobrepeso y obesidad (cuatro niveles del factor peso). Así vemos que tendremos 8 grupos experimentales, correspondientes a las combinaciones de los niveles de los factores. Entonces lo que deseamos ver es que factor influye sobre la variable y cuales son las combinaciones de los niveles de los factores que inducen cambios significativos en la variable.

Hablando mal y pronto, decimos que analizaremos los datos con una ANOVA A DOS CRITERIOS.

Cuando analizamos datos en los que participan dos o más factores, además podemos evaluar la interacción entre los factores. Haremos un ejemplo sin analizar la interacción y analizándola

7.1.1. Anova a dos criterios (sin interacción)

Veamos entonces un ejemplo. Para ello introduzcamos la tablaR371 que se halla en la planilla de cálculo tablaR3-7.xls/ods.

Esta planilla muestra valores medidos de insulina de 40 animales que están afectados por dos

factores (sexo y tratamiento), a su vez el sexo está a dos niveles (m:machos y h: hembras) y el tratamiento también tiene dos niveles (t1 t t2), donde t1 corresponde a animales sedentarios y t2 a animales que realizan ejercicio físico.

```
> tablaR371<-read.table("clipboard",header=T,dec=",",sep="\t",encoding="latin1")
```

```
> tablaR371
   numero sexo tratamiento insulina
1       1    m         t1      200
2       2    m         t1      210
3       3    m         t1      350
4       4    m         t1      170
5       5    m         t1      240
6       6    m         t1      250
7       7    m         t1      200
8       8    m         t1      250
9       9    m         t1      220
10     10    m         t1      230
11     11    m         t2      140
12     12    m         t2      150
13     13    m         t2      290
14     14    m         t2      110
15     15    m         t2      181
16     16    m         t2      190
17     17    m         t2      140
18     18    m         t2      189
19     19    m         t2      160
20     20    m         t2      171
21     21    h         t1      220
22     22    h         t1      240
23     23    h         t1      370
24     24    h         t1      200
25     25    h         t1      260
26     26    h         t1      270
27     27    h         t1      220
28     28    h         t1      260
29     29    h         t1      240
30     30    h         t1      250
31     31    h         t2      160
32     32    h         t2      180
33     33    h         t2      311
34     34    h         t2      140
35     35    h         t2      200
36     36    h         t2      210
37     37    h         t2      160
38     38    h         t2      202
39     39    h         t2      180
40     40    h         t2      190
```

De observar la tabla podemos a clasificar a los 40 animales estudiados en cuatro grupos

machos con t1

machos con t2

hembras con t1

hembras con t2

Analizamos la normalidad de los datos en conjunto

```
> shapiro.test(tablaR371$insulina)
        Shapiro-Wilk normality test
data:  tablaR371$insulina
W = 0.95473, p-value = 0.1104
```

Podemos aceptar que tienen distribución normal. Analizamos ahora dentro de los machos los dos tratamientos

```
> shapiro.test(tablaR371$insulina[tablaR371$sexo=="m" & tablaR371$tratamiento=="t1"])
        Shapiro-Wilk normality test
data:  tablaR371$insulina[tablaR371$sexo == "m" & tablaR371$tratamiento ==     "t1"]
W = 0.85517, p-value = 0.06691

> shapiro.test(tablaR371$insulina[tablaR371$sexo=="m" & tablaR371$tratamiento=="t2"])
        Shapiro-Wilk normality test
data:  tablaR371$insulina[tablaR371$sexo == "m" & tablaR371$tratamiento ==     "t2"]
W = 0.85537, p-value = 0.06727
```

Por los valores de p-value aceptamos que cada grupo tiene distribución normal. Procedemos de igual manera con las hembras

```
> shapiro.test(tablaR371$insulina[tablaR371$sexo=="h" & tablaR371$tratamiento=="t1"])
        Shapiro-Wilk normality test
data:  tablaR371$insulina[tablaR371$sexo == "h" & tablaR371$tratamiento ==     "t1"]
W = 0.8103, p-value = 0.01933

> shapiro.test(tablaR371$insulina[tablaR371$sexo=="h" & tablaR371$tratamiento=="t2"])
        Shapiro-Wilk normality test
data:  tablaR371$insulina[tablaR371$sexo == "h" & tablaR371$tratamiento ==     "t2"]
W = 0.81217, p-value = 0.02037
```

Si tomamos el nivel de significación en 0.01, no podemos descartar la hipótesis nula, es decir que no podemos rechazar la hipótesis que los valores de insulina tienen distribución normal. Por ende aceptamos normalidad de nuestros datos y seguiremos adelante evaluando la homogeneidad de las variancias de los grupos.

Ahora analicemos la homogeneidad de variancias con el test de Bartlett, para los cuatro grupos de animales

```
> bartlett.test(list(tablaR371$insulina[tablaR371$sexo=="h" &
tablaR371$tratamiento=="t1"],tablaR371$insulina[tablaR371$sexo=="h" &
tablaR371$tratamiento=="t2"],tablaR371$insulina[tablaR371$sexo=="m" &
tablaR371$tratamiento=="t1"],tablaR371$insulina[tablaR371$sexo=="m" &
tablaR371$tratamiento=="t2"]))
```

```
        Bartlett test of homogeneity of variances

data: list(tablaR371$insulina[tablaR371$sexo == "h" & tablaR371$tratamiento == "t1"],
tablaR371$insulina[tablaR371$sexo == "h" & tablaR371$tratamiento == "t2"],
tablaR371$insulina[tablaR371$sexo == "m" & tablaR371$tratamiento == "t1"],
tablaR371$insulina[tablaR371$sexo == "m" & tablaR371$tratamiento == "t2"])
Bartlett's K-squared = 0.027232, df = 3, p-value = 0.9988
```

Como el valor de p-value es mayor a 0.05, aceptamos la hipótesis que las variancias son homogéneas. Entonces podemos realizar el ANOVA a dos criterios. Para ello utilizamos la función aov() y evaluamos los valores de insulina en función de las variables tratamiento y sexo. Para expresar que la insulina se analiza en función de tratamiento y sexo se utiliza el símbolo: ~. Además como hacemos una análisis sin interacción entre las variables, tratamiento y sexo en el código van separadas por un signo +

```
> aovtablaR371<-aov(insulina~tratamiento+sexo,data=tablaR371)
pedimos un summary()
> summary(aovtablaR371)
             Df Sum Sq  Mean Sq  F value    Pr(>F)
tratamiento  1  35760   35760      16.258    0.000265 ***
sexo         1   4452    4452       2.024    0.163202
Residuals   37  81385    2200
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

El valor Pr(>F) es el p-value y en la tabla anterior nos indica que el tratamiento tiene un efecto significativo sobre el valor de insulina ya que el valor del p-value es menor a 0.05 (valor de la tabla: 0.000265 ***), mientras que el sexo no es un factor que afecte el valor de insulina dado que su p-value es mayor a 0.05 (valor de tabla: 0.163202 **).** Es decir que nuestra primer conclusión es que los valores de insulina depende del tratamiento de cada unidad experimental pero no del sexo, por supuesto con una probabilidad de error de tipo I de 0.05.

Ahora podremos comparar los datos de los cuatro grupos experimentales utilizando un test de comparaciones múltiples, como es el LSD.test, que ya hemos utilizados. Hasta ahora hemos demostrado que el tratamiento influye en los valores de insulina. Deseamos concer cuales de los cuatro grupos difieren entre sí. Introduciremos ahora algunos conceptos adicionales.

Cargamos la biblioteca agricolae y realizamos el LSD.test. Este test se aplica sobre el objeto en el guardamos los datos de la función aov() a dos criterios y nos interesan los factores tratamiento y sexo. A continuación se muestra la forma de organizar las variables dentro de la función LSD.test(). El resultado lo guardaremos en el objeto lsdaovtablaR371. La función LSD.test() la aplicaremos al objeto aovtablaR371 y consideraremos las variables tratamiento y sexo. En análisis lo haremos con un nivel de significación error alfa de 0.05 y aplicaremos dentro de las comparaciones múltiples la prueba de Bonferroni. El argumento group=T permite visualizar los datos de una forma amigable.

