

Entrega 4 y FINAL. Problemas y Talleres MATIII Estadística grado informática 2019-2020

Ricardo Alberich

13-05-2020

Contenidos

1 Entrega 4 Problemas: Estadística Inferencial 2	1
1.1 Problema 1: Contraste de proporciones de dos muestras independientes.	1
1.2 Problema 2 : Contraste de proporciones de dos muestras emparejadas.	2
1.3 Problema 3 : ANOVA comparación media puntuaciones según fabricante.	2
1.4 Problema 4: Distribución de los grados de un grafo de contactos.	2

1 Entrega 4 Problemas: Estadística Inferencial 2

Contestad en GRUPOS del proyecto a los siguientes problemas y cuestiones en un fichero Rmd y su salida en html o pdf.

Cambien podéis incluir capturas de problemas hechos en papel. Cada pregunta vale lo mismo y se reparte la nota entre sus apartados.

1.1 Problema 1: Contraste de proporciones de dos muestras independientes.

Queremos comparar las proporciones de aciertos de dos redes neuronales que detectan tipos si una foto con un móvil de una avispa es una [avispa velutina o asiática] (https://es.wikipedia.org/wiki/Vespa_velutina). Esta avispa es una especie invasora y peligrosa por el veneno de su picadura. Para ello disponemos de una muestra de 1000 imágenes de insectos etiquetadas como avispa velutina y no velutina.

[Aquí tenéis el acceso a los datos](#). Cada uno está en fichero los aciertos están codificados con 1 y los fallos con 0.

Se pide:

1. Cargad los datos desde el servidor y calcular el tamaño de las muestras y la proporción de aciertos de cada muestra. 2. Contrastad si hay evidencia de que las proporciones de aciertos del algoritmo 1 son mayores que las del algoritmo 2. Definid bien las hipótesis y las condiciones del contraste. Tenéis que hacer el contraste de forma manual y con funciones de **R** y resolver el contraste con el p -valor. (**vale doble**) 3. Calculad e interpretad los intervalos de confianza para la diferencia de proporciones asociados al test anterior.

1.2 Problema 2 : Contraste de proporciones de dos muestras emparejadas.

En el problema anterior hemos decidido quedarnos con el mejor de los algoritmos y mejorarlo. Pasamos las mismas 1000 imágenes a la version_beta del algoritmo y a la version_alpha. [Aquí tenéis el acceso a los datos en el mismo orden para las 1000 imágenes](#). Cada uno está en fichero los aciertos están codificados con 1 y los fallos con 0.

1. Cargad los datos desde el servidor y calcular el tamaño de las muestras y la proporción de aciertos de cada muestra. 2. Contrastad si hay evidencia de que las proporciones de aciertos del algoritmo 1 son mayores que las del algoritmo 2. Definid bien las hipótesis y las condiciones del contraste. Tenéis que hacer el contraste de forma manual y con funciones de R y resolver el contraste con el p -valor. (**vale doble**) 3. Calculad e interpretad los intervalos de confianza para la diferencia de proporciones asociados al test anterior.

1.3 Problema 3 : ANOVA comparación media puntuaciones según fabricante.

Una vez mejorado nuestro algoritmo queremos saber su comportamiento bajo distintos tipos de móviles.

Seleccionamos 6 móviles de la misma gama de calidad de 6 fabricantes distintos. A los fabricantes los denotamos por F1, F2, F3, F4, F5 y F6.

Vamos a jugar no con la clasificación sino con el score que produce el algoritmo. Para ello seleccionamos 4 muestra aleatorias de fotos de insectos enviadas por los usuarios y la puntuación (*score*) que nos da el algoritmo que es una variable aleatoria continua de con rango de 0 a 100.

La idea es comprobar si la media de las puntuaciones del algoritmo es la misma para cada uno de los fabricantes.

Los datos los podéis descargar de esta dirección del [servidor bioinfo.uib.es](http://servidor.bioinfo.uib.es).

Antes de descargarlo, visualizar el fichero desde el navegador, para saber cómo descargarlo.

