Centro de Investigación en Matemáticas (CIMAT). Unidad Monterrey

Cómputo Estadístico
Tarea 1

Marcelo Alberto Sanchez Zaragoza

30 de septiembre de 2021

1. Problema 1

La siguiente tabla muestra los resultados parciales de dos encuestas que forman parte de un estudio para evaluar el desempeño del Primer Ministro del Canadá. Se tomó una muestra aleatoria de 1600 ciudadanos canadienses mayores de edad y en los renglones se observa que 944 ciudadanos aprobaban el desempeño del funcionario, mientras que las columnas muestran que, seis meses después de la primera encuesta, sólo 880 aprueban su desempeño.

Inciso a)
Escriba la logverosimilitud correspondiente. Muestre explícitamente (i. e. maximizando la logverosimilitud que el estimador máximo verosimilitud

Primera encuesta	Y=1,Aprueba(S.E)	Y=0, Desaprueba(S.E)	Total
X=1, Aprueba	794	150	944
X=2, Desaprueba	86	570	656
Total	880	720	1600

Cuadro 1.1: Encuesta con datos de primera y segunda encuentas(S.E.)

para β_1 1 es el logaritmo de la tasa de momios de la tabla dada (En general, en regresión logística los estimadores de máxima verosimilitud no tienen una forma explícita, sin embargo, en el presente caso si)

Inciso b)
Sea p_1 la proporción de ciudadanos que aprueban el desempeño del ministro al tiempo inicial y sea p_2 la proporción correspondiente seis meses después. Considere la hipótesis: $H0 : p_1 = p_2$, ¿Cómo puede hacerse esta prueba?

Solución

Inciso a)
Para el primer inciso tenemos que partir de lo siguiente:

Observamos que $Y|x$ tiene distribución Bernoulli ya que hay dos posibles resultados, aprobación o no al desempeño del Primer Ministro. Por lo que tenemos: $P(Y = 1|x) = p$ y $P(Y = 0|x) = 1 - p$ y bajo el modelo de regresión logística, tenemos: $p_i = P(Y_i = 1|x_i) = \frac{1}{1+e^{-\beta_0-\beta_1 x_i}}$.
La función de verosimilitud es:

$$L(\beta) = L(\beta|y_i|x_1, ..., y_{1600}|x_{1600}) = \prod_{i=0}^{n} f_\beta(Y_i|x_i) = \prod_{i=1}^{n} p_i^{y_i}(1 - p_i^{1-y_i})$$

donde $p_i = \frac{1}{1+e^{-\beta_0-\beta_1 x_i}}$, tomando logaritmo a $L(\beta)$, así tenemos la logverosimilitud:

$$l(\beta) = log(L(\beta)) = \sum_{i=1}^{n} y_i log(p_i) + (1 - y_i) log(1 - p_i)$$

Ahora queremos encontrar la derivada respecto de β, pero antes vamos a encontrar la derivada de $\frac{\partial p_i}{\partial \beta}$, pero como tenemos dos variables en este caso vamos a encontrar la derivada respecto a β_0 y β_1, así:

$$\frac{\partial p_i}{\partial \beta_0} = -\frac{(e^{-\beta_0 - \beta_1 x_i})(-1)}{(1 + e^{-\beta_0 - \beta_1 x_i})^2} = p_i(1 - p_i)$$
$$\frac{\partial p_i}{\partial \beta_1} = -\frac{(e^{-\beta_0 - \beta_1 x_i})(x_i)}{(1 + e^{-\beta_0 - \beta_1 x_i})^2} = x_i p_i(1 - p_i)$$

así:

$$\begin{aligned} \frac{\partial l(\beta)}{\partial \beta} &= \sum_{i=1}^{n} y_i \frac{1}{p_i} \frac{\partial p_i}{\partial \beta} - (1 - y_i) \frac{1}{(1 - y_i)} \frac{\partial p_i}{\partial \beta} \\ &= \sum_{i=1}^{n} \left[\tfrac{y_i}{p_i} - \tfrac{1 - y_i}{1 - p_i} \right] \frac{\partial p_i}{\partial \beta} \\ &= \sum_{i=1}^{n} \left[\tfrac{y_i - p_i}{p_i(1 - p_i)} \right] \frac{\partial p_i}{\partial \beta} \end{aligned}$$

Ahora sustituimos la derivada que encontramos respecto a β_0 en lo anterior y tenemos:

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n} \left[\tfrac{y_i - p_i}{p_i(1 - p_i)} \right] [p_i(1 - p_i)] = \sum_{i=1}^{n} (y_i - p_i)$$

Análogamente para β_1, es:

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n} \left[\tfrac{y_i - p_i}{p_i(1 - p_i)} \right] [x_i p_i(1 - p_i)] = \sum_{i=1}^{n} x_i(y_i - p_i)$$

Ya que tenemos nuestras expresiones sustituidas las podemos igualar a cero y se puede obsevar que n = 1600, además hay casos para ambas variables x y y. Para cada variable sabemos que $x = \{0,1\}$ y $y = \{0,1\}$. De esta forma podemos ir separando nuestra suma en pequeñas sumas dependiendo los posibles casos. Así para la expreón donde se sustituyo la derivada respecto a β_0, tenemos:

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n}(y_i - p_i) = \sum_{\substack{i=1 \\ y_i=1 \\ x_i=1}}^{794}(1 - \frac{1}{1+e^{-\beta_0-\beta_1}}) + \sum_{\substack{i=1 \\ y_i=0 \\ x_i=1}}^{150}(0 - \frac{1}{1+e^{-\beta_0-\beta_1}}) + \sum_{\substack{i=1 \\ y_i=1 \\ x_i=0}}^{86}(1 - \frac{1}{1+e^{-\beta_0}}) +$$

$$\sum_{\substack{i=1 \\ y_i=0 \\ x_i=0}}^{794}(0 - \frac{1}{1+e^{-\beta_0}}) = 794 - \frac{794}{1+e^{-\beta_0-\beta_1}} - \frac{150}{1+e^{-\beta_0-\beta_1}} + 86 - \frac{86}{1+e^{-\beta_0}} - \frac{570}{1+e^{-\beta_0}}$$

$$= 880 - \frac{944}{1+e^{-\beta_0-\beta_1}} - \frac{656}{1+e^{-\beta_0}} = 0$$

Donde podemos mover las fracciones de un lado y tendremos: $880 = \frac{944}{1+e^{-\beta_0-\beta_1}} + \frac{656}{1+e^{-\beta_0}}$.
Análogamente volvemos a hacer las pequeñas sumas con la expresión donde se sustituyo la derivada con respecto a β_1, así:

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n} x_i(y_i - p_i) = \sum_{\substack{i=1 \\ y_i=1 \\ x_i=1}}^{794}(1)(1 - \frac{1}{1+e^{-\beta_0-\beta_1}}) + \sum_{\substack{i=1 \\ y_i=0 \\ x_i=1}}^{150}(1)(0 - \frac{1}{1+e^{-\beta_0-\beta_1}}) + 0 + 0$$

$$= 794 - \frac{794}{1+e^{-\beta_0-\beta_1}} - \frac{150}{1+e^{-\beta_0-\beta_1}} = 794 - \frac{944}{1+e^{-\beta_0-\beta_1}} = 0$$

Volvemos a colocar las fracciones de un lado y tenemos la siguiente expresión: $794 = \frac{944}{1+e^{-\beta_0-\beta_1}}$.
Lo que nos resta hacer es realizar un pequeño sistema de ecuaciones para encontrar los valores estimados para β_0 y β_1, como primer paso va-

mos a sustituir lo que encontramos recientemente ($794 = \frac{944}{1+e^{-\beta_0-\beta_1}}$) en $880 = \frac{944}{1+e^{-\beta_0-\beta_1}} + \frac{656}{1+e^{-\beta_0}}$, así:

$$880 = 944(\frac{794}{944}) + \frac{656}{1+e^{-\beta_0}}$$
$$\frac{880-794}{656} = \frac{1}{1+e^{-\beta_0}}$$
$$\hat{\beta}_0 = -log(\frac{656}{86} - 1) = -1.8912$$

Ahora solo sustituimos en $794 = \frac{944}{1+e^{-\beta_0-\beta_1}}$ y encontramos que $\hat{\beta}_1 = -log(\frac{944}{794} - 1) - \beta_0 = 3.5577$.

Inciso b)
Para este inciso debemos realizar una prueba de igualdad de proporciones, en primer lugar debemos establecer la hipótesis nula y alternativa:

$$H_0 : p_1 = p_2 \;\; vs \;\; H_1 : p_1 \neq p_2$$

El estadístico de prueba z es:

$$z = \frac{(x_1/n_1) - (x_2/n_2)}{\sqrt{m(1-m)[(1/n_1) + (1/n_2)]}}$$

donde $m = \frac{x_1+x_2}{n_1+n_2}$. Ahora solo debemos sustituir los valores, en este caso $p_1 = \frac{944}{1600}, p_2 = \frac{880}{1600}, m = \frac{944+880}{1600+1600}$, así:

$$z = \frac{0.59 - 0.55}{\sqrt{0.57(1-0.57)[(1/1600) + (1/1600)]}} = 2.2852$$

También podemos observar el valor del p-valor donde tenemos que p-valor $= P(Z < -2.2852) + P(Z > 2.2852) = 0.0223$.

Dado que el valor tabular con $\alpha = 0.05$ es 1.96, observamos que nuestro estadístico de prueba es mayor por lo que rechazamos H_0.
Podemos decir que la proporción de personas que aprueban el desempeño del Primer Ministro de Canadá ha cambiado despúes de 6 meses.

2. Problema 2

Se tiene la siguiente tabla donde se eligen varios niveles de ronquidos y se ponen en relación con una enfermedad cardíaca. Se toman como puntuaciones relativas de ronquidos los valores $\{0\ 2\ 4\ 5\}$.

Ajuste un modelo lineal generalizado logit y probit (*investigar sobre el link probit*) para analizar si existe una relación entre los ronquidos y la posibilidad de tener una enfermedad cardiaca.

Solución

```
## Lo primero que hicimos fue ajustar un modelo lineal generalizado logit
roncas <- c(0, 2, 4, 5)
modelo.logit <- glm( cbind( SI = c(24, 35, 21, 30),
                    NO = c(1355, 603, 192, 224) ) ~ roncas,
                  family = binomial(link = logit) )

summary(modelo.logit)$coefficients

              Estimate Std. Error    z value      Pr(>|z|)
(Intercept) -3.8662481 0.16621436 -23.260614 1.110885e-119
roncas       0.3973366 0.05001066   7.945039  1.941304e-15

## El segundo modelo lineal generado probit
roncas <- c(0, 2, 4, 5)
modelo.probit <- glm( cbind( SI = c(24, 35, 21, 30),
```

```
                         NO = c(1355, 603, 192, 224) ) ~ roncas,
                        family = binomial(link = probit) )

summary(modelo.probit)$coefficients

              Estimate Std. Error   z value      Pr(>|z|)
(Intercept) -2.0605516 0.07016673 -29.36651 1.471042e-189
roncas       0.1877705 0.02348052   7.99686  1.276323e-15
```

Hay que comenzar con el primer caso, la regresión logistica, en dichos resultados el valor para β fue de 0.3973, podemos decir que no hay mucho aporte por parte de la variable *Roncar* para decir si hay una enfermedad cardiaca.
En la segundo modelo lineal tampoco proporciona información sobre la variable *Roncar* al problema de tener una enfermendad cardiaca.
Finalmente observamos que a medida que crecen la cantidad de ronquidos no da como resultado una enfermedad cardiaca. Observando los resultados de cada modelo podemos decir que de acuerdo al AIC nos inclinamos por el resultado que nos dio el modelo lineal probit.

3. Problema 3

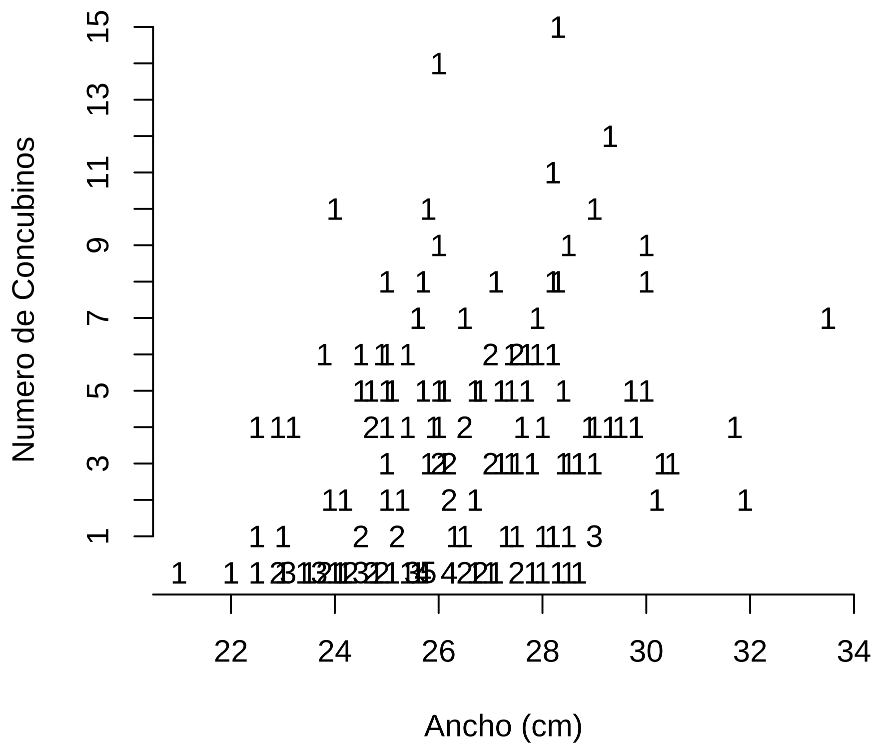
Entre los cangrejos cacerola se sabe que cada hembra tiene un macho en su nido, pero puede tener más machos concubinos. Se considera que la variable respuesta es el número de concubinos y las variables explicativas son color, estado de la espina central, peso y anchura del caparazón
Realiza e interpretar los resultados de ajustar un modelo lineal generalizado tipo poisson.

Solución

```r
tabla <- read.csv("C:/Users/Marcelo Sanchez/OneDrive/Escritorio/Tercer Semestre CIMA
                 , header = T)
dimnames(tabla)[[2]] <- c("color","spine","width","satell","weight")
names(tabla)

[1] "color"  "spine"  "width"  "satell" "weight"
```

Realizamos un primer gráfico que nos ayuda a observar la dispersión de la cantidad de machos concubinos para cada hembra de acuerdo al ancho del caparazón.

```r
# Grafico de dispersion
plot.tabla <- aggregate(rep (1, nrow(tabla)),
                      list(Sa = tabla$satell, W = tabla$width), sum)
plot (y = plot.tabla$Sa , x = plot.tabla$W, xlab = "Ancho (cm)",
      ylab = "Numero de Concubinos", bty = "L", axes = F, type = "n")
axis(2, at = 1:15)
axis(1, at = seq(20, 34, 2))
text(y = plot.tabla$Sa , x = plot.tabla$W, labels = plot.tabla$x)
```




```
############### Se puede ajustar un modelo GLM de Poisson.
###############
log.fit <- glm(satell ~ width, family = poisson ,
               data = tabla)
summary(log.fit )

Call:
```

```
glm(formula = satell ~ width, family = poisson, data = tabla)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8526  -1.9884  -0.4933   1.0970   4.9221

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.30476    0.54224  -6.095  1.1e-09 ***
width        0.16405    0.01997   8.216  < 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: 632.79  on 172  degrees of freedom
Residual deviance: 567.88  on 171  degrees of freedom
AIC: 927.18

Number of Fisher Scoring iterations: 6
```

```
p0 = log.fit$null.deviance - log.fit$deviance ## calculamos la residual
p0
```

```
[1] 64.91309
```

```
1 - pchisq(p0,1)
```

```
[1] 7.771561e-16
```

```
log.fit2 <- glm(satell ~ weight, family = poisson ,
               data = tabla)
summary(log.fit2 )
```

```
Call:
glm(formula = satell ~ weight, family = poisson, data = tabla)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9307  -1.9981  -0.5627   0.9298   4.9992

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -4.284e-01  1.789e-01  -2.394   0.0167 *
weight       5.893e-04  6.502e-05   9.064   <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: 632.79  on 172  degrees of freedom
Residual deviance: 560.87  on 171  degrees of freedom
AIC: 920.16

Number of Fisher Scoring iterations: 5

p0 = log.fit2$null.deviance - log.fit2$deviance ## calculamos la residual
p0

[1] 71.92524

1 - pchisq(p0,1)

[1] 0

log.fit3 <- glm(satell ~ spine, family = poisson ,
               data = tabla)
summary(log.fit3 )
```

```
Call:
glm(formula = satell ~ spine, family = poisson, data = tabla)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6199  -2.3425  -0.4720   0.7918   5.1429

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.34508    0.13185   10.20   <2e-16 ***
spine       -0.11193    0.05159   -2.17     0.03 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 632.79  on 172  degrees of freedom
Residual deviance: 628.22  on 171  degrees of freedom
AIC: 987.52

Number of Fisher Scoring iterations: 5

p0 = log.fit3$null.deviance - log.fit3$deviance ## calculamos la residual
p0

[1] 4.569933

1 - pchisq(p0,1)

[1] 0.03253784

log.fit4 <- glm(satell ~ color, family = poisson ,
               data = tabla)
summary(log.fit4 )
```

```
Call:
glm(formula = satell ~ color, family = poisson, data = tabla)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9072  -2.2128  -0.2959   0.9189   4.9423

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.98715    0.19933   9.969  < 2e-16 ***
color       -0.27295    0.05932  -4.601  4.2e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 632.79  on 172  degrees of freedom
Residual deviance: 610.66  on 171  degrees of freedom
AIC: 969.96

Number of Fisher Scoring iterations: 6
```

```
p0 = log.fit4$null.deviance - log.fit4$deviance ## calculamos la residual
p0
```

```
[1] 22.1309
```

```
1 - pchisq(p0,1)
```

```
[1] 2.546769e-06
```

```
log.fit5 <- glm(satell ~ width + color + spine + weight, family = poisson ,
               data = tabla)
summary(log.fit5 )
```

```
Call:
glm(formula = satell ~ width + color + spine + weight, family = poisson,
    data = tabla)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0126  -1.8846  -0.5406   0.9448   4.9602

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.3435447  0.9684204  -0.355  0.72278
width        0.0275251  0.0479425   0.574  0.56588
color       -0.1849325  0.0665236  -2.780  0.00544 **
spine        0.0399764  0.0568062   0.704  0.48160
weight       0.0004725  0.0001649   2.865  0.00417 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 632.79  on 172  degrees of freedom
Residual deviance: 551.85  on 168  degrees of freedom
AIC: 917.15

Number of Fisher Scoring iterations: 6
```

```
p0 = log.fit5$null.deviance - log.fit5$deviance ## calculamos la residual
p0
```

```
[1] 80.93836
```

```
1 - pchisq(p0,1)
```

```
[1] 0
```

```r
log.fit6 <- glm(satell ~ color + weight, family = poisson ,
               data = tabla)
summary(log.fit6 )
```

```
Call:
glm(formula = satell ~ color + weight, family = poisson, data = tabla)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9785  -1.9159  -0.5471   0.9181   4.8338

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.614e-01  3.008e-01   0.869  0.38496
color       -1.728e-01  6.155e-02  -2.808  0.00499 **
weight       5.459e-04  6.749e-05   8.088 6.05e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 632.79  on 172  degrees of freedom
Residual deviance: 552.79  on 170  degrees of freedom
AIC: 914.09

Number of Fisher Scoring iterations: 6
```

```r
p0 = log.fit6$null.deviance - log.fit6$deviance ## calculamos la residual
p0
```

```
[1] 79.99794
```

```r
1 - pchisq(p0,1)
```

```
[1] 0
```

Posteriormente realizamos diversos modelos para observar que tan bien se ajustan a nuestros datos, ya que la intención es encontrar un buen modelo nos vamos a apoyar del AIC que nos regreso cada uno de ellos. Con este criterio encontramos que el mejor modelo es cuando tomamos en cuenta las 4 variables(width, color, spine, y weight) pero observamos que solo dos variables parecen ser significativas que son el color y weight. Al realizar el modelo tomando en cuenta estas dos variables mejora un poco el valor del AIC.

Una vez que exploramos los anteriores casos vamos a realizar un nuevo ajuste a los datos para trabajar con la sobredispersión.

```
#### Nuevo ajuste a los datos

t1 <- tabla[order(tabla$width),]

int1 <- t1[t1$width<23.25,]
int2 <- t1[t1$width>23.25 & t1$width<24.25,]
int3 <- t1[t1$width>24.25 & t1$width<25.25,]
int4 <- t1[t1$width>25.25 & t1$width<26.25,]
int5 <- t1[t1$width>26.25 & t1$width<27.25,]
int6 <- t1[t1$width>27.25 & t1$width<28.25,]
int7 <- t1[t1$width>28.25 & t1$width<29.25,]
int8 <- t1[t1$width>29.25,]


s <- c("23.25","23.25-24.25","24.25-25.25","25.25-26.25",
            "26.25-27.25","27.25-28.25","28.25-29.25",">29.25")

casos <- c(length(int1$color), length(int2$color),
               length(int3$color),length(int4$color),
               length(int5$color),length(int6$color),
               length(int7$color),length(int8$color))
```

```r
a_m <- c(mean(int1$width),mean(int2$width),mean(int3$width),
                 mean(int4$width),mean(int5$width),mean(int6$width),
                 mean(int7$width),mean(int8$width))

s_g <- c(sum(int1$satell),sum(int2$satell),sum(int3$satell),
              sum(int4$satell),sum(int5$satell),sum(int6$satell),
              sum(int7$satell),sum(int8$satell))

tasa <- c(sum(int1$satell)/length(int1$color),
              sum(int2$satell)/length(int2$color),
            sum(int3$satell)/length(int3$color),
            sum(int4$satell)/length(int4$color),
            sum(int5$satell)/length(int5$color),
            sum(int6$satell)/length(int6$color),
            sum(int7$satell)/length(int7$color),
            sum(int8$satell)/length(int8$color))

Tabla_final <- data.frame(cbind(s,casos,
                        a_m,s_g,tasa))

## Finalmente obtenemos toda la informaci<U+663C><U+3E33>n para el modelo
S_total <- c(14, 20 ,67, 105, 63, 93, 71, 72)

width <- c(22.6928571428571, 23.8428571428571, 24.775 ,
         25.8384615384615 ,26.7909090909091, 27.7375,
         28.6666666666667 ,30.4071428571429)

lcases <- log(c(14, 14, 28, 39, 22, 24, 18 ,14))

log.fit = glm(S_total ~ width, family = poisson,
              offset=lcases )
summary(log.fit )
```

```
Call:
glm(formula = S_total ~ width, family = poisson, offset = lcases)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5322  -0.7750  -0.3454   0.7151   1.0347

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.54018    0.57658  -6.140 8.26e-10 ***
width        0.17290    0.02125   8.135 4.11e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 72.3772  on 7  degrees of freedom
Residual deviance:  6.5164  on 6  degrees of freedom
AIC: 56.961

Number of Fisher Scoring iterations: 4
```

Finalmente observamos que al realizar dicho ajuste nuesto modelo obtiene un mejor valor de AIC por lo que ya no observamos la sobredispersión.

4. Problema 4

Suponga (x_1, y_1),...,(x_n, y_n) observaciones independientes de variables aleatorias definidas como sigue:

$$Y_i\ Bernoulli(p), \quad i = 1, ..n.$$
$$X_i|Y_i = 1\ N(\mu_1, \sigma^2)$$
$$X_i|Y_i = 0\ N(\mu_0, \sigma^2)$$

Usando el Teorema de Bayes, muestre que $P(Y_i = 1|X_i)$ satisface el modelo de regresión logística, esto es

$$logit(P(Y_i = 1|X_i)) = \alpha + \beta X_i$$

con $\beta = (\mu_1 - \mu_0)/\sigma^2$.

Solución

Podemos partir de los siguiente:

$$\frac{p}{1-p} = \frac{P(y_i = 1|x_i)}{P(y_i = 0|x_i)} = \frac{\frac{P(y_i=1)P(x_i|y_1=1)}{\sum_{i=1}^{n} P(y_i)P(x_i|y_i)}}{\frac{P(y_i=0)P(x_i|y_1=0)}{\sum_{i=1}^{n} P(y_i)P(x_i|y_i)}} = \frac{P(y_i = 1)P(x_i|y_i = 1)}{P(y_i = 0)P(x_i|y_i = 0)}$$
$$= \frac{pP(x_i|y_i = 1)}{(1-p)P(x_i|y_i = 0)} = \frac{p(\frac{1}{\sigma^2\sqrt{2\pi}}e^{\frac{-(x_i-\mu_1)^2}{2\sigma^2}})}{(1-p)(\frac{1}{\sigma^2\sqrt{2\pi}}e^{\frac{-(x_i-\mu_0)^2}{2\sigma^2}})}$$

aplicamos logaritmo a la expresión anterior y tenemos:

$$\begin{aligned}
log(\frac{p}{1-p}) &= log\left[p\left(\frac{1}{\sigma^2\sqrt{2\pi}}e^{\frac{-(x_i-\mu_1)^2}{2\sigma^2}}\right)\right] - log\left[(1-p)\left(\frac{1}{\sigma^2\sqrt{2\pi}}e^{\frac{-(x_i-\mu_0)^2}{2\sigma^2}}\right)\right]\\
&= log\left(\frac{p}{1-p}\right) + log\left(\frac{1}{\sigma^2\sqrt{2\pi}}e^{\frac{-(x_i-\mu_1)^2}{2\sigma^2}}\right) - log\left(\frac{1}{\sigma^2\sqrt{2\pi}}e^{\frac{-(x_i-\mu_0)^2}{2\sigma^2}}\right)\\
&= log\left(\frac{p}{1-p}\right) + log\left(\frac{1}{\sigma^2\sqrt{2\pi}}\right) - \frac{(x_i-\mu_1)^2}{2\sigma^2} - log\left(\frac{1}{\sigma^2\sqrt{2\pi}}\right) + \frac{(x_i-\mu_0)^2}{2\sigma^2}\\
&= log\left(\frac{p}{1-p}\right) - \frac{1}{2}\left(\frac{x_i^2-2x_i\mu_1+\mu_1^2}{\sigma^2}\right) + \frac{1}{2}\left(\frac{x_i^2-2x_i\mu_0+\mu_0^2}{\sigma^2}\right)\\
&= log\left(\frac{p}{1-p}\right) + \frac{x_i\mu_1}{\sigma^2} - \frac{\mu_1^2}{2\sigma^2} - \frac{x_i\mu_0}{\sigma^2} + \frac{\mu_0^2}{2\sigma^2}\\
&= \left[log\left(\frac{p}{1-p}\right) + \frac{1}{2}\left(\frac{\mu_0^2-\mu_1^2}{\sigma^2}\right)\right] + \frac{(\mu_1-\mu_0)}{\sigma^2}x_i\\
&= \alpha + \beta x_i
\end{aligned}$$

donde $\beta = \frac{\mu_1-\mu_0}{\sigma^2}$ y $\alpha = log\left(\frac{p}{1-p}\right) + \frac{1}{2}\left(\frac{\mu_0^2-\mu_1^2}{\sigma^2}\right)$.

5. Problema 5

Cuando usamos un modelo de regresión logística para clasificación, tenemos que definir el umbral, p, a partir del cual declaramos un "positivo". Las curvas ROC grafucan las tasas TPR vs FPR para diferentes umbrales p.

$$TPR = True\ Positive\ Rate = \frac{TP}{P} = \text{"}sensitividad\text{"}$$
$$FPR = False\ Positive\ Rate = \frac{FP}{N} = 1 - \text{"}especificidad\text{"}$$

a) La gráfica de TPR vs FPR puede interpretarse como una gráfica de "poder"vs error tipo I".

b) Idealmente, una regla de decisión estaría en el punto (0,1)

c) El área bajo la curva, AUC, puede verse, es la probabilidad de un individuo de los positivos, tomando al azar, tenga un riesgo estimado mayor que un individuo de los negativos, tomado al azar.

d) El estadístivo J de Youden, es una medida que, con un sólo número, trata de capturar el desempeño de una prueba de diagnóstico. Es la máxima distancia vertical, entre la diagonal y la curva ROC, o equivalentemente: $J = sensitividad - (1 - especifidad)$

Construya la curva ROC para el problema de daño coronario y su relación con la edad visto en la clase 3 del curso.

Solución

En la siguiente sección de código se realizo la implementación para encontrar la curva ROC para el problema de daño coronario.

```
# primero cargamos nuestra libreria
library(ROCR)

edad <- c(20,23,24,25,25,26,26,28,28,29,30,30,30,
          30,30,30,32,32,33,33,34,34,34,34,34,35,
          35,36,36,36,37,37,37,38,38,39,39,40,40,
          41,41,42,42,42,42,43,43,43,44,44,44,44,
          45,45,46,46,47,47,47,48,48,48,49,49,49,
          50,50,51,52,52,53,53,54,55,55,55,56,56,
          56,57,57,57,57,57,57,58,58,58,59,59,60,
          60,61,62,62,63,64,64,65,69)

coro <- c(0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0
          ,0,1,0,0,0,0,0,1,0,0,1,0,0,0,0,1,0,1,0,0,
          0,0,0,1,0,0,1,0,0,1,1,0,1,0,1,0,0,1,0,1,
          1,0,0,1,0,1,0,0,1,1,1,1,0,1,1,1,1,1,0,
```

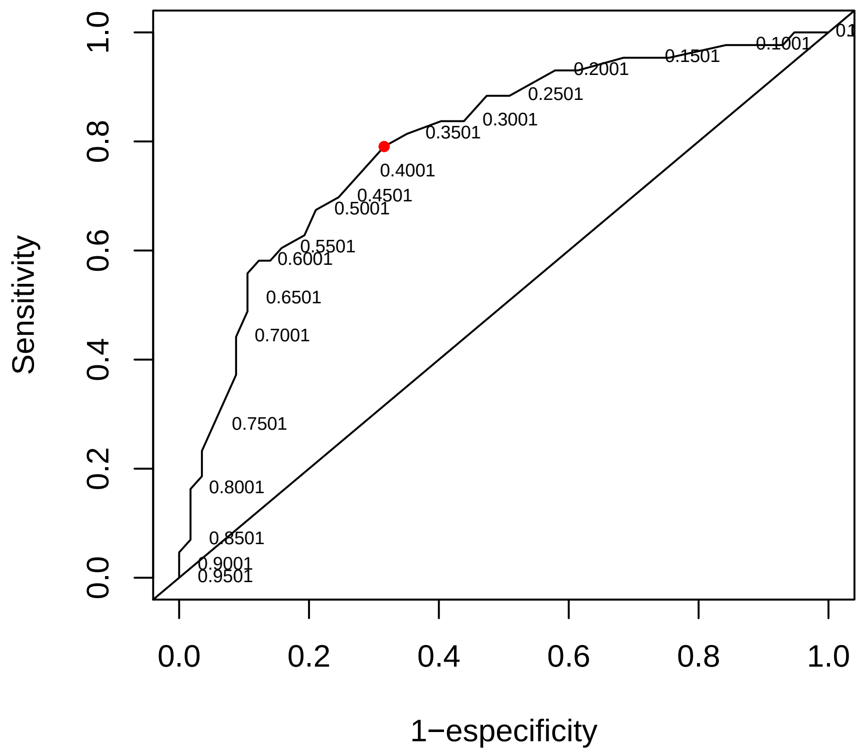
```
          0,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1,0,1,1,1)

Data <- data.frame(edad, coro)

logit_reg <- glm(coro ~ edad, data = Data,
                 family = "binomial")

roc_data <- function(modelo, coro,step) {
  out<-data.frame()
  cut <- seq(step, 1, by = step)
  for (i in cut) {

    predicted_value <- predict(modelo,type = "response")
    predicted_class <- ifelse(predicted_value>i, "1","0")
    performance_data<-data.frame(observed=coro,
                             predicted= predicted_class)


    positive <- sum(performance_data$observed=="1")
    negative <- sum(performance_data$observed=="0")
    predicted_positive <- sum(performance_data$predicted=="1")
    predicted_negative <- sum(performance_data$predicted=="0")
    total <- nrow(performance_data)

    tp<-sum(performance_data$observed=="1"&performance_data$predicted=="1")
    tn<-sum(performance_data$observed=="0"&performance_data$predicted=="0")
    fp<-sum(performance_data$observed=="0"&performance_data$predicted=="1")
    fn<-sum(performance_data$observed=="1"&performance_data$predicted=="0")

    accuracy <- (tp+tn)/total
    error_rate <- (fp+fn)/total
    sensitivity <- tp/positive
    especificity <- tn/negative
```

```r
    precision <- tp/predicted_positive
    npv <- tn / predicted_negative
    fpr <- fn/negative
    J <-  sensitivity - (1-especificity)

    out<-rbind(out,c(i,1-especificity,sensitivity,J,
                     especificity,precision,npv,fpr))
  }

  names(out)<-c("cut","1-Especificity","Sensivity",
                "Youden(J)","Especificity",
                "Precision","npv","fpr")
  return(out)
}


roc_graph <- roc_data( logit_reg, coro, step = 0.0001)
Youden <- roc_graph$`Youden(J)`
max(Youden)
```

```
[1] 0.4749082
```

```r
x<-roc_graph$`1-Especificity`[which.max(Youden)]
y<-roc_graph$Sensivity[which.max(Youden)]

plot(roc_graph$`1-Especificity`,roc_graph$Sensivity,
     xlab = "1-especificity", ylab = "Sensitivity",
     type = "l")
abline(a=0,b=1)
index<-seq(1,nrow(roc_graph), by=nrow(roc_graph)*0.05)
points(x,y, pch=20, col="red")
text(roc_graph$`1-Especificity`[index],
     roc_graph$Sensivity[index],
     labels = roc_graph$cut[index],
```

```
    cex=0.6, pos=4)
```




```
Youden <-roc_graph$`Youden(J)`
max(Youden)

[1] 0.4749082
```

```
#x
#y
which.max(Youden)

[1] 3683

x<-roc_graph$`1-Especificity`[which.max(Youden)]
y<-roc_graph$Sensivity[which.max(Youden)]

x

[1] 0.3157895

y

[1] 0.7906977
```

En el gráfico agregamos el punto optimo donde este nos garantiza que no cometamos un error mayor al tomar un dato de entrada con daño coronario cuando realmente no lo es.
Y encontramos que el valor del estadístico J de Youden es 0.474.

6. Problema 6

La siguiente tabla muestra conteos de células T_4 por mm^3 en muestra de sangra de 20 pacientes (en remisión) con enfermedad de Hodgkin, así como conteos en 20 pacientes en remisión de otras enfermedades. Una cuestión de interés es si existen diferencias en las distribuciones de conteos en ambos grupos.

a) Haga una comparación gráfica exploratoria de estos datos.

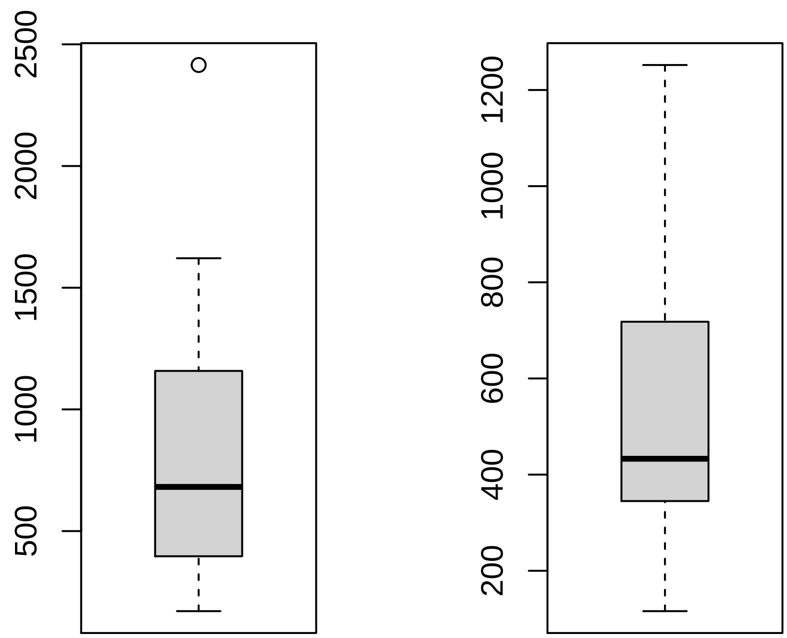
b) Ajuste un modelo de Poisson apropiado.

c) Usando la normalidad asintótica de los estimadores de máxima verosimilitud, dé un intervalo del 90 % de confianza para la diferencia en medias. ¿Hay evidencia de diferencias en los dos grupos en cuanto a las medias de los conteos?

Solución

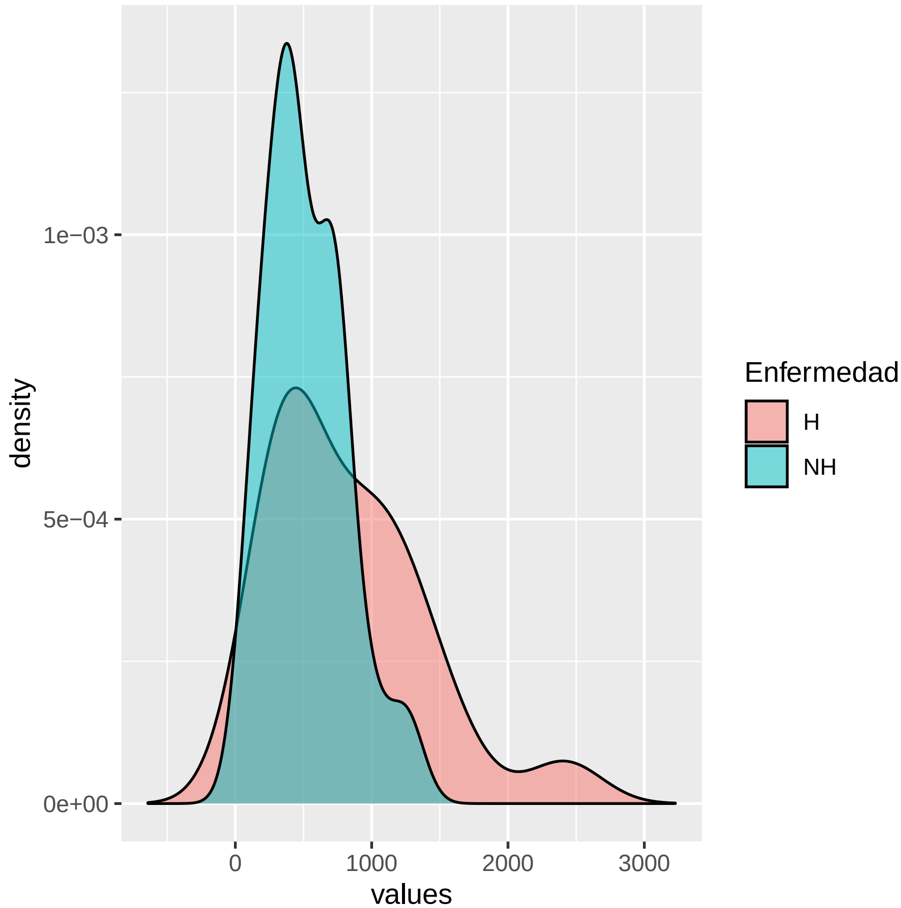
Inciso a)

```r
H <- c(396, 568, 1212, 171, 554, 1104, 257, 435, 295, 397,
          288, 1004, 431, 795, 1621, 1378, 902, 958, 1283,
       2415)

NH <- c(375, 375, 752, 208, 151, 116, 736, 192, 315, 1252,
          675, 700, 440, 771, 688, 426, 410, 979, 377, 503)

# Realizamos nuestro dataframe
dat <- data.frame('H'=H, 'NH'=NH)

## Realizamos las graficas exploratorias de los datos
par(mfrow = c(1, ncol(dat)))
invisible(lapply(1:ncol(dat), function(i) boxplot(dat[, i])))
```



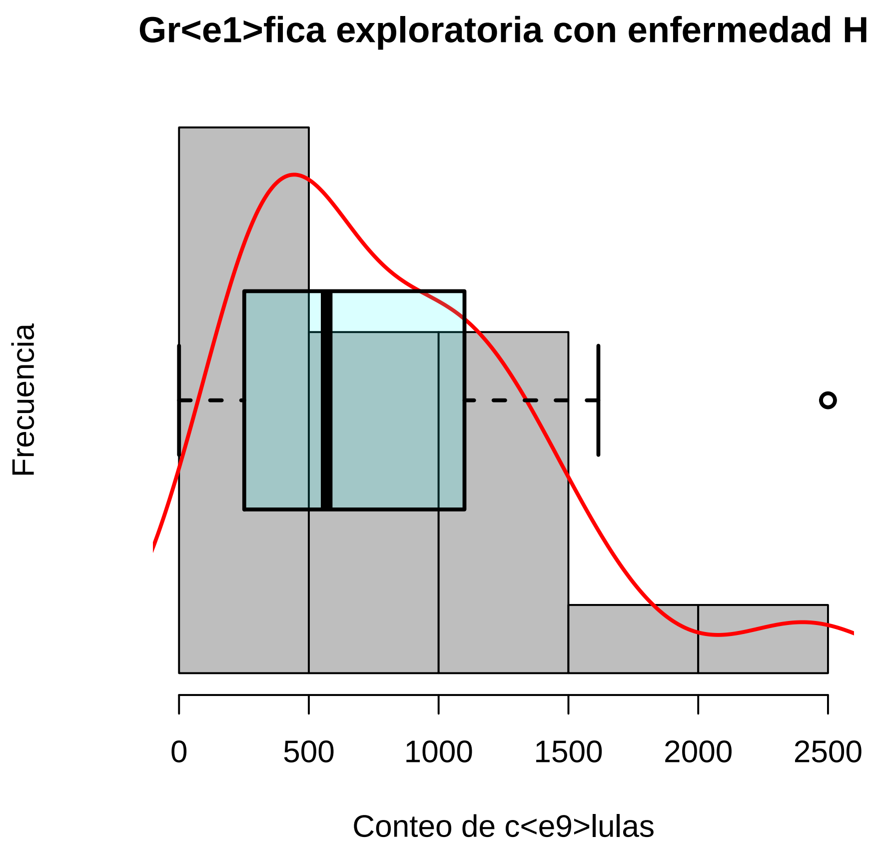

```
par(mfrow = c(1, 1))


###########3
library(ggplot2)
df <- dat
df <- stack(df)
```

```
dx <- density(H)
dy <- density(NH)

ggplot(df, aes(x = values, fill = ind)) +
       geom_density(alpha = 0.5) + # Densidades con transparencia
       xlim(c(min(dx$x, dy$x), # L<U+653C><U+3E64>mites del eje X
            c(max(dx$x, dy$x)))) +
       scale_fill_discrete(name = "Enfermedad", # Cambiar el t<U+653C><U+3E64>tulo d
                           labels = c("H", "NH")) # + # Cambiar las etiquetas de la
```



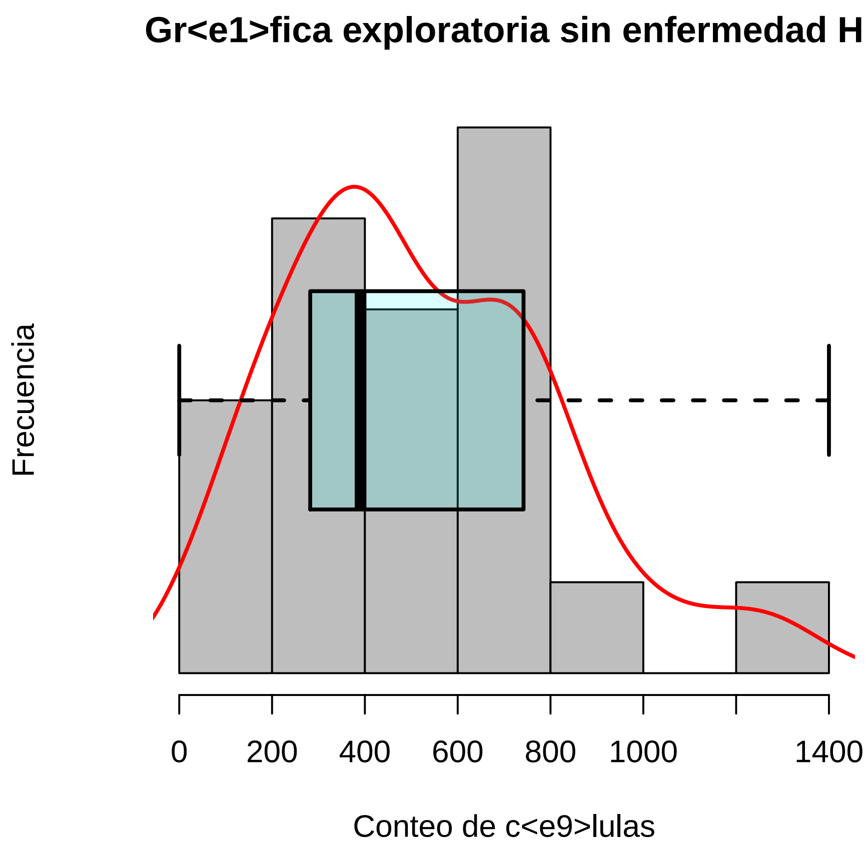

```
    # theme(legend.position = "none") # Eliminar leyenda


############

hist(dat$H, probability = TRUE, ylab = "", col = "grey",xlab='',
     axes = FALSE, main = "")
```

```r
# Eje
axis(1)

# Densidad
lines(density(dat$H), col = "red", lwd = 2)

# Boxplot
par(new = TRUE)
boxplot(dat$H, horizontal = TRUE, axes = FALSE,
        main = "Gr<U+653C><U+3E31>fica exploratoria con enfermedad H",
        xlab = "Conteo de c<U+653C><U+3E39>lulas", ylab = "Frecuencia",lwd = 2,
        col = rgb(0, 1, 1, alpha = 0.15))
```




```
######################
hist(dat$NH, probability = TRUE, ylab = "", xlab='' ,col = "grey",
     axes = FALSE, main = "")

# Eje
axis(1)
```

```r
# Densidad
lines(density(dat$NH), col = "red", lwd = 2)

# Boxplot
par(new = TRUE)
boxplot(dat$NH, horizontal = TRUE, axes = FALSE,
        main = "Gr<U+653C><U+3E31>fica exploratoria sin enfermedad H",
        xlab = "Conteo de c<U+653C><U+3E39>lulas", ylab = "Frecuencia", lwd = 2,
        col = rgb(0, 1, 1, alpha = 0.15))
```



Al realizar los distintos gráficos observamos los valores respecto a la media se parecen ligeramente pero la varianza de dichos datos cambia para cada categoria ya que para los que tienen la enfermendad los datos se encuentran ligeramente más dispersos mientras que para los que no tienen la enfermedad se encuentran más centrados.
Se anexaron distintos gráficos con la intención de ser más ilustrativo con los datos.

Inciso b)

```r
### caso 1
valores_0 <- 1:20
H <- c(396, 568, 1212, 171, 554, 1104, 257, 435, 295, 397,
       288, 1004, 431, 795, 1621, 1378, 902, 958, 1283,
       2415)

NH <- c(375, 375, 752, 208, 151, 116, 736, 192, 315, 1252,
        675, 700, 440, 771, 688, 426, 410, 979, 377, 503)

##### primer modelo
modelo_H <- glm(H ~ valores_0, poisson)
summary(modelo_H)

Call:
glm(formula = H ~ valores_0, family = poisson)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-20.401  -14.007   -4.394    7.273   30.022

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  5.87940    0.01978  297.24   <2e-16 ***
valores_0    0.07146    0.00142   50.31   <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: 6860.2  on 19  degrees of freedom
Residual deviance: 4200.6  on 18  degrees of freedom
```

```
AIC: 4371.1

Number of Fisher Scoring iterations: 5

##### segundo modelo

modelo_NH <- glm(NH ~ valores_0, poisson)
summary(modelo_NH)

Call:
glm(formula = NH ~ valores_0, family = poisson)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-19.056   -9.509   -3.336    7.062   27.719

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  5.96089    0.02181  273.37   <2e-16 ***
valores_0    0.02711    0.00171   15.86   <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: 3104.9  on 19  degrees of freedom
Residual deviance: 2851.5  on 18  degrees of freedom
AIC: 3014.1

Number of Fisher Scoring iterations: 5
```

En este inciso realizamos diversos modelo con la intensión de entender el comportamiento del conteo de celular T_4. Se propusierón modelos separados para cada una de las categorias, una donde esta presente la enfermedad

y otro donde no lo esta presente.
Para el primer modelo observamos que el conteo de celulas de personas con la enfermedad(H) tiene cierta dispersión y esto se justifica en los resultados del modelo ya que el residual deviance es mayor que los grados de libertad.
El segundo modelo sucede lo mismo, observamos que dado los resultados hay una dispersión en el conteo de celulas de las personas sin la enfermedad.
Al observar que nuestros modelos en ambos casos los modelos son significativos pero debido al residual deviance observamos que hay dispersión por lo que no podemos afirmar alguna tendencia en el conteo de celulas al estar presente la enfermedad.

Inciso c)

```
t.test(x = H, y = NH, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = FALSE, conf.level = 0.90)

Welch Two Sample t-test

data:  H and NH
t = 2.112, df = 28.489, p-value = 0.04358
alternative hypothesis: true difference in means is not equal to 0
90 percent confidence interval:
  58.72914 543.57086
sample estimates:
mean of x mean of y
   823.20    522.05
```

Por medio del comando anterior encontramos que si hay diferencia de medias, ya que nuestra hótesis nula es que son iguales se rechaza dicha hótesis. y el intervalo queda de la siguente forma: (823.20, 522.05)

7. Problema 7

Los datos de la tabla en la siguiente hoja son números, n, de pólizas de seguros y los correspondientes números, y, de reclamos (esto es, números de accidentes en los que se pidió el amparo de la póliza). La variable CAR es una condificación de varias clases de carros, $EDAD$ es la edad del titular de la póliza y $DIST$ es el distrito donde vive el titular.

a) Calcule la tasa de reclamos, y/n, para cada categoría y grafique estas tasas contra las diferentes variables para tener una idea de los efectos principales.

b) Use regresión logística para estimar los efectos principales(cada variable tratada como categórica y modelada usando variables indicadoras) así como sus interacciones.