```
>lsdaovtablaR371<-
LSD.test(aovtablaR371,c("tratamiento","sexo"),alpha=0.05,p.adj="bonferroni",group=T)
> lsdaovtablaR371
$statistics
  Mean       CV  MSerror      LSD
 212.6 22.06014 2199.597 58.46789
```

$parameters
  Df ntr bonferroni alpha   test   name.t
  37   4   2.787602 0.05 bonferroni tratamiento:sexo

$means
         insulina   std   r   LCL   UCL   Min Max
t1:h   253.0   46.43993 10   222.9495 283.0505 200 370
t1:m   232.0   48.48826 10   201.9495 262.0505 170 350
t2:h   193.3   46.75717 10   163.2495 223.3505 140 311
t2:m   172.1   48.46408 10   142.0495 202.1505 110 290

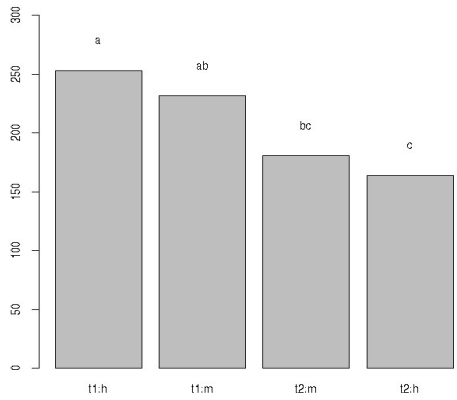
$comparison
NULL

$groups
       trt   means M
1   t1:h   253.0   a
2   t1:m   232.0 ab
3   t2:h   193.3 bc
4   t2:m   172.1 c

Esta última tabla tiene el resumen del análisis. Todas las letras distintas en la última columna, entre dos filas cualquiera indica diferencias significativas. Haciendo un análisis podemos decir que los animales hembras (h) con tratamiento t1 tienen un valor de insulina que no difiere de los machos (m) con tratamiento t1, pero si de los m y h con tratamiento 2. A su vez, siguiendo el mismo razonamiento en la línea 2 de la tabla, los animales machos con tratamiento 1 (t1:m) no difieren de t1:h ni de t2:h (ya que comparten la letra **a** con un grupo y la **b** con el otro), pero sí son diferentes de los animales t2:m

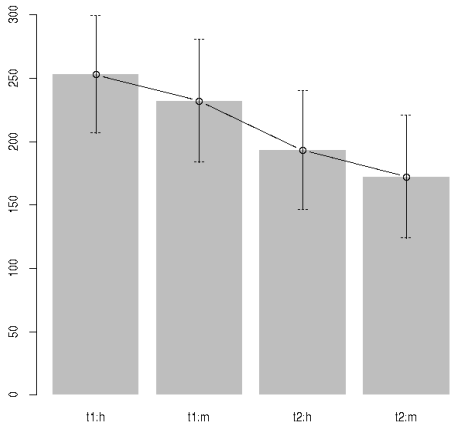
podemos hacer unas gráficas rápidas provistas también por agricolae, que nos muestra en forma de barras los valores de las medias de cada grupo con su respectivo significado estadístico.

> bar.group(lsdaovtablaR371$groups,ylim=c(0,300))



Otra representación preformada en agricolae se da a continuación

> bar.err(lsdaovtablaR371$means,variation="SD",ylim=c(0,300),bar=FALSE)



que nos muestra el valor de la media de cada grupo y el desvío estándar de los mismos grupos.

7.1.2. Anova a dos factores (con interacción)

Se entiende por interacción cuando el efecto de un factor sobre una variable puede ser condicionado por el valor de otro factor. Por ejemplo, como en el caso anterior vemos que t1 tiene mayores valores de insulina que t2. No habría interacción entre tratamiento (t1 y t2) con el sexo, si al aplicar t2 tenemos menores valores de insulina tanto en machos como en hembras, pero en magnitudes similares. En cambio, si el tratamiento t2 disminuye los valores de insulina, pero este descenso está condicionado en su magnitud según sea macho o hembra, estaríamos en presencia de una interacción entre los factores. Esto analizaremos ahora y es muy importante a la hora de las conclusiones.

Analizaremos los mismos datos de la tablaR361, pero evaluaremos si los valores de insulina obtenidos con cada tratamiento no son afectados de la misma manera en machos que en hembras. Es decir estamos analizando la interacción entre los factores.

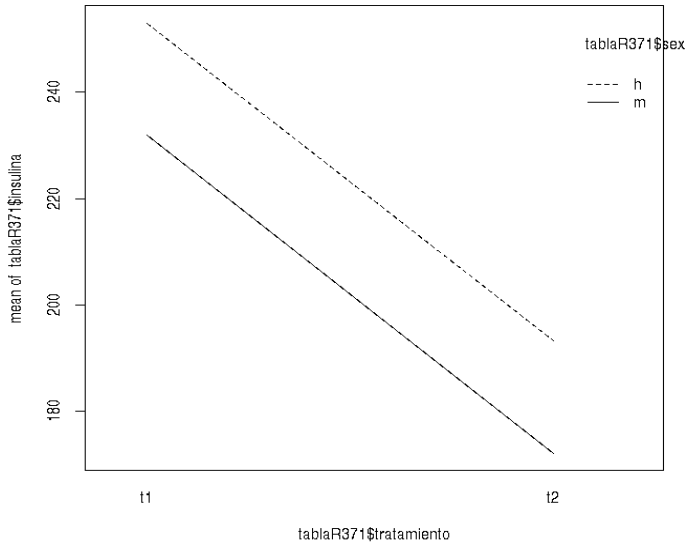
El código a aplicar es similar, con pequeños cambios. Si deseamos analizar la interacción en lugar de utilizar el signo "+" entre las variables utilizaremos el signo "*"

```
> aovtablaR371<-aov(insulina~tratamiento*sexo,data=tablaR371)

> summary(aovtablaR371)
                 Df  Sum Sq  Mean Sq  F value  Pr(>F)
tratamiento       1   35760   35760  15.818  0.000322 ***
sexo              1    4452    4452   1.969  0.169084
tratamiento:sexo  1       0       0   0.000  0.994730
Residuals        36   81385    2261
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

vemos que en la tabla anterior, además de las líneas tratamiento y sexo que aparecían cuando analizamos sin interacción, ahora aparece una línea extra que se llama tratamiento:sexo cuyo p-value es 0,994730. Esta línea es la que evalúa si el sexo interacciona con el tratamiento en lo que respecto a los valores de insulina. Como el valor de p-value es mayor de 0.05, concluimos que no hay interacción entre peso y tratamiento. En otras palabras el cambio en la insulina inducido por el tratamiento no discrepa si se trata de machos o hembras.

El paquete agricolae provee una función para hacer una gráfica de interacción, cuyo código se describe a continuación

```
> interaction.plot(tablaR371$tratamiento,tablaR371$sexo,tablaR371$insulina)
```



cuando las gráficas no se cruzan es signo de que no existe interacción. En caso contrario, si las líneas se cruzaran estaríamos en presencia de interacción entre los factores en estudio. En este caso vemos un caso clásico de no interacción entre las variables (sexo – tratamiento).

