tp4

Trabajo práctico 4

Cargamos los paquetes necesarios.

```
library(MASS)
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 3.5.3
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

```
## Warning: package 'foreach' was built under R version 3.5.3
```

```
## Loaded glmnet 2.0-18
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.3
```

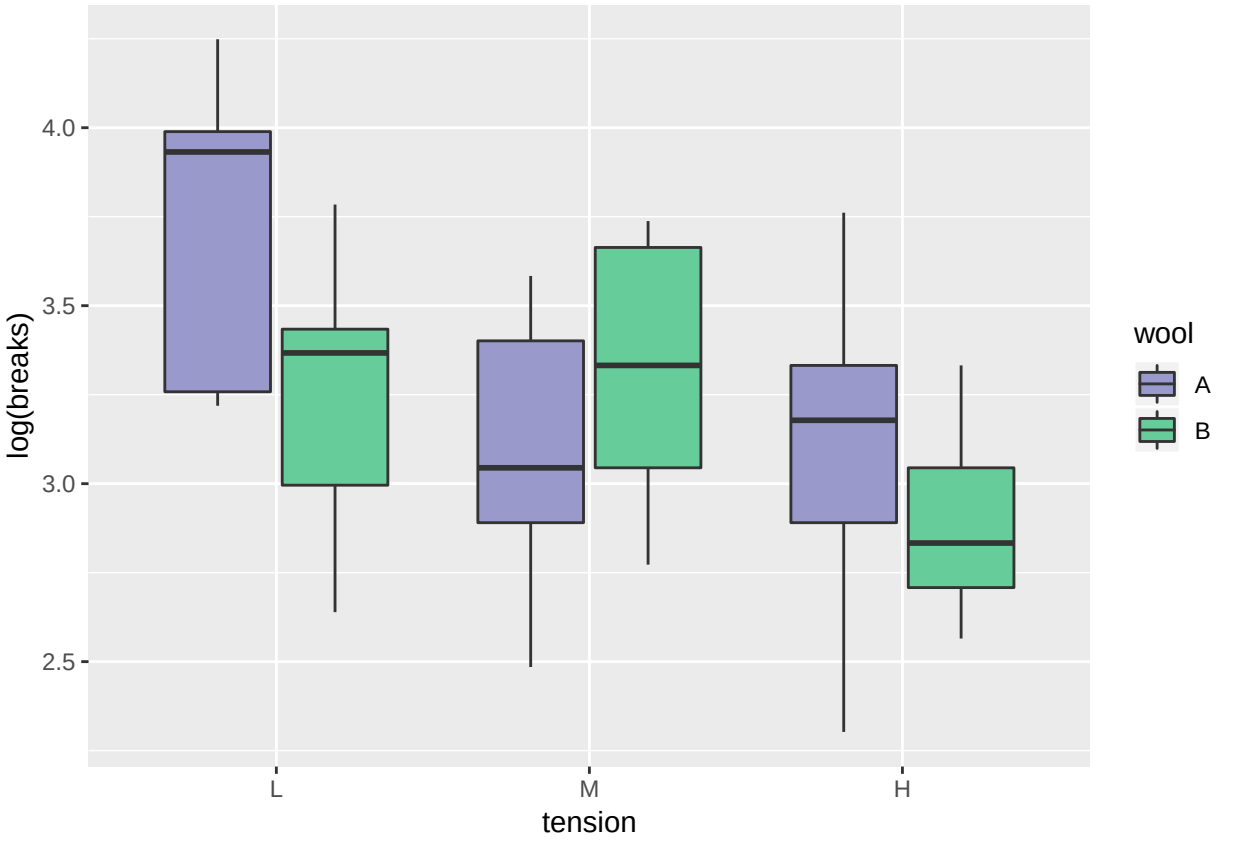