¿Qué contrastes realiza la función 1. ¿Podemos asegurar que las muestras son normales en cada grupo? ¿y son homocedásticas? Sea cual sea la respuesta justificad que parte del código la confirma. 2. Escribid formalmente la hipótesis nula y la alternativa. Calcular la tabla de ANOVA y resuelve el test de forma manual. 3. Calcular la tabla de ANOVA y resuelve el test con la función `aov` de R. 4. Haced una comparación de pares con la función adecuada de R para la corrección del holm al nivel de significación $\alpha = 0.1$. Interpretad el resultado. 5. Comparar por grupos con el test de Duncan del paquete `agricolae`. Interpretad el resultado.

1.4 Problema 4: Distribución de los grados de un grafo de contactos.

En el artículo de A. Broder et al., [Graph structure in the Web](#). *Computer Networks* 33, 309 (2000).

Se recopiló el número de enlaces a sitios web encontrados en un rastreo web de 1997 de aproximadamente 200 millones de páginas web,

Con el se construyó una [tabla](#) con la frecuencia de sitios por número de enlaces. El código siguiente carga del enlace que han puesto los autores del artículo

```
data_links=read.table("http://tuvalu.santafe.edu/~aaronc/powerlaws/data/weblinks.hist",header=TRUE)
head(data_links)
```

```
## degree frequency
## 1      0 35159835
## 2      1 106649769
## 3      2  40711748
## 4      3  22648832
## 5      4  12617832
## 6      5   8188854
```

```
str(data_links)
```

```
## 'data.frame': 14480 obs. of 2 variables:
## $ degree : int 0 1 2 3 4 5 6 7 8 9 ...
## $ frequency: int 35159835 106649769 40711748 22648832 12617832 8188854 6438634 4690068 4954649 373
```

```
# eliminamos la páginas con menos de 8 enlaces y las de más de 1000 enlaces
data_links_central=data_links[data_links$degree>8&data_links$degree<10^3,]
head(data_links_central)
```

```
## degree frequency
## 10 9 3731928
## 11 10 3036333
## 12 11 2496648
## 13 12 2119312
## 14 13 1790068
## 15 14 1546579
```

```
tail(data_links_central)
```

```
## degree frequency
## 995 994 213
## 996 995 193
## 997 996 157
## 998 997 137
## 999 998 178
## 1000 999 153
```

El siguiente código calcula las regresiones exponencial, potencial y lineal (en algún orden) de las frecuencias (frequency) contra los enlaces (degree).

```
sol1=lm(frequency~ degree,data=data_links_central)
summary(sol1)
```

```
##
## Call:
## lm(formula = frequency ~ degree, data = data_links_central)
##
## Residuals:
## Min 1Q Median 3Q Max
## -96861 -69548 -25033 22374 3598744
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 134974.49 13778.17 9.796 <0.0000000000000002 ***
## degree -198.98 23.77 -8.369 <0.0000000000000002 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 214100 on 989 degrees of freedom
## Multiple R-squared: 0.06614, Adjusted R-squared: 0.06519
## F-statistic: 70.04 on 1 and 989 DF, p-value: < 0.00000000000000022
```

```
sol2=lm(log10(frequency)~ degree,data=data_links_central)
summary(sol2)
```

```
##
## Call:
## lm(formula = log10(frequency) ~ degree, data = data_links_central)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43758 -0.26558 -0.07671  0.16681  2.13097
##
## Coefficients:
##              Estimate Std. Error t value      Pr(>|t|)
## (Intercept)  4.46504979  0.02381018  187.53 <0.0000000000000002 ***
## degree      -0.00267658  0.00004109  -65.15 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.37 on 989 degrees of freedom
## Multiple R-squared:  0.811, Adjusted R-squared:  0.8108
## F-statistic: 4244 on 1 and 989 DF, p-value: < 0.00000000000000022
```