Como no hubo interacción es conveniente eliminarla del ensayo y hacerlo sin interacción, como lo realizamos al principio de este capítulo

```
> aovtablaR371<-aov(insulina~tratamiento+sexo,data=tablaR371)

> summary(aovtablaR371)
             Df Sum Sq  Mean Sq  F value    Pr(>F)
tratamiento  1  35760   35760      16.258    0.000265 ***
sexo         1   4452    4452      2.024     0.163202
Residuals   37  81385    2200
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Como los tratamientos tienen efecto sobre la insulina correspondería hacer el LSD.test, que ya fue hecho en el título anterior.

8. Clase 3.8

Video: https://youtu.be/um3Tsos3ocM
Tabla de datos:

8.1. Correlación de variables.

8.1.1. Test de correlación

A menudo necesitamos comprobar si una variable se correlaciona con otra, es decir si cuando una aumenta de valor la otra aumenta o disminuye de manera sistemática. Este tipo de estudio es importante a la hora de hallar relaciones entre variables sin que por ello esto implique alguna relación de causa consecuencia.

El análisis de correlación permite hallar una estadística que es el coeficiente de correlación (r). Este valor puede tomar valor que se hallan en el intervalo [-1,+1]. Cuanto más cercano sea a +1 indica una fuerte relación positiva entre las variables estudiadas, es decir cuando una aumenta, la otra también lo hace. Por su parte un valor de r cercano a -1 también muestra una gran correlación entre las variables, pero de manera inversa, cuando una crece, la otra decrece. Contrariamente, un valor cercano a 0, nos indica falta de relación entre las variables.

El coeficiente de correlación (r) elevado al cuadrado (r2) se conoce como coeficiente de determinación y nos indica que porcentaje de una variable es explicado por la variación de la otra. Por ejemplo si tenemos dos variables: peso corporal y cantidad de kilocalorías ingeridas y tenemos un valor de r=0.9 y un r2 = 0.81, podemos concluir (en el caso que la correlación fuera significativa desde el punto de vista estadístico) que al aumentar la cantidad de kilocalorías ingeridas tendremos pesos corporales mayores. Por otra parte podemos decir que un 81% del aumento de peso se puede explicar por las kilocalorías ingeridas. El otro 19% se explicará posiblemente por otras variables como pueden ser: actividad física, edad, sexo, etc.

Por otra parte en una correlación tendremos un valor de p-value, que será interpretado como en todos los casos. Un valor de p-value<0.05 nos indica que las variables están correlacionadas, mientras que un p-value>0.05 indicará falta de correlación. Por supuesto este valor de p-value puede modificarse de acuerdo a la correlación estudiada y el significado biológico que tiene hallar o no dicha correlación.

Veamos algunos ejemplos. Para esto introduzcamos en el espacio de trabajo la tablaR381 de la planila tablaR3-8

```
> tablaR381<-read.table("clipboard",header=T,sep="\t",dec=",",encoding="latin1")

> summary(tablaR381)
      hto             visc          proteínas
 Min.   :29.00   Min.   :3.000   Min.   :5.000
 1st Qu.:34.75   1st Qu.:3.175   1st Qu.:6.900
 Median :38.00   Median :3.400   Median :7.150
 Mean   :38.38   Mean   :3.475   Mean   :7.125
 3rd Qu.:41.25   3rd Qu.:3.725   3rd Qu.:7.825
 Max.   :49.00   Max.   :4.200   Max.   :8.100
```

Evaluamos los supuestos de normalidad con test de Shapiro, para cada una de las variables

```
> shapiro.test(tablaR381$hto)
        Shapiro-Wilk normality test
```

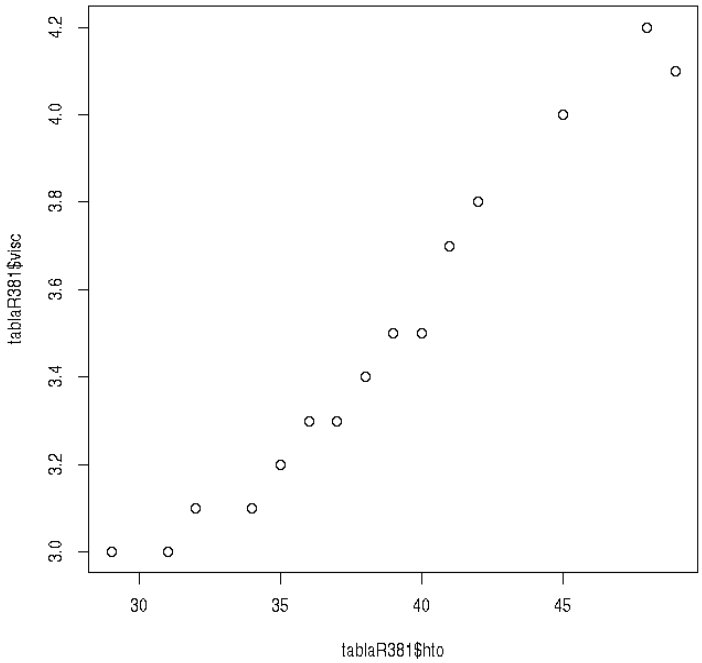
data: tablaR381$hto
W = 0.97339, p-value = 0.8901

> shapiro.test(tablaR381$visc)
	Shapiro-Wilk normality test
data: tablaR381$visc
W = 0.92413, p-value = 0.1965

> shapiro.test(tablaR381$prot)
	Shapiro-Wilk normality test
data: tablaR381$prot
W = 0.90451, p-value = 0.09495

Por los tres test anteriores aceptamos que las tres variables de la tabla tienen distribución normal. A continuación hacemos un gráfico de los datos: viscosidad en función del hto y otro de la viscosidad en función de proteínas.

> plot(tablaR381$hto,tablaR381$visc)



La gráfica nos sugiere una correlación positiva entre hto y visc.

Plantemos el análisis de correlación

> cor.test(tablaR381$vis,tablaR381$hto,alternative="greater",method="pearson",conf.level=0.95)
	Pearson's product-moment correlation
data: tablaR381$vis and tablaR381$hto
t = 18.892, df = 14, p-value = 1.163e-11
alternative hypothesis: true correlation is greater than 0
95 percent confidence interval:

0.9532137 1.0000000
sample estimates:
cor
0.9809458

El resultado no indica un alto significado estadístico de la correlación (p-value<0.05) y alta correlación entre las variables, r=0.98 (El valor del coeficiente de correlación (r) en salida del análisis lo encontramos como cor).

los argumentos:

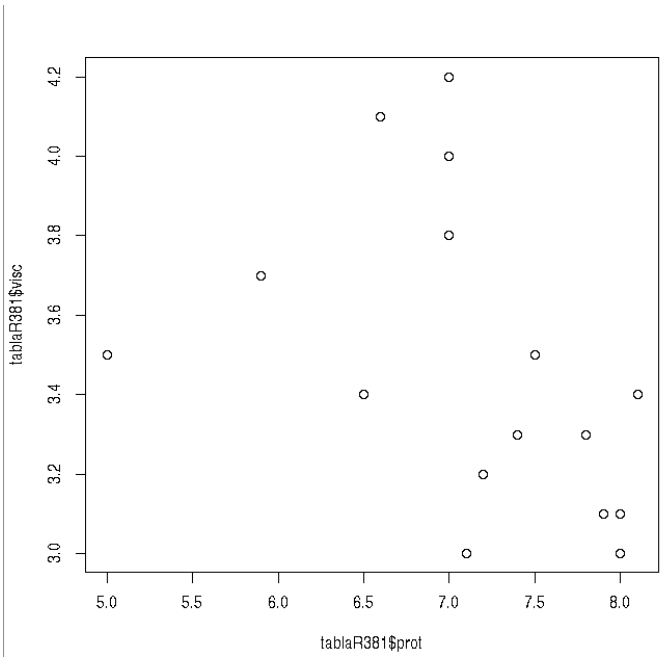
alternative: puede ser greater, si anticipamos una correlación positiva como en este caso, less si es negativa y two.sided si los datos no nos permiten inclinarnos por ninguna de las dos.

method: "pearson" cuando las variables tienen distribución normal, como en este caso (ver shapiro.test) y "spearman" cuando no tiene distribución normal.

conf.level: fija la probabilidad de error de tipo 1 o p-value que tomamos para considerar que la correlación es significativa o no. Además este valor nos permite obtener el intervalo de confianza del 95% que se observa en la salida del análisis: 0,9532137 -1. Que nos indica que el valor de r de nuestro análisis se halla en ese intervalo con una confianza del 95%.

Veamos el caso de las variables viscosidad y proteínas.

> plot(tablaR381$prot,tablaR381$visc)



La gráfica siguiente es menos convincente por lo tanto nos inclinaremos por alternative: two.sided., aunque bien podríamos utilizar "less" pero nunca "greater"

>
cor.test(tablaR381$vis,tablaR381$prot,alternative="two.sided",method="pearson",conf.level=0.95
)