```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 3.5.3
```

Ejercicio 1

a) Primero cargamos los datos necesarios para resolver el ejercicio, del paquete "datasets". Luego, para cada una de las categorías de tension, graficamos boxplots paralelos para la variable breaks clasificando por tipo de wool

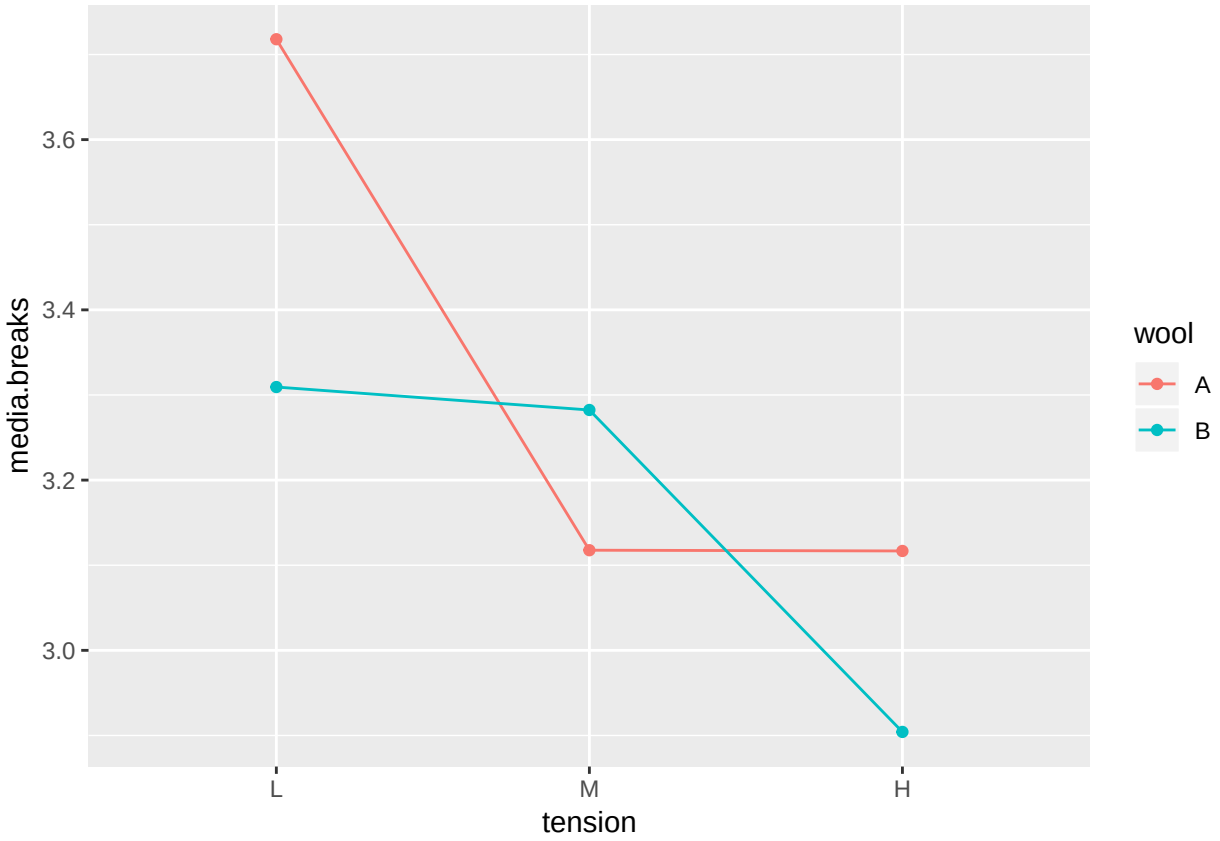
```
require(datasets)
telas <- warpbreaks