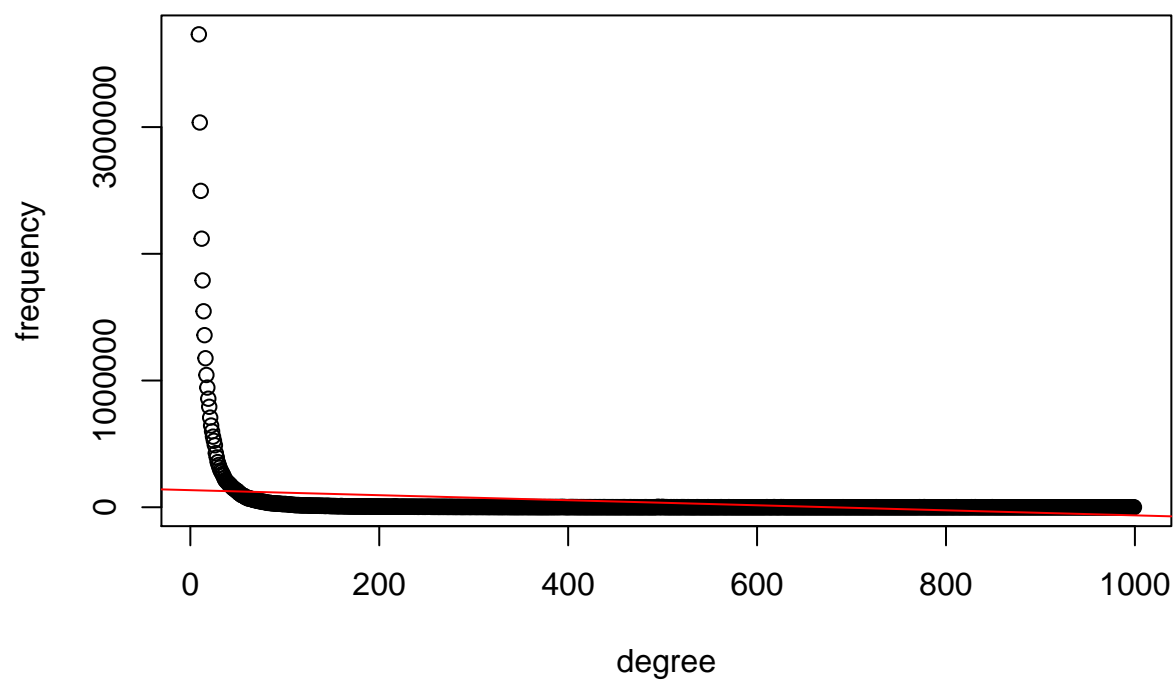
```
sol3=lm(log10(frequency)~ log10(degree),data=data_links_central)
summary(sol3)
```

```
##
## Call:
## lm(formula = log10(frequency) ~ log10(degree), data = data_links_central)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.21376 -0.04747 -0.01555  0.01958  0.73976
##
## Coefficients:
##              Estimate Std. Error t value      Pr(>|t|)
## (Intercept)  8.722036  0.020623  422.9 <0.0000000000000002 ***
## log10(degree) -2.170129  0.007894 -274.9 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09674 on 989 degrees of freedom
## Multiple R-squared:  0.9871, Adjusted R-squared:  0.9871
## F-statistic: 7.557e+04 on 1 and 989 DF, p-value: < 0.00000000000000022
```

Ahora dibujamos los gráficos adecuados a cada modelo

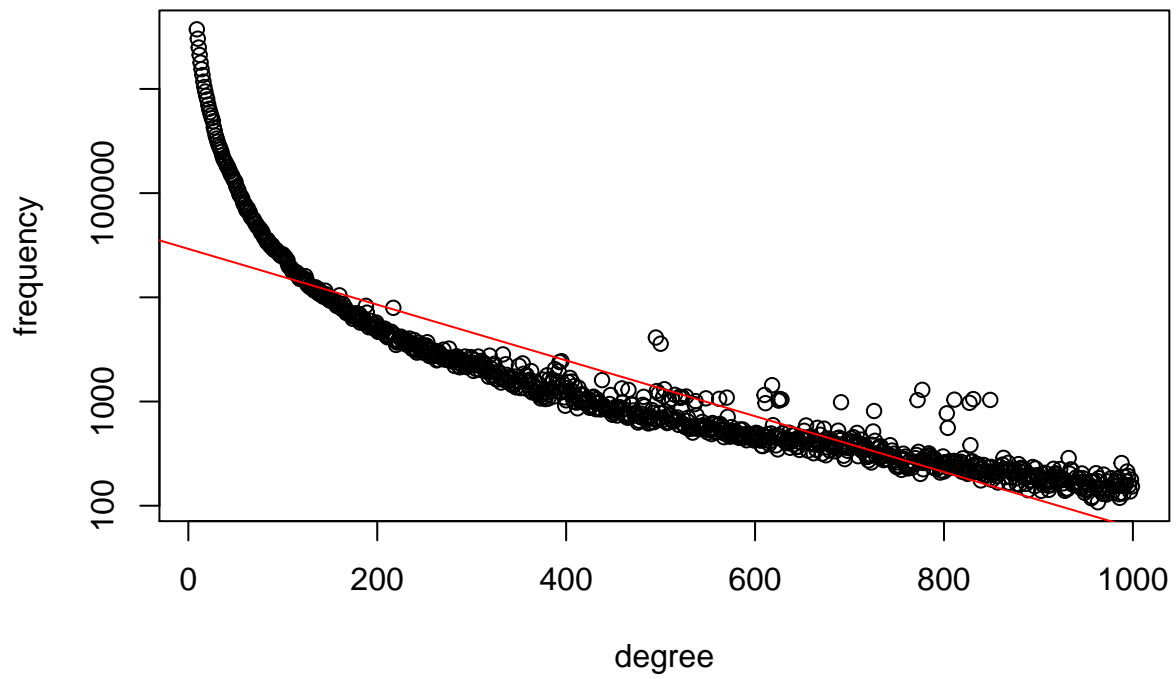
```
plot(data_links_central,main="Modelo .....")
abline(sol1,col="red")
```