```
        Pearson's product-moment correlation
 data:  tablaR381$vis and tablaR381$prot
t = -1.7824, df = 14, p-value = 0.09638
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.76308494  0.08343862
sample estimates:
      cor
-0.4300542
```

vemos que la correlación entre las variables es negativa, r=-0.43, sin embargo esta correlación no es significativa ya que p-value>0.05.

8.1.1.1. Potencia de una correlación

Con la función cor.test() evaluamos la correlación entre dos variables. Luego del análisis rechazamos o aceptamos la hipótesis nula con un cierto error de tipo I (p-value). Pero como en todos los ensayos es importante conocer la potencia del ensayo, es decir que probabilidad tenemos de haber aceptado la correlación entre las variables y que eso sea realmente cierto.

El paquete pwr que ya hemos utilizado tiene la función pwr.r.test() que permite calcular la potencia del ensayo. Cargamos la biblioteca

```
> library(pwr)
```

Este ensayo lleva como argumentos el número de pares de datos (n) que lo calcularemos utilizando una función que nos permite leer el número de filas de la tabla: nrow(tablaR381). Bien podríamos ir a la tabla contar las líneas y utilizar el número. Otro argumento es el coeficiente de correlación: r, que sacamos del ensayo realizado recientemente con la función cor.test(). El argumento sig.level lo fijamos en 0.05 y es el valor de p-value elegido para decidir si la correlación es o no significativa. El argumento power, lo dejamos con el valor NULL, para que lo calcule y será la potencia de nuestro ensayo.

Calculemos entonces la potencia del ensayo para la correlación entre vis y hto que había resultado ser significativa con r= 0.98. Si fijamos el nivel de significación en 0.05, resultará

```
> pwr.r.test(n=nrow(tablaR381),r=0.98,sig.level=0.05,power=NULL)
     approximate correlation power calculation (arctangh transformation)
              n = 16
              r = 0.98
      sig.level = 0.05
          power = 1
    alternative = two.sided
```

El resultado nos indica que la potencia de nuestro ensayo es de 100%. Es decir que nos inclinamos por aceptar la correlación y podemos estar casi seguros que ella existe.

En el caso de la correlación entre proteínas y viscosidad hallamos una correlación negativa, pero que concluimos que no era significativa ya que el valor del p-value fue 0.096. Sin embargo al ser tan cercano a 0.05 podríamos dudar de la decisión tomada de rechazar la correlación. Podríamos preguntarnos que número de datos necesitaríamos para que ese valor de r sea significativo con potencia de 0.8. Entonces hacemos el test de potencia, dejando libre el valor de n.

```
> pwr.r.test(n=NULL,r=-0.43,sig.level=0.05,power=0.8)
     approximate correlation power calculation (arctangh transformation)
```

n = 39.30827
r = 0.43
sig.level = 0.05
power = 0.8
alternative = two.sided

El test nos indica que necesitaríamos 40 datos y no 16 como tiene la tabla. Es decir que valdría la pena realizar un experimento con 40 datos y comprobar si es cierto. No quiere decir que deba darnos significativo, pero de mantenerse la tendencia, con 40 datos hallaríamos una correlación negativa y significativa.

Hallada correlación entre dos variables podemos hallar la regresión entre las variables, es decir hallar una función o modelo matemático que ajuste los valores y relacione los valores de una variable con la otra a través de fórmulas matemáticas. Esta regresión puede ser lineal, cuando la función de ajuste es una función lineal (representación gráfica: recta) o bien no lineal.

El proceso de hallar el modelo más adecuado para un conjunto de datos excede a los límites de este curso e implica la utilización de herramientas del modelado matemático.

8.1.2. Regresión lineal

Existen diferentes formas de hallar una regresión lineal en R. La misma puede desarrollarse obteniendo diferente información con las funciones
line()
lm()
lsfit()

8.1.2.1. regresión lineal con lm()

A los fines prácticos realizaremos el ajuste con la función lm() que utilizaremos luego en el módulo siguiente con otros fines. Además lm() provee la información necesaria y de importancia para este análisis.

Como hemos visto en el análisis de correlación de datos de la tablaR381, las variables visc y hto está positivamente correlacionadas y podemos interpretar que una recta sería un buen ajuste para los puntos. Esta conclusión se deduce del valor de r elevado y con alto significado estadístico (p-value<0.05) pero por sobre todo de la gráfica obtenida.

Recordemos que una recta tiene una ecuación matemática del tipo

y = a*x + h

donde y es la variable dependiente, en nuestro caso tomaremos a visc

x, la variable independiente. En nuestro caso será hto

a: es la pendiente que da la inclinación de la recta. Deberá ser positiva en este caso.

h: la ordenada al origen, es decir el valor de la variable ***y*** para un valor 0 de la variable hto. En otras palabras es el valor en que la recta corta al eje vertical (de la variable visc).

Aplicamos lm() y su resultado lo asignamos a un objeto. En esta función la variable dependiente (visc) se expresa en función de la independiente (hto) y dicha relación se indica con el símbolo ~

> lmvisc_hto<-lm(visc~hto,data=tablaR381)
pedimos un summary() del objeto que almacena los datos del análisis de regresión
> summary(lmvisc_hto)
Call:
lm(formula = visc ~ hto, data = tablaR381)

Residuals:

Min	1Q	Median	3Q	Max
-0.08658	-0.05824	-0.01732	0.06047	0.14304

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.94517	0.13531	6.985	6.39e-06 ***
hto	0.06592	0.00349	18.892	2.33e-11 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.07754 on 14 degrees of freedom
Multiple R-squared: 0.9623, Adjusted R-squared: 0.9596
F-statistic: 356.9 on 1 and 14 DF, p-value: 2.325e-11

En la salida del programa debemos centrar la atención en ciertos valores que hemos resaltado en amarillo. Veamos los valores de la tabla anterior, analizando las diferentes líneas

(Intercept) 0.94517 0.13531 6.985 6.39e-06 ***
Nos indica que la ordenada al origen (intercept), es significativamente diferente de cero, con un p-value (Pr(>|t|) <0.05 y que su valor es 0.94517. Es decir que nuestro modelo nos indica que cuando el hematocrito sea cero, la viscosidad tendrá un valor de 0.94517.
Pasemos a la otra línea
hto 0.06592 0.00349 18.892 2.33e-11 ***

Nos indica que la pendiente (hto en la tabla) es significativamente diferente de cero y tiene un valor de 0.06592. Es decir que la viscosidad aumenta 0.06592 unidades por cada una unidad que aumenta el hematocrito (hto)
La línea siguiente de la tabla nos muestra
Multiple R-squared: 0.9623

es el valor de r^2, que coincide con el valor de r hallado con cor.test. r= 0.9809458, pero elevado al cuadrado. Si este valor lo elevamos al cuadrado, obtendremos el valor de Multiple R-squared.

> 0.9809458 ^2

[1] 0.9622547

redondeado a cuatro decimales

[1] 0.9623

por último la línea

p-value: 2.325e-11

nos indica que el ajuste de la recta a los puntos es altamente significativo. En otras palabras haber elegido una regresión lineal para relacionar las variables visc y hto, ha sido acertada..

Es decir entonces que la función o modelo que relaciona visc con hto quedará conformada como

visc = 0.06592 * hto + 0.94517

Veamos si la recta ajusta bien los puntos. Para esto utilizaremos conceptos obtenidos en clases anteriores.

1- Creación de vectores con secuencias numéricas.

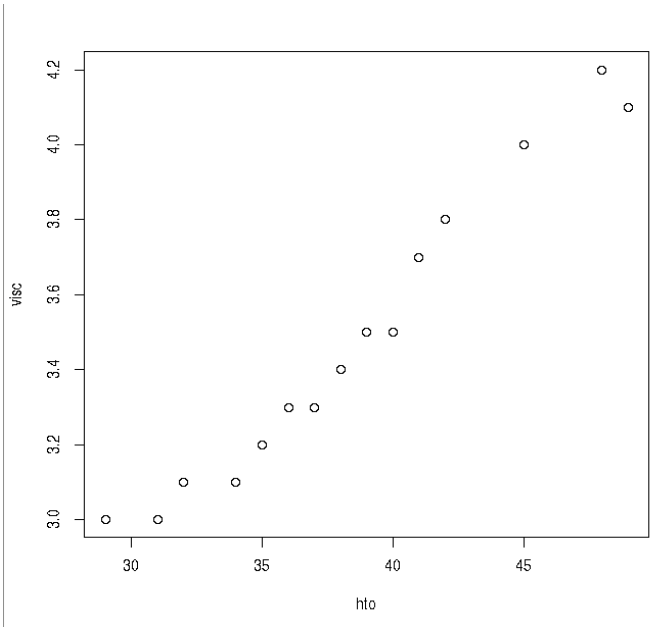
2- Formulación de funciones

3- Plotlines()

4- Spline()

grafiquemos primero los datos de visc y hto

> plot(tablaR381$hto,tablaR381$visc)



construimos nuestra función lineal, utilizando la codificación de R. Llamamos visc a la función y hto a la variable independiente, es decir utilizamos los mismos símbolos de nuestra tabla y colocamos los valores de los parámetros hallados (ordenada al origen y pendiente)

> visc<-function(hto){0.06592 * hto + 0.94517}

creamos un vector de valores de hto que va de 30 a 50 (rango de nuestro gráfico) en saltos de a 0.1

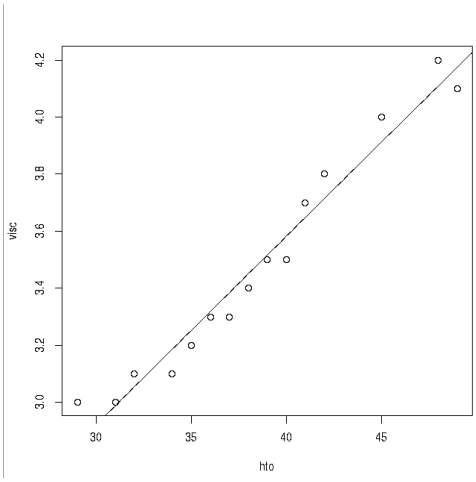
> htofit<-seq(30,50,0.1)

con estos valores y utilizando la función "visc", obtenemos los datos de visc con los valores de htofit

> viscfit<-visc(htofit)

sobre la gráfica anterior colocamos los puntos de hto (htofit) y visc (viscfit) construidos con nuestra función.

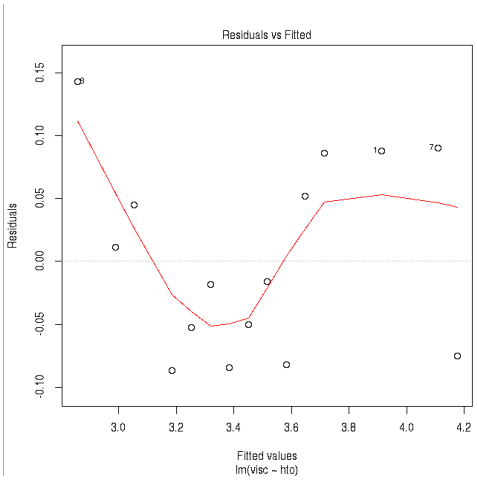
> lines(spline(htofit,viscfit))



la gráfica nos confirma que los parámetros de la función hallada ha sido correcta y coincidente con los valores de r obtenidos.

Ahora veremos algunas gráficas que nos permiten verificar si la regresión lineal fue la elección adecuada.

> plot(lmvisc_hto,which=1)



Esta gráfica nos muestra los residuales, es decir las diferencias entre el valor de nuestra función y los valores experimentales. Si bien la recta a simple vista ajustaba muy bien los valores experimentales, esta gráfica nos indica que hay algo de desviación. Si observamos la gráfica de los desvíos vemos que hasta el valor 3 de la abscisa los valores de los residuos son todos positivos, entre 3 y 3.6 son todos negativos y luego vuelven a ser positivos. Una buena regresión lineal tendría que tener residuos positivos y negativos alternantes. De todas maneras convengamos que sigue siendo un buen ajuste, pero la suma de otras herramientas de la modelización matemática podría permitirnos obtener un modelo que represente mejor los datos experimentales.

8.1.2.2. Regresión lineal con lm con ordenada al origen 0

Cuando buscamos la ecuación de una recta, es decir hacemos una regresión lineal, como por ejemplo lo descripto para la función lm(), obtenemos habitualmente una ecuación que tiene dos

parámetros: pendiente y ordenada al origen.

En algunos casos en particular, la función que nos interesa debe tener ordenada al origen = 0, como ocurre en las curvas de calibración utilizadas en numerosas determinaciones bioquímicas.

En esta situación, el procedimiento es similar al descripto en el apartado anterior.

Introduzcamos para este ejemplo los datos de la tablaR382