gg1<-ggplot(data=telas,aes(x=tension,y=log(breaks),fill=wool))+
  geom_boxplot()+
  scale_fill_manual(values=c( "#9999CC", "#66CC99"))

gg1
```



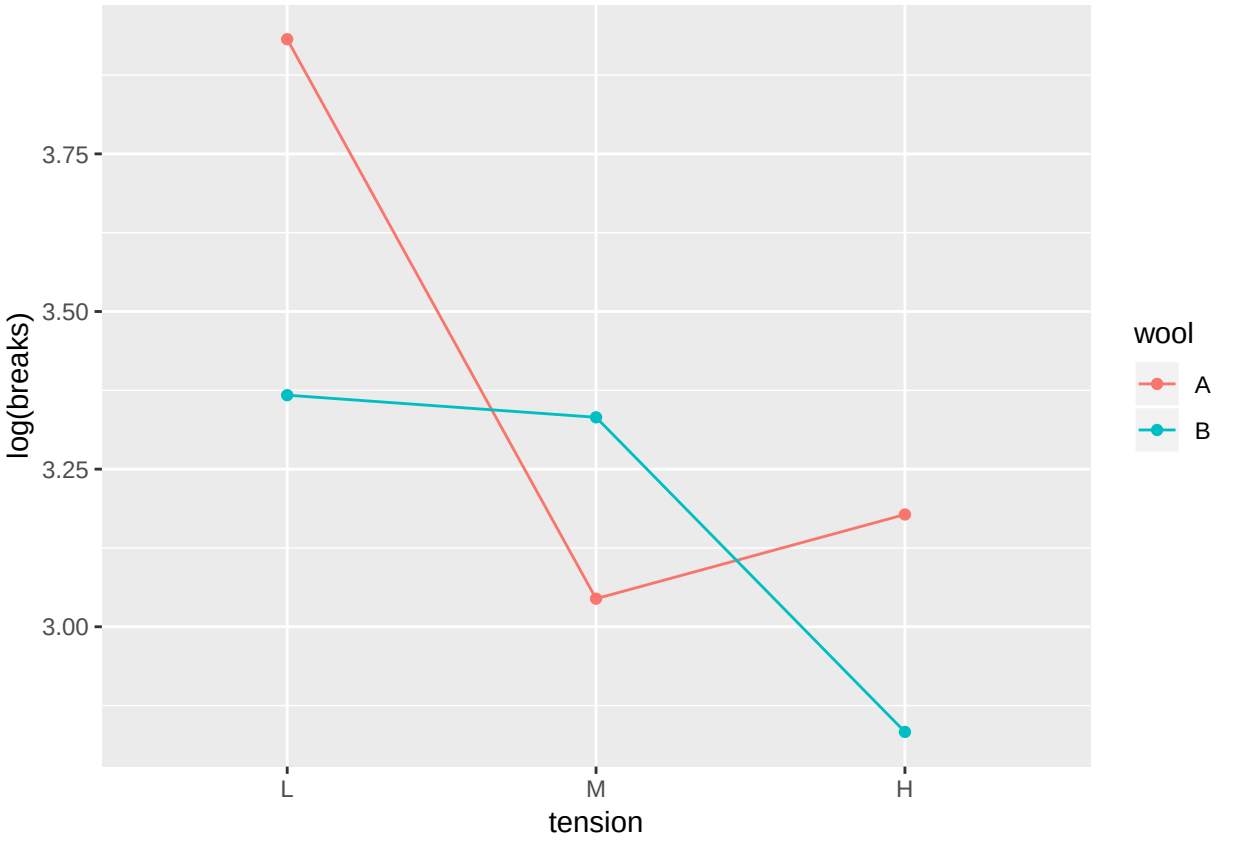
Podemos observar que la tendencia de los boxplot para los diferentes tipos de lana son diferentes. Esto podría estar indicando que hay interacción entre la tensión aplicada y el tipo de lana, al momento de predecir la cantidad de fallas en la tela.

b) Para poder observar de otra forma la presencia de interacción, graficamos los promedios de la variable "breaks" en función de la tensión, distinguiendo en los dos tipos de lana:

```
#haciendo cada cuenta

b1a<-mean(log(telas$breaks[telas$tension=="L" & telas$wool=="A"]))
b1b<-mean(log(telas$breaks[telas$tension=="L" & telas$wool=="B"]))
b2a<-mean(log(telas$breaks[telas$tension=="M" & telas$wool=="A"]))
b2b<-mean(log(telas$breaks[telas$tension=="M" & telas$wool=="B"]))
b3a<-mean(log(telas$breaks[telas$tension=="H" & telas$wool=="A"]))
b3b<-mean(log(telas$breaks[telas$tension=="H" & telas$wool=="B"]))

medias<-data.frame("media breaks"=c(b1a,b1b,b2a,b2b,b3a,b3b),"wool"=rep(c("A","B"),3),"tension"=rep(c("
medias$tension<-relevel(medias$tension,"M") #ordeno los factores como quiero
medias$tension<-relevel(medias$tension,"L")

gg3<-ggplot(data=medias,aes(x=tension,y=media.breaks,col=wool,group=wool))+
  geom_point()+
  geom_line()
gg3
```




```
# o mejor, usando funciones de ggplot2

ggplot(telas, aes(x = tension, y = log(breaks),group=wool,color = wool)) +
  stat_summary(fun.y = median, geom = "point", aes()) +
  stat_summary(fun.y = median, geom = "line", aes(group=wool))
```



Podemos observar más claramente la tendencia que se veía en los boxplot, parece haber interacción entre las variables.

c) Realizamos el ajuste del modelo para predecir el número de roturas, proponiendo como distribución de la variable respuesta a la Poisson y un modelo con interacción.

El modelo será:

$log(breaks) = \beta_0 + \beta_1 * B + \beta_2 * H + \beta_3 * M + \beta_4 * B * H + \beta_5 * B * M$

```
ajuste<-glm(breaks~wool*tension,family="poisson", data=telas)
summary(ajuste)
```

```
##
## Call:
## glm(formula = breaks ~ wool * tension, family = "poisson", data = telas)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3383  -1.4844  -0.1291   1.1725   3.5153
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     3.79674    0.04994  76.030  < 2e-16 ***
## woolB          -0.45663    0.08019  -5.694 1.24e-08 ***
## tensionM       -0.61868    0.08440  -7.330 2.30e-13 ***
## tensionH       -0.59580    0.08378  -7.112 1.15e-12 ***
## woolB:tensionM  0.63818    0.12215   5.224 1.75e-07 ***
```

```
## woolB:tensionH  0.18836    0.12990   1.450    0.147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 297.37  on 53  degrees of freedom
## Residual deviance: 182.31  on 48  degrees of freedom
## AIC: 468.97
##
## Number of Fisher Scoring iterations: 4
```

d) Estimamos los coeficientes y evaluamos cuales son significativos:

```
ajuste$coefficients
```

```
##    (Intercept)          woolB       tensionM       tensionH woolB:tensionM
##      3.7967368     -0.4566272     -0.6186830     -0.5957987      0.6381768
## woolB:tensionH
##      0.1883632
```

Vemos de la salida que son significativos todos los coeficientes salvo el correspondiente a la interacción entre tipo de lana B y tension alta (High)

e) Analizamos para que combinación de lana y tensión el modelo predice el mayor y el menor número de roturas. Para eso utilizamos la estimación de los coeficientes hallada en el ítem anterior. Para cada vector $\underline{x} = (1, B, H, M, B * H, B * M)$ calculamos la predicción del número de roturas:

```
x1<-c(1,0,0,0,0,0)
x2<-c(1,0,1,0,0,0)
x3<-c(1,0,0,1,0,0)
x4<-c(1,1,0,0,0,0)
x5<-c(1,1,1,0,1,0)
x6<-c(1,1,0,1,0,1)

exp(t(x1)%*%ajuste$coefficients)
```

```
##          [,1]
## [1,] 44.55556
```

```
exp(t(x2)%*%ajuste$coefficients)
```

```
##      [,1]
## [1,]   24
```

```
exp(t(x3)%*%ajuste$coefficients)
```

```
##          [,1]
## [1,] 24.55556
```

```
exp(t(x4)%*%ajuste$coefficients)
```

```
##          [,1]
## [1,] 28.22222
```

```
exp(t(x5)%*%ajuste$coefficients)
```

```
##          [,1]
## [1,] 28.77778
```

```r
exp(t(x6)%*%ajuste$coefficients)
```

```
##          [,1]
## [1,] 18.77778
```

```r
#Mayor cantidad esperada de fallas: Lana A y TL
#Menor cantidad esperada de fallas: Lana B y TH
```

f) Calculamos un intervalo asintótico de nivel 0.95 para el valor esperado de roturas pala las dos combinaciones extremas que encontramos en el ítem anterior:

```r
intervalo_media<-function(salida,alfa,x0)
{
  d<-as.matrix(x0)
  psi<-t(d)%*%(summary(salida)$coef[,1])
  var<-summary(salida)$cov.scaled
  e<-qnorm(1-alfa/2)*sqrt(t(d)%*%var%*%(d))
  ic<-c(exp(psi-e),exp(psi+e))
  return(ic)
}

ic1<-intervalo_media(ajuste,0.05,x1)
ic2<-intervalo_media(ajuste,0.05,x6)
```

Calculamos las longitudes de los intervalos:

```r
L1<-ic1[2]-ic1[1]
L2<-ic2[2]-ic2[1]
L2;L1
```

```
## [1] 5.683593
```

```
## [1] 8.735749
```

Las longitudes no son iguales.

g) Realizamos una tabla de ANOVA para este caso. Podemos observar que cada variable (factor) que agregamos aporta al modelo, al igual que cuando agregamos las interacciónes. EL hecho de rechazar todas las hipótesis nulas implica que el modelo con todas las variables es un mejor predictor para la cantidad esperada de fallas en la tela.

