

Taller de R. Grados de los vértices de un grafo

Pregunta 1 (1 punto)

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  <2e-16 ***
## degree      -198.98     23.77  -8.369  <2e-16 ***
## ---
## 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: < 2.2e-16
```

```
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)  4.465e+00  2.381e-02  187.53  <2e-16 ***
## degree      -2.677e-03  4.109e-05  -65.15  <2e-16 ***
## ---
## 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: < 2.2e-16
```

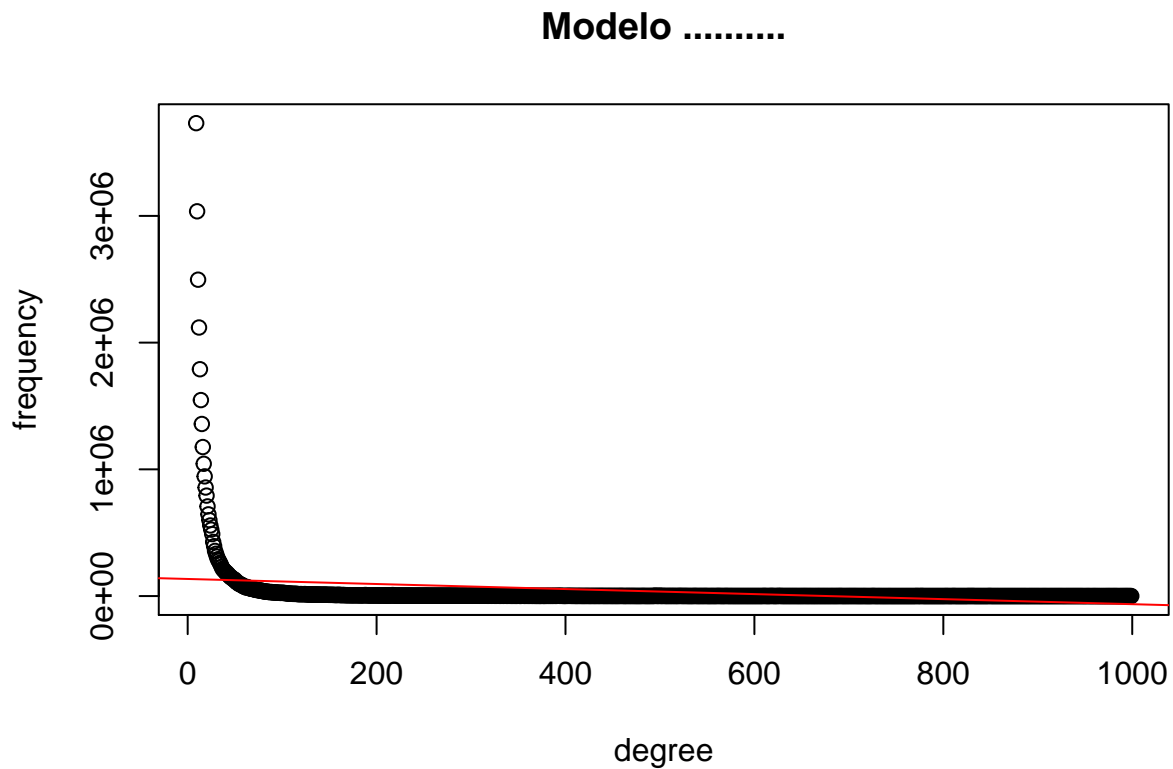
```
sol3=lm(log10(frequency)~ log10(degree),data=data_links_central)
summary(sol3)
```