```
> tablaR382<-read.table("clipboard",header=T,sep="\t",dec=",",encoding="latin1")

> tablaR382
  concentracion absorbancia
1             0           0
2            10          71
3            20         150
4            30         200
5            40         298
6            50         355
```

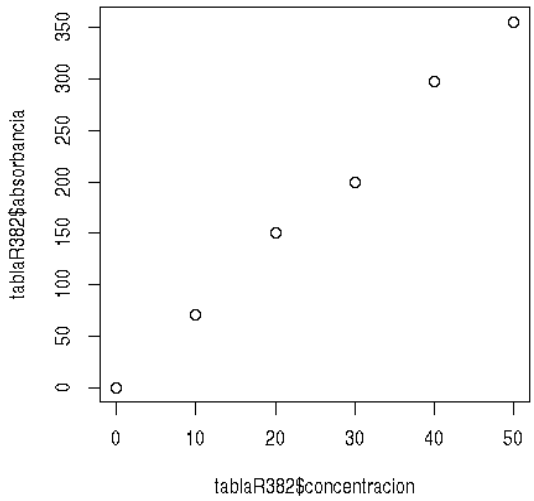
probamos normalidad de nuestras variables

```
> shapiro.test(tablaR382$concentracion)
        Shapiro-Wilk normality test
data:  tablaR382$concentracion
W = 0.98189, p-value = 0.9606

> shapiro.test(tablaR382$absorbancia)
        Shapiro-Wilk normality test
data:  tablaR382$absorbancia
W = 0.97473, p-value = 0.9226
```

en ambos casos no podemos descartar la distribución normal.

Graficamos las variables

```
> plot(tablaR382$concentracion,tablaR382$absorbancia)
```



vemos claramente un correlación positiva y una posible regresión lineal entre las variables.

Entonces realicemos primero la regresión lineal con lm() como conocemos

```
> lmtablaR382<-lm(absorbancia~concentracion,tablaR382)
> summary(lmtablaR382)
Call:
lm(formula = absorbancia ~ concentracion, data = tablaR382)

Residuals:
          1           2           3           4           5           6
 3.136e-14 -6.000e-01  6.800e+00 -1.480e+01  1.160e+01 -3.000e+00

Coefficients:
                Estimate   Std. Error   t value    Pr(>|t|)
(Intercept)    -1.133e-14  7.320e+00      0.00           1
concentracion  7.160e+00   2.418e-01     29.61   7.74e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.11 on 4 degrees of freedom
Multiple R-squared:  0.9955,          Adjusted R-squared:  0.9943
F-statistic:   877 on 1 and 4 DF,  p-value: 7.742e-06
```

La tabla anterior nos indica que la regresión lineal ha sido bien elegida como modelo para ajustar nuestro datos (p-value: 7.742e-06). La pendiente de nuestra recta es significativamente diferente de cero Pr(>|t|= 7.74e-06) y la ordenada al origen no discrepa de cero: Pr(>|t|) = 1, aunque el valor no es cero, sino (Intercept) = -1.133e-14.

A la hora de construir nuestra función deberíamos incluir ese valor de ordenada al origen.

Pero en realidad, si el ensayo nos indicó que no discrepa de cero podemos hacer que la regresión lineal lo considere cero.

Para ello se introduce en el código la siguiente modificación, que eliminará la ordenada al origen.