```r
anova(ajuste, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: breaks
##
## Terms added sequentially (first to last)
##
##
##              Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                         53     297.37
## wool          1   16.039        52     281.33 6.206e-05 ***
## tension       2   70.942        50     210.39 3.938e-16 ***
## wool:tension  2   28.087        48     182.31 7.962e-07 ***
## ---
```

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

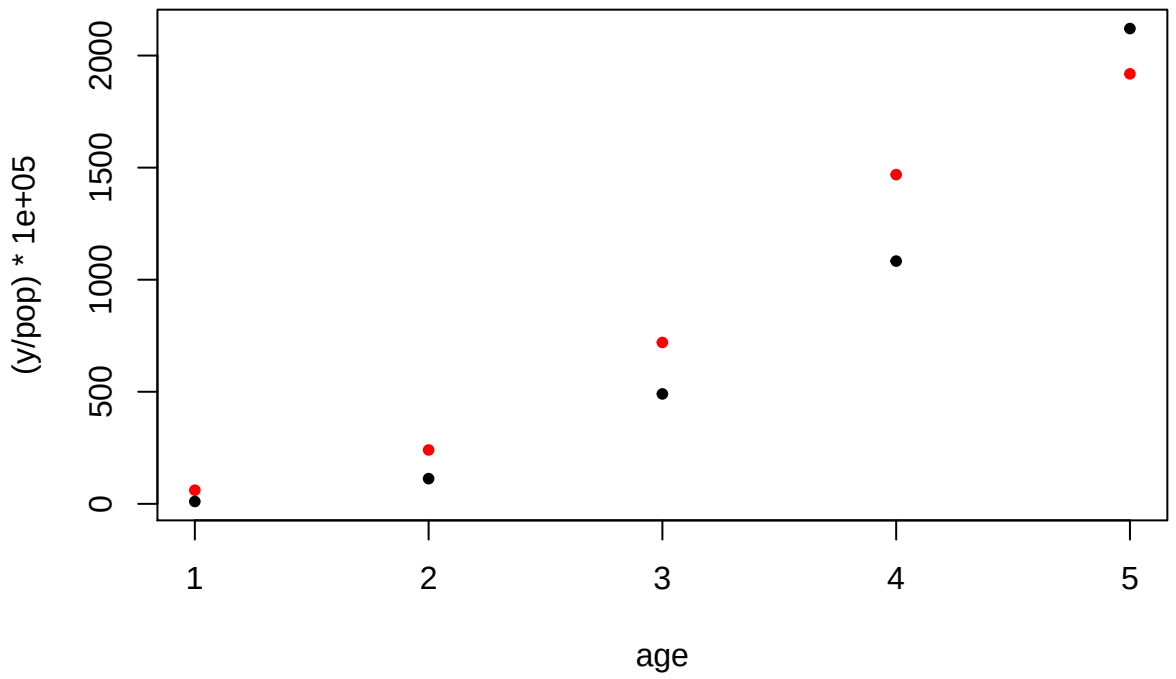
Ejercicio 2

En este ejercicio analizamos los datos provenientes de un estudio muy famoso realizado por Sir Richard Doll y colegas en 1951. Se quiere estudiar si la tasa de muerte es mayor en médicos que fuman sobre los médicos que no fuman.

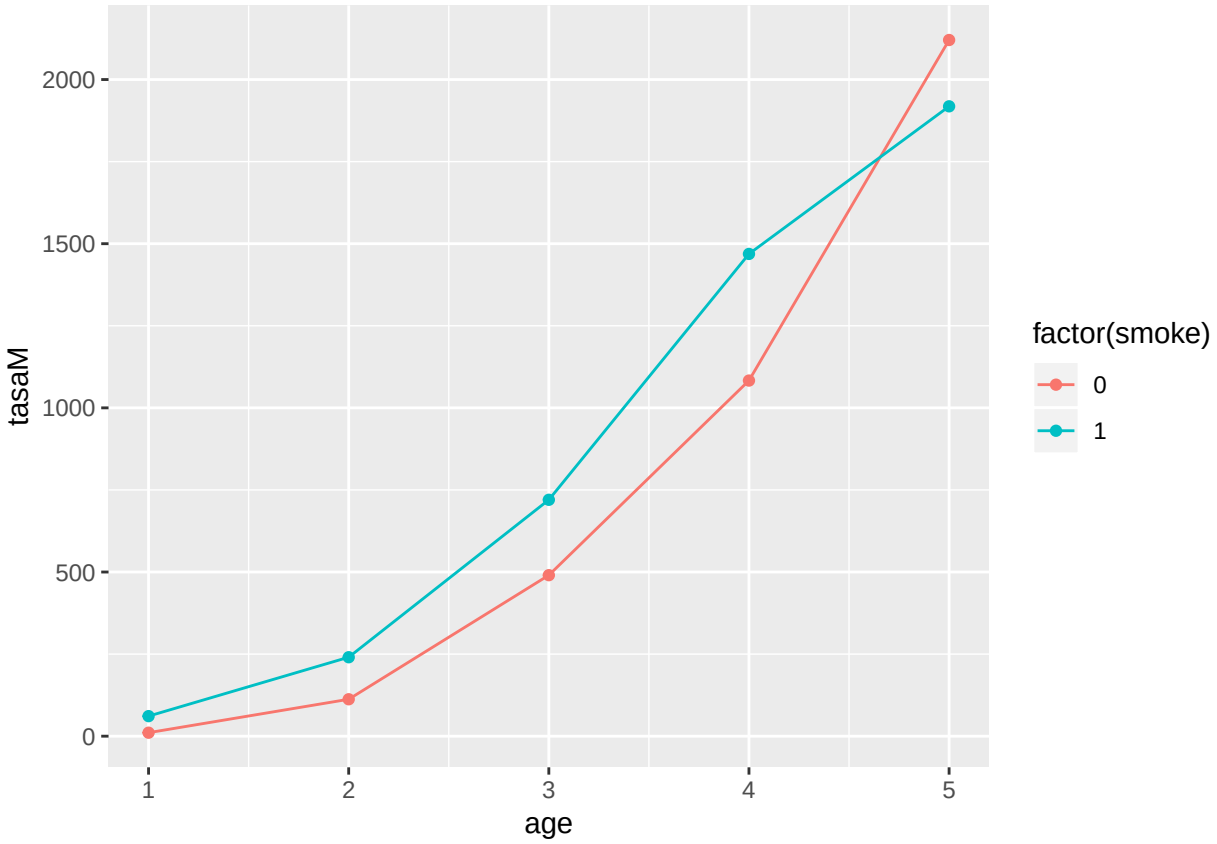
a) Según el modelo propuesto, el término log(population) es una constante que cambia para cada observación, pero que no es un valor a estimar ni una variable, da una componente a la ordenada al origen constante pero diferente para cada observación. Por este motivo no está acompañada de un coeficiente a estimar, por lo que para incluirlo en la función glm de R lo tenemos que agregar como opción de offset.

b) Graficamos la tasa de muerte cada 100000 habitantes distinguiendo entre los fumadores y no fumadores, en función de la edad.