```
##
## 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  <2e-16 ***
## log10(degree) -2.170129   0.007894  -274.9  <2e-16 ***
## ---
## 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: < 2.2e-16
```

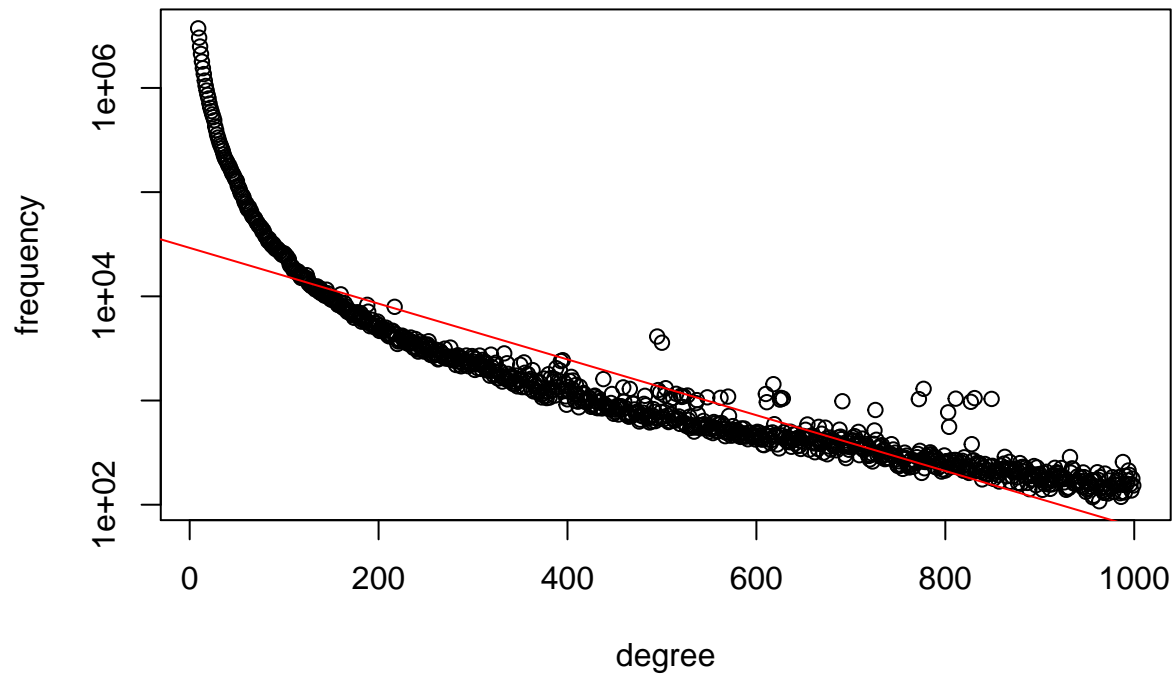
Ahora dibujamos los gráficos adecuados a cada modelo

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



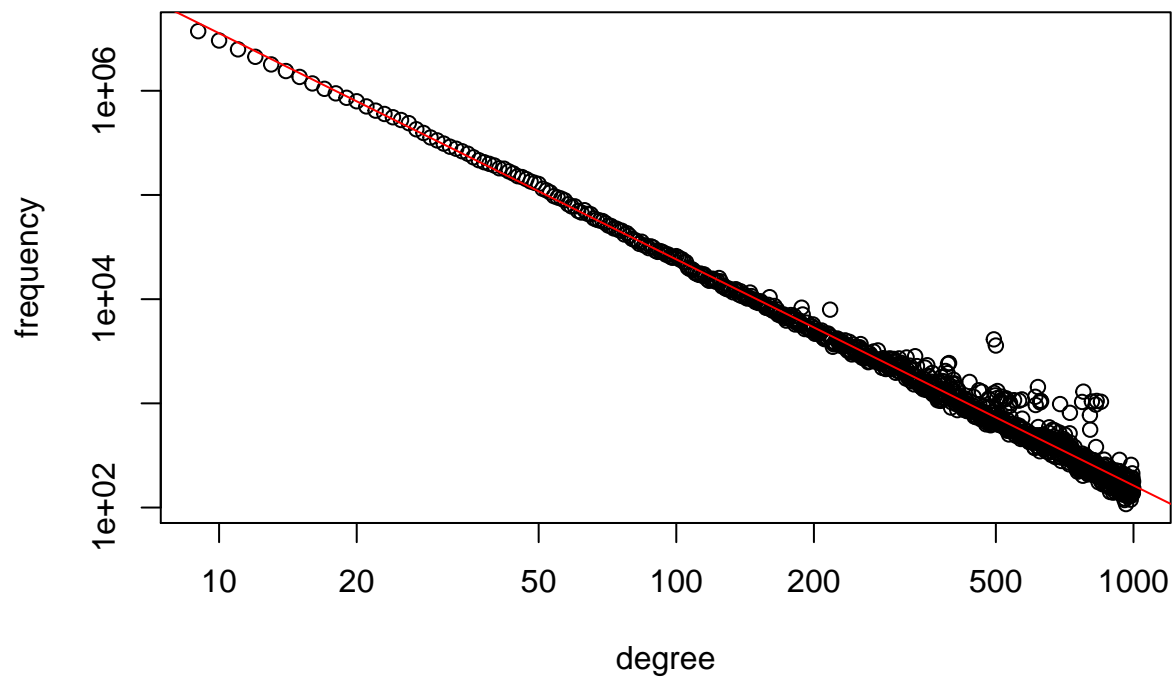
```
plot(data_links_central,main="Modelo .....",log="y")
abline(sol2,col="red")
```

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` (1)

- 2.) ¿Qué modelo y en función de qué parámetros es el mejor? (1.5)
- 3.) Para el mejor modelo calcular los coeficientes en las unidades originales y escribir la ecuación del modelos (1.5)