```
> lmtablaR382<-lm(absorbancia~concentracion-1,tablaR382)

> summary(lmtablaR382)
Call:
lm(formula = absorbancia ~ concentracion - 1, data = tablaR382)

Residuals:
          1           2           3           4           5           6
 5.418e-14 -6.000e-01  6.800e+00 -1.480e+01  1.160e+01 -3.000e+00

Coefficients:
              Estimate   Std. Error   t value    Pr(>|t|)
concentracion     7.160        0.122    58.7       2.72e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Residual standard error: 9.047 on 5 degrees of freedom
Multiple R-squared: 0.9986, Adjusted R-squared: 0.9983
F-statistic: 3445 on 1 and 5 DF, p-value: 2.716e-08

que nos dice que la regresión lineal sigue siendo un muy buen modelo, con un alto grado de ajuste, R-squared= 0.9986 y un alto significado estadístico, p-value= 2.716e-08 . En la tabla vemos solo la pendiente con un valor de 7.160, la que es significativamente diferente de cero.

La pendiente significativamente diferente de cero indica que la absorbancia en este caso depende del valor de concentración, aumentando 7.160 por cada unidad de concentración.

Grafiquemos entonces puntos y función.

En este caso veremos otro método de hacer lo mismo que en el apartado anterior. Así que comparemos ambos métodos.

Para ambos métodos nos vendrá bien crear la función que representa nuestra recta. Llamamos rectatablaR382 a la variable dependiente y concentración a la independiente. Bien podríamos utilizar ***x*** e ***y*** si lo deseáramos.

```
> rectatablaR382<-function(concentracion){7.16*concentracion}
```

método 1

Estos son nuestros datos

```
> tablaR382
  concentracion absorbancia
1             0           0
2            10          71
3            20         150
4            30         200
5            40         298
6            50         355
```

creamos un vector de concentración con valores de 0 a 50 en escalones de 0,1

```
> concentracionfit<-seq(0,50,0.1)
```

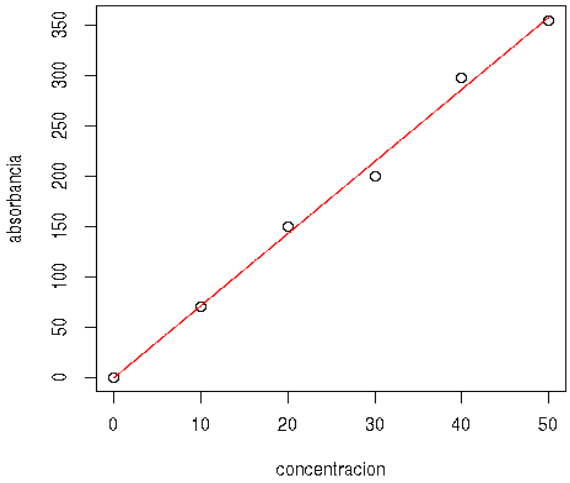
con este vector generamos otros de valores de absorbancia utilizando nuestra función

```
> absorbanciafit<-rectatablaR382(concentracionfit)
```

graficamos los datos y la línea de nuestra recta

```
> plot(absorbancia~concentracion,tablaR382)
```

```
> lines(spline(concentracionfit,absorbanciafit),col="red")
```



Vemos que el ajuste de la recta obtenida a los puntos experimentales es buena. Además los puntos van quedando a ambos lados de la recta, lo cual es un buen indicador que el modelo lineal utilizado es un buen ajuste de los puntos.

Método 2

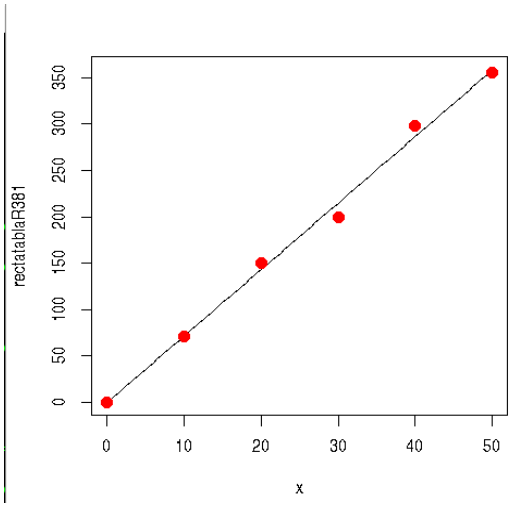
primero graficamos la función en un intervalo [0,50]

> plot(rectatablaR382,0,50)

luego graficamos los puntos

> points(tablaR382$concentracion,tablaR382$absorbancia,pch=19,col="red",cex=1.5)

obtenemos así una gráfica con la misma información.



8.1.3. Test de potencia para una regresión lineal

Otra vez estamos en el mismo punto. Aceptamos por los valores de p-value y R-squared que el modelo lineal es bueno. Sin embargo no sabemos que probabilidad hay de estar en lo cierto. Para saber ello debemos calcular la potencia.

El paquete pwr tiene una función que permite calcular la potencia para modelos lineales, es la función pwr.f2.test(). La regresión lineal es el modelo más sencillo.

Cargamos la biblioteca pwr

```
> library(pwr)
```

En general la función se escribe

```
pwr.f2.test(u = NULL, v = NULL, f2 = NULL, sig.level = NULL, power = NULL)
```

donde los argumentos son

u: grados de libertad del numerador

v: grados de libertad del denominador

f2: effect size

sig.level: probabilidad de error tipo I

power: potencia del ensayo.

Veamos nuevamente el primer caso realizado en esta clase, para entender los argumentos de la funcion pwr.f2.test()

Con los datos de la tablaR381 realizamos una regresión lineal

```
> lmvisc_hto<-lm(visc~hto,data=tablaR381)
pedimos un summary() del objeto generado
> summary(lmvisc_hto)
Call:
lm(formula = visc ~ hto, data = tablaR381)

Residuals:
     Min       1Q       Median       3Q      Max
-0.08658  -0.05824 -0.01732  0.06047  0.14304

Coefficients:
             Estimate   Std. Error    t value   Pr(>|t|)
(Intercept)  0.94517    0.13531       6.985     6.39e-06 ***
hto          0.06592    0.00349      18.892     2.33e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.07754 on 14 degrees of freedom
Multiple R-squared:  0.9623,          Adjusted R-squared:  0.9596
F-statistic: 356.9 on 1 and 14 DF,  p-value: 2.325e-11
```

y hallamos la tabla anova de la regresión

```
> anova(lmvisc_hto)
```

Analysis of Variance Table
Response: visc

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
hto	1	2.14583	2.14583	356.91	2.325e-11	***
Residuals	14	0.08417	0.00601			

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Analicemos la tabla anterior:

En la tabla tenemos dos renglones: hto y Residual, que llamaremos numerador y denominador, ya que el valor F value surge de dividir el Mean Sq del numerador por el del denominador.

La primer columna de la tabla muestra los valores de los grados de libertad (Df) de cada línea.

Df numerador

Df hto: = 1

El Df de una variable cuantitativa siempre es 1

Df denominador

Df Residuals= 14 = número de individuos - Df numerador -1 = 16 -1 -1

El effect size podemos estimarlo por la pendiente de nuestra recta.

hto 0.06592 0.00349 18.892 2.33e-11 ***

como podemos ver, nos está diciendo que la pendiente es significativamente diferente de cero, por lo que podemos poner un effect size =large. Es decir asumimos importante el efecto del hto sobre la viscosidad. Si no lo fuera, la pendiente no sería significativamente diferente de cero.