```
y <- c(32,104,206,186,102,2,12,28,28,31)
smoke<- c(rep(1,5),rep(0,5))
age <- c(1:5,1:5)
pop<- c(52407,43248,28612,12663,5317,18790,10673,5710,2585,1462)

logpop<- log(pop)
age2 <- age*age
smkage<- smoke*age

plot(age,(y/pop)*100000,col=factor(smoke),pch=20)
```



```
datos<-data.frame("tasaM"=(y/pop)*100000,"smoke"=smoke,
                  "age"=age)

ggplot(datos,aes(x=age,y=tasaM,col=factor(smoke)))+
  geom_point()+
  geom_line()
```



Este gráfico podría estar dando indicios de algún tipo de interacción entre la variable age y smoke.

c) Ajustamos el modelo y calculamos intervalos de confianza de nivel asintótico 0.95 para los coeficientes del modelo

```
ajuste<-glm(y~age+age2+smoke+smkage, offset = logpop, family = poisson,x=T)
summary(ajuste)
```

```
##
## Call:
## glm(formula = y ~ age + age2 + smoke + smkage, family = poisson,
##     offset = logpop, x = T)
##
## Deviance Residuals:
##        1         2         3         4         5         6         7
##  0.43820  -0.27329  -0.15265   0.23393  -0.05700  -0.83049   0.13404
##        8         9        10
##  0.64107  -0.41058  -0.01275
##
## Coefficients:
```

```
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.79176    0.45008 -23.978  < 2e-16 ***
## age           2.37648    0.20795  11.428  < 2e-16 ***
## age2         -0.19768    0.02737  -7.223 5.08e-13 ***
## smoke         1.44097    0.37220   3.872 0.000108 ***
## smkage       -0.30755    0.09704  -3.169 0.001528 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 935.0673  on 9  degrees of freedom
## Residual deviance:   1.6354  on 5  degrees of freedom
## AIC: 66.703
##
## Number of Fisher Scoring iterations: 4
```

```
#intervalo de confianza...
IC<-matrix(0,ncol=2,nrow=5)

for(i in 1:5)
{
  IC[i,]<-c(summary(ajuste)$coef[i,1]-qnorm(0.975)*summary(ajuste)$coef[i,2],
            summary(ajuste)$coef[i,1]+qnorm(0.975)*summary(ajuste)$coef[i,2])
}
colnames(IC)<-c("LI","LS")
rownames(IC)<-c("beta 0", "beta 1", "beta 2", "beta 3",
                "beta 4")

IC
```

```
##                 LI         LS
## beta 0 -11.6738977 -9.9096274
## beta 1   1.9689066  2.7840501
## beta 2  -0.2513157 -0.1440374
## beta 3   0.7114755  2.1704682
## beta 4  -0.4977452 -0.1173509
```

d) Calculamos un intervalo de confianza para el RR del coeficiente β_2 es decir queremos analizar si el simple hecho de fumar afecta la tasa de muerte. Lo calculamos para el caso de no interacción, por ahí no el más adecuado para este caso particular.

```
ic<-exp(IC[3,])
```

Podríamos también calcularlo para cada estrato de edad, ya que al haber solo 5 podríamos hacer todos los cálculos.

Ejercicio 3

Consideremos el conjunto de datos del archivo postdoc.txt para estudiar el efecto de sobredispersión en regresión Poisson.

a) Realizamos un ajuste para CITS en función de AGE, MAR, DOC, UND y una ordenada al origen. Observamos con la salida que todos los coeficientes resultan significativamente distintos de cero

```r
doc<-read.table("postdoc.txt",header = TRUE)
ajuste<-glm(cits~age+mar+doc+und, family = poisson,data=doc)
summary(ajuste)
```

```
##
## Call:
## glm(formula = cits ~ age + mar + doc + und, family = poisson,
##     data = doc)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.5444  -2.2922  -1.4563  -0.2132  15.7529
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.9199234  0.1906739   4.825 1.40e-06 ***
## age         -0.0326929  0.0048392  -6.756 1.42e-11 ***
## mar          0.2373556  0.0694976   3.415 0.000637 ***
## doc          0.0006705  0.0002302   2.913 0.003585 **
## und          0.1545935  0.0153426  10.076  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 4303.7  on 556  degrees of freedom
## Residual deviance: 4129.1  on 552  degrees of freedom
## AIC: 5179.3
##
## Number of Fisher Scoring iterations: 6
```

b) Calculamos los residuos deviance y de Pearson, luego los comparamos con sus grados de libertad. Observamos que dan muy diferentes, lo cual podría estar indicando que hay Sobredisperción, ya que no se está realizando un buen ajuste del modelo usando regresión de Poisson.