Modelo



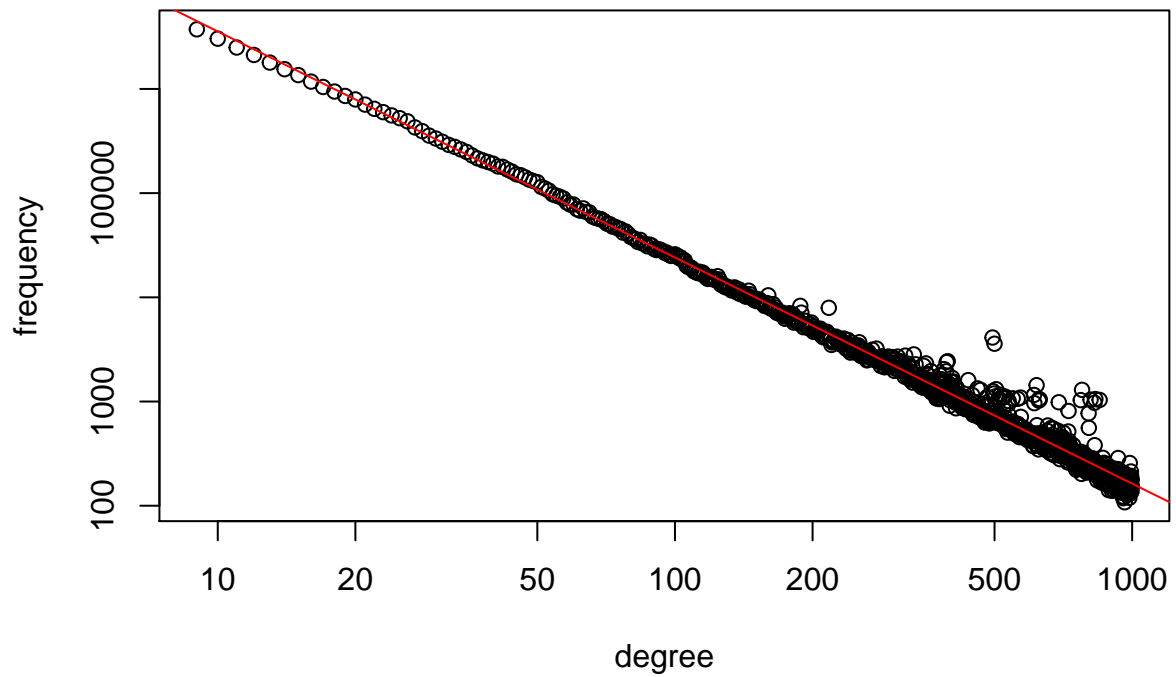
```
plot(data_links_central,main="Modelo .....",log="y")
abline(sol2,col="red")
```

Modelo



```
plot(data_links_central,main="Modelo .....",log="xy")  
abline(sol3,col="red")
```

Modelo



Se pide:

1. Explicad el modelo de regresión que calcula cada función `lm`
2. ¿Qué modelo y en función de qué parámetros es el mejor?
3. Para el mejor modelo calcular los coeficientes en las unidades originales y escribir la ecuación del modelo (1.5)