Entonces aplicamos una función de pwr y calculamos el valor del effect size con la función cohen.ES()

```
> cohen.ES(test = "f2", size = "large")

     Conventional effect size from Cohen (1982)

          test = f2

          size = large

    effect.size = 0.35
```

el valor de effect.size hallado es 0.35

aplicamos la función para calcular potencia de una regresión lineal

> pwr.f2.test(u = NULL, v = NULL, f2 = NULL, sig.level = NULL, power = NULL)

donde

u: grados de libertad del numerador

v: grados de libertad del denominador

f2: effect size

sig.level: probabilidad de error tipo I

power: potencia del ensayo.

si volvemos a ver la tabla para obtener los valores de los argumentos

```
> anova(lmvisc_hto)
Analysis of Variance Table
Response: visc
           Df   Sum Sq   Mean Sq   F value     Pr(>F)
hto         1  2.14583   2.14583    356.91  2.325e-11 ***
Residuals  14  0.08417   0.00601
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
resulta


> pwr.f2.test(u=1,v=14,f2=0.35,sig.level=0.05,power=NULL)
    Multiple regression power calculation
             u = 1
             v = 14
            f2 = 0.35
     sig.level = 0.05
       power = 0.5958113
```

Nos indica que aceptamos que los puntos son ajustados por una recta con un error de tipo I de 0.05 y por otro lado la certeza de que la recta es una buena opción de ajuste es del 59.5 %. Es decir rechazamos la hipótesis nula que dice que no hay una relación lineal entre las variables. La probabilidad de habernos equivocados a rechazar dicha hipótesis es menor al 5% sin embargo la probabilidad de estar en lo cierto al haber aceptado el modelo lineal es solo del 59.5%. Quizás con mayor número de datos hallemos una mejor probabilidad de estar en lo cierto, u otro modelo que no sea el lineal se constituya en una mejor opción.

9. Clase 3.9

Video: https://youtu.be/LDLQK8ttz0g
Tabla de datos:

9.1. Comparación de proporciones

Es habitual que tengamos que comparar proporciones entre dos o más grupos de individuos. R provee varias herramientas para cumplir con este objetivo.

9.1.1. Comparación de dos proporciones

Supongamos que queremos comparar si la proporción de exámenes aprobados entre alumnos a cargo de un profesor (p1) es igual que entre los alumnos del profesor 2 (p2). Para ello en el grupo de alumnos contaremos cuantos aprobados hay y luego compararemos las proporciones.

Introduzcamos los datos de la tabla

```
> tablaR393<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")

> tablaR393
  profesor totalalumnos aprobados
1       p1         150        95
2       p2         180        71
```

realicemos el test de proporciones

```
> prop.test(tablaR393$aprobados,tablaR393$totalalumnos,conf.level=0.95)
        2-sample test for equality of proportions with continuity correction
data:  tablaR393$aprobados out of tablaR393$totalalumnos
X-squared = 17.734, df = 1, p-value = 2.54e-05
alternative hypothesis: two.sided
95 percent confidence interval:
 0.1276839 0.3500939
sample estimates:
   prop 1    prop 2
0.6333333 0.3944444
```

El valor de p-value<0.05 nos indica que las proporciones son diferentes con un error de tipo I<0.05.
¿Cual es la potencia de nuestro ensayo?

9.1.1.1. Test de potencia para la comparación de dos proporciones

El paquete pwr propone dos funciones para el cálculo de potencia de un ensayo de comparación de dos proporciones, cuando el número de individuos de cada grupo es igual (n)

pwr.2p.test(h =NULL , n = NULL , sig.level = NULL, power = NULL)

En este caso la magnitud del efecto se calcula con la siguiente fórmula

$$h = 2 * asin\left(\sqrt{(p1)}\right) - 2 * asin\left(\sqrt{(p2)}\right)$$

Ecuación 9.1.

donde p1 y p2 son las proporciones de los grupos.

La función mencionada no puede ser utilizada en nuestro caso debido a que los números de datos de cada grupo son diferentes.

Si el número de datos es diferente la función a utilizar es

pwr.2p2n.test(h = NULL , n1 = NULL , n2 = NULL , sig.level = NULL , power = NULL)

Veamos la potencia de nuestro ensayo. p1 es la proporción de aprobados para el profesor 1 y p2 la proporción de aprobados para p2 que podemos obtener de la tablaR391

p1 = 95/150 = 0,633

p2= 71/180 = 0,394

definimos un objeto h, que corresponde a la magnitud del efecto

```
> h<-2*asin(sqrt(0.633333))-2*asin(sqrt(0.3944444))
> h
[1] 0.4826437
```

Cargamos la biblioteca pwr

```
> library(pwr)
```

ejecutamos la función

```
> pwr.2p2n.test(h=h,n1=150,n2=180,sig.level=0.05,power=NULL)

     difference of proportion power calculation for binomial distribution (arcsine transformation)
              h = 0.4826437
             n1 = 150
             n2 = 180
      sig.level = 0.05
          power = 0.9919295
    alternative = two.sided

NOTE: different sample sizes
```

Conclusión general. Con una probabilidad de error de tipo I<0.05 rechazamos que las proporciones son iguales. Por ende, nos quedamos con que son diferentes y afirmamos esto con una potencia de 0,9919 (un 99%).

9.1.2. Comparación de más de dos proporciones

El mismo test de igualdad de proporciones (prop.test) nos permite dar esta respuesta a este tipo de interrogantes. Introduzcamos los datos de la tablaR391 de la planilla de cálculo tablaR3-9.ods/xls. En esta tabla tenemos número de fumadores (fumadores) sobre el total de personas encuestadas (total) en 6 sitios diferentes de una ciudad.

```
> tablaR391<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")

> tablaR391
  total fumadores
1   100        51
2   105        43
3    95        81
```

4 189 120
5 154 90
6 210 151

Deseamos comprobar si las proporciones de fumadores en los diferentes grupos son iguales o diferentes.

```
> prop.test(tablaR391$fumadores,tablaR391$total,conf.level=0.95)
        6-sample test for equality of proportions without continuity correction
data:  tablaR391$fumadores out of tablaR391$total
X-squared = 56.698, df = 5, p-value = 5.836e-11
alternative hypothesis: two.sided
sample estimates:
   prop 1    prop 2    prop 3    prop 4    prop 5    prop 6
0.5100000 0.4095238 0.8526316 0.6349206 0.5844156 0.7190476
```

Conclusión: con un error de tipo I menor a 0.05 concluimos que las proporciones de fumadores entre los 6 grupos no es la misma, ya que p-value fue menor a 0.05.
Una vez verificada la diferencia de proporciones podemos comprobar cuales son los grupos que difieren. Para ello utilizamos la función pairwise.prop.test()

9.1.3. pairwise.prop.test

```
> pairwise.prop.test(tablaR391$fumadores,tablaR391$total,conf.level=0.95)
        Pairwise comparisons using Pairwise comparison of proportions
data:  tablaR391$fumadores out of tablaR391$total
          1       2       3       4       5
2 0.57779     -               -               -               -
3 9.1e-06   4.4e-09         -               -               -
4 0.26834   0.00314   0.00269         -               -
5 0.59969   0.06668   0.00021   0.59969 -
6 0.00445   2.8e-06   0.10188   0.36402   0.07103
P value adjustment method: holm
```

Interpretemos la tabla anterior. Por ejemplo en la intersección de la columna 2 y la fila 5 hallamos el valor de p-value de la comparación de la proporción 2 con la 5. Como el valor es superior a 0,05 decimos que no hay diferencias entre esas dos proporciones. Vemos los valores arrojados por el prop.test, realizado anteriormente

```
  prop 1    prop 2    prop 3    prop 4    prop 5    prop 6
0.5100000 0.4095238 0.8526316 0.6349206 0.5844156 0.7190476
```

pero si nos colocamos en la intersección de columna 2 y 6 hallamos un valor de p-value<0.05, indicando que dichas proporciones no son iguales. Si observamos la proporciones 2 y 6 podemos notar la diferencia entre ambas

```
  prop 1    prop 2    prop 3    prop 4    prop 5    prop 6
0.5100000 0.4095238 0.8526316 0.6349206 0.5844156 0.7190476
```

Así, analizando toda la tabla podemos decir que el grupo 1 difiere significativamente del 3 y 6, el 2 difier significactivamente del 3, 4 y 6. El grupo 3 difiere del 4 y 5.

En R no se dispone de un test de potencia para más de dos proporciones.

9.1.4. Pruebas de asociación

En muchas situaciones necesitamos conocer si existe asociación entre dos variables cualitativas. Por ejemplo tenemos ratas machos y hembras a las que hemos clasificadas según un criterio establecido en diabéticas o normales. Lo que deseamos conocer es si la probabilidad de ser diabético o no está asociado a ser macho o hembra. Dicho en otras palabras nos preguntamos si ser diabético o no está asociado al sexo.