```r
 #Deviance
gl<-summary(ajuste)$df.residual
residuos_dev<-resid(ajuste,type="deviance")
D<-sum(residuos_dev^2)

#comparo D con GL
D/gl #es mucísimo más grande que 1, debo rechazar
```

```
## [1] 7.480336
```

```r
#Pearson
sum(resid(ajuste,type="pearson")*resid(ajuste,type="pearson"))/summary(ajuste)$df.residual
```

```
## [1] 13.45508
```

c) Vemos el summary de los residuos y graficamos boxplot para comparar.

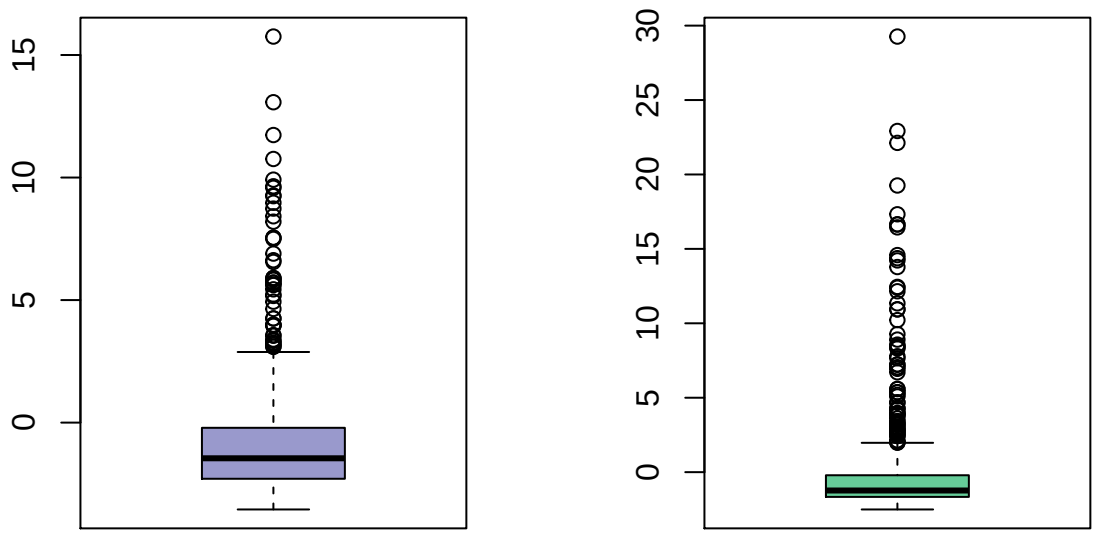
```r
summary(resid(ajuste,type="deviance"))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.5444 -2.2922 -1.4563 -0.6529 -0.2132 15.7529
```

```r
summary(resid(ajuste,type="pearson"))
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -2.506236 -1.662904 -1.233450  0.001319 -0.209035 29.264947
```

```r
par(mfrow=c(1,2))
boxplot(resid(ajuste,type="deviance"), col= "#9999CC")
boxplot(resid(ajuste,type="pearson"),col= "#66CC99")
```




```r
par(mfrow=c(1,1))
```

d) Realizamos el mismo estudio pero ahora considerando el ajuste con una Binomial Negativa.

```r
library(MASS)
postdoc.cit.bn<- glm.nb(cits ~ age + mar + doc + und, data = doc)
summary(postdoc.cit.bn)
```

```
##
## Call:
## glm.nb(formula = cits ~ age + mar + doc + und, data = doc, init.theta = 0.4372597351,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5319  -1.3009  -0.5818  -0.0918   2.9621
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)  0.9524015  0.5968806   1.596  0.11057
## age         -0.0318508  0.0156514  -2.035  0.04185 *
## mar          0.2097149  0.1973908   1.062  0.28804
## doc          0.0007496  0.0006660   1.126  0.26036
## und          0.1434709  0.0441906   3.247  0.00117 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.4373) family taken to be 1)
##
##     Null deviance: 586.05  on 556  degrees of freedom
## Residual deviance: 566.26  on 552  degrees of freedom
## AIC: 2432.6
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.4373
##           Std. Err.:  0.0323
##
##  2 x log-likelihood:  -2420.5750
```

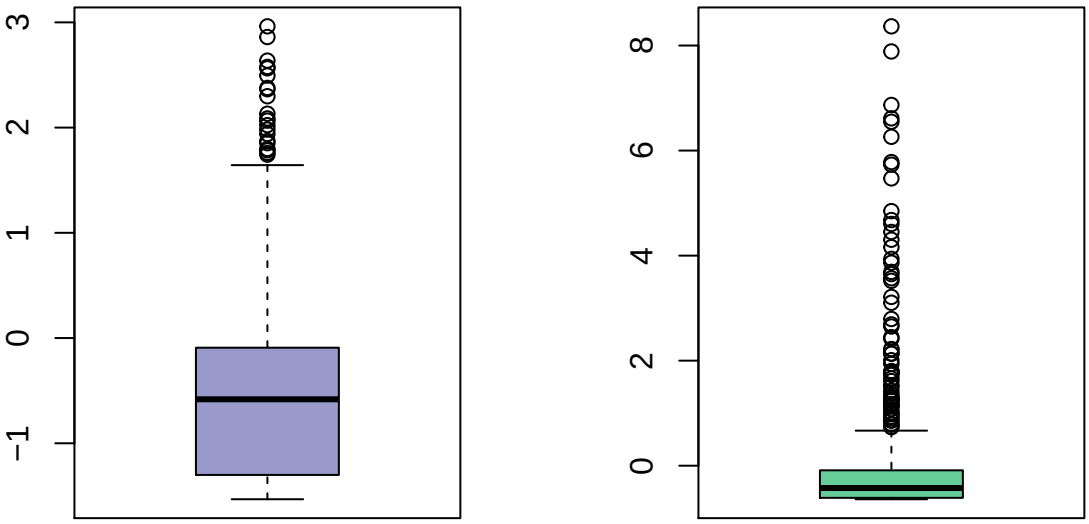
```
summary(resid(postdoc.cit.bn,type="deviance"))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.5319 -1.3008 -0.5818 -0.4686 -0.0918  2.9621
```

```
summary(resid(postdoc.cit.bn,type="pearson"))
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.638262 -0.611645 -0.424655  0.000021 -0.087548  8.364750
```

```
par(mfrow=c(1,2))
boxplot(resid(postdoc.cit.bn,type="deviance"), col= "#9999CC")
boxplot(resid(postdoc.cit.bn,type="pearson"),col= "#66CC99")
```




```
par(mfrow=c(1,1))


##Calculo el estadistico de deviance y comparo con grados de libertad
sum((resid(postdoc.cit.bn,type="deviance")*resid(postdoc.cit.bn,type="deviance")))/
summary(postdoc.cit.bn)$df.residual
```

```
## [1] 1.025831
```

```
##Calculo el estadistico de Pearson y comparo con grados de libertad
sum(resid(postdoc.cit.bn,type="pearson")*resid(postdoc.cit.bn,type="pearson"))/
summary(postdoc.cit.bn)$df.residual
```

```
## [1] 1.555445
```

Por un lado vemos, que si bien loas estimaciones obtenidas para los coeficientes son similares a las anteriores con el ajuste Poisson, cambia su significación. También vemos que los residuos, han cambiado la escala, en particular los valores absolutos de los residuos de la deviance son inferiores a 3.

Por otro lado, ahora si podemos ver que ambos estadísticos de bondad de ajuste comparados con sus grados de libertad dan muy similares, y el valor de AIC es casi la mitad, podemos decir que el ajuste considerando la Binomial Negativa es mucho mejor que considerando la distribución de Poisson.

f) Con esta nueva idea, revisamos lo resuelto en el ejercicio 1:

```
require(datasets)
telas <- warpbreaks
ajuste.bn<-glm.nb(breaks~wool*tension,data=telas)
summary(ajuste.bn)
```

```
##
## Call:
## glm.nb(formula = breaks ~ wool * tension, data = telas, init.theta = 12.08216462,
##     link = log)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.09611  -0.89383  -0.07212   0.65270   1.80646
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)      3.7967     0.1081  35.116  < 2e-16 ***
## woolB           -0.4566     0.1576  -2.898 0.003753 **
## tensionM        -0.6187     0.1597  -3.873 0.000107 ***
## tensionH        -0.5958     0.1594  -3.738 0.000186 ***
## woolB:tensionM   0.6382     0.2274   2.807 0.005008 **
## woolB:tensionH   0.1884     0.2316   0.813 0.416123
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(12.0822) family taken to be 1)
##
##     Null deviance: 86.759  on 53  degrees of freedom
## Residual deviance: 53.506  on 48  degrees of freedom
## AIC: 405.12
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  12.08
##           Std. Err.:  3.30
##
##  2 x log-likelihood:  -391.125
```

Para este caso, el valor de AIC da similar, pero es menor usando la distribución Binomial Negativa.