Introduzcamos los datos de tablaR392.

```
> tablaR392<-read.table("clipboard",header=TRUE,dec=",",sep="\t",encoding="latin1")
```

```
> tablaR392
   unidadexperimental   sexo diabetico
1                   1  macho       si
2                   2  macho       si
3                   3  macho       no
4                   4  macho       si
5                   5  macho       si
6                   6  macho       si
7                   7  macho       si
8                   8  macho       no
9                   9  macho       si
10                 10  macho       no
11                 11  macho       si
12                 12  macho       no
13                 13 hembra       no
14                 14 hembra       no
15                 15 hembra       si
16                 16 hembra       no
17                 17 hembra       no
18                 18 hembra       si
19                 19 hembra       no
20                 20 hembra       no
21                 21 hembra       si
22                 22 hembra       no
23                 23 hembra       no
24                 24 hembra       no
```

la función table() nos permite contar los animales dentro de cada grupo por sexo y por la presencia de diabetes o no. La tabla siguiente nos indica por ejemplo que tenemos 8 machos diabéticos entre 12 machos y 3 hembras diabéticas entre 12 hembras.

```
> table(tablaR392$sexo,tablaR392$diabetico)
        no  si
 hembra  9   3
 macho   4   8
```

la última tabla permite mostrarnos que parecería existir una asociación en que las hembras tiende a no tener diabetes (solo 3 de 12) mientras que los machos parecen tener más tendencia a presentar diabetes (8 de 12). Para comprobar la asociación tenemos varios test: test de Fisher y Chisq.test

Si bien ambos test ponen a prueba las mismas hipótesis. La prueba de Fischer es recomendada cuando existen frecuencias teóricas menores a 5 o las sumas de las frecuencias marginales son muy diferentes. Contrariamente, cuando no hay frecuencias bajas o las frecuencias marginales son parecidas es conveniente el chisq.test() por tener mayor potencia.

Reproduzcamos nuevamente la tabla incluyendo las frecuencias marginales. Las frecuencias marginales serían el número de hembras y machos y por otro lado el número de diabéticos o no diabéticos.

	no	si	totales (freq marginal)
hembra	9	3	12
macho	4	8	12
totales (freq marginal)	13	11	24

Las frecuencias teóricas serían los valores que se obtendrían bajo la hipótesis que no hay asociación entre las variables. Si no hubiera asociación la cantidad de diabéticos en hembras y machos debería seguir la proporción hallada en el total de individuos, es decir 11 por cada 24 individuos. Así
24 individuos ..... 11 diabéticos
12 machos ..... x = 5,5

24 individuos ..... 11 diabéticos
12 hembras ..... x = 5,5

24 individuos ..... 13 normales
12 machos ..... x = 6,5

24 individuos ..... 13 normales
12 hembras ..... x = 6,5

pasando esto a una tabla de frecuencias teóricas

	no	si	totales
hembra	6,5	5,5	12
macho	6,5	5,5	12
totales	13	11	24

Vemos que ninguna frecuencia es menor que 5 y las marginales (totales) son muy parecidos. Sería recomendable aplicar el Chisq.test. Pero aplicaremos los dos

9.1.4.1. Test exacto de Fisher

```
> fisher.test(table(tablaR392$sexo,tablaR392$diabetico))

        Fisher's Exact Test for Count Data

data:  table(tablaR392$sexo, tablaR392$diabetico)
p-value = 0.09953
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
  0.7824285 51.9088650
sample estimates:
odds ratio
  5.512388
```

Concluimos que no existe asociación entre la presencia o no diabetes y el sexo de las ratas estudiadas con alfa=0.05, debido a que p-value>0.05.

Aplicamos a los mismos datos la funcion chisq.test()

```
> chisq.test(table(tablaR392$diabetico,tablaR392$sexo))
        Pearson's Chi-squared test with Yates' continuity correction
data:  table(tablaR392$diabetico, tablaR392$sexo)
X-squared = 2.6853, df = 1, p-value = 0.1013
```

Concluimos que no existe asociación entre la presencia o no diabetes y el sexo de las ratas estudiadas con alfa=0.05

Si hubiera asociación entre las variables y una de ellas, como en este caso es un evento negativo (diabetes si), se puede calcular el riesgo y el riesgo relativo de padecer diabetes entre machos y hembras.

9.1.4.2. Riesgo

En estos casos se puede calcular el riesgo absoluto y relativo

El riesgo absoluto de padecer diabetes entre los machos, es el cociente entre el número de machos con diabetes y el número total de machos.

Rm= 8/12 = 0.66

Es decir que el riesgo de padecer diabetes entre los machos es del 66%.

Para la hembras, el riesgo absoluto es el cociente entre el número de hembras con diabetes y el número total de hembras.

Rh= 3/12 = 0.25.

Es decir que entre las hembras existe un riesgo del 25% de padecer diabetes.

El riesgo relativo es el cociente entre ambos riesgos

RR= Rm/Rh = 0.66/0.25 = 2.64

Es decir que existe 2.64 veces más de riesgo de padecer diabetes entre los machos comparados con las hembras.

9.2. Test de potencia para tablas de contingencia

El paquete pwr provee una función que nos permite calcular la potencia de un ensayo de asociación entre dos variables o bien calcular el número de datos necesarios para demostrar asociación entre dos variables con una dada magnitud del efecto y un nivel de significación establecido.
Para ello hagamos el siguiente razonamiento. En primer lugar tomemos las frecuencias reales, que las obtenemos de la tabla con la que venimos trabajando

	no	si	totales
hembra	9	3	12
macho	4	8	12
totales	13	11	24

vemos ahora las frecuencias bajo el supuesto de no existe asociación que ya calculamos

	no	si	totales
hembra	6,5	5,5	12
macho	6,5	5,5	12
totales	13	11	24

En base a estas tablas calculamos las frecuencias relativas reales y teóricas. Las frecuencias relativas reales, se calculan dividiendo cada frecuencia real por el número total de individuos

	no	si	totales
hembra	9/24=0.375	3/24= 0.125	12/24= 0.5
macho	4/24 =0.166	8/24= 0.333	12/24= 0.5
totales	13/24 =0.542	11/24= 0.458	24/24 =1

de la misma manera calculamos las frecuencias relativas teóricas

	no	si	totales
hembra	6.5/24=0.271	5.5/24= 0.229	12/24
macho	6.5/24= 0.271	5.5/24= 0.229	12/24
totales	13/24= 0.542	11/24= 0.458	24/24

Para el ensayo de potencia se necesitan los grados de libertad (df), los que se calculan con el número de filas y columnas de la tabla, sin contar las frecuencias marginales

df= (nº filas – 1) * (nº columnas -1)

df = (2-1)*(2-1)= 1

construimos un vector con las frecuencias relativas reales, siguiendo una dado orden

p1<-c(0.375,0.125,0.166,0.333)

y otro con las frecuencias teóricas, en el mismo orden

p0<-c(0.271,0.229,0.271,0.229)

se calcula la magnitud del efecto con ambos vectores. Es obvio que la asociación será mayor cuanto más diferentes sean los valores de p1 y p0. Para ello se aplica la función ES.w1()

> w<-ES.w1(p0,p1)

la magnitud del efecto resulta

w

[1] 0.4183981

Como nuestro ensayo de asociación nos indicó que no existía asociación significativa entre el sexo y la presencia de diabetes, pero el p-value fue cercano a 0.05, decidimos calcular con qué número de datos (N) sería necesario para demostrar asociación entre las variables

```
> pwr.chisq.test(w=w,N=NULL,df=1,sig.level=0.05,power=0.8)
    Chi squared power calculation
              w = 0.4183981
              N = 44.83601
             df = 1
      sig.level = 0.05
          power = 0.8
NOTE: N is the number of observations
```

Si en lugar de tener 24 individuos en nuestro estudio tuviéramos 45, podríamos llegar a demostrar que existe asociación con p-value de 0.05 y pot= 0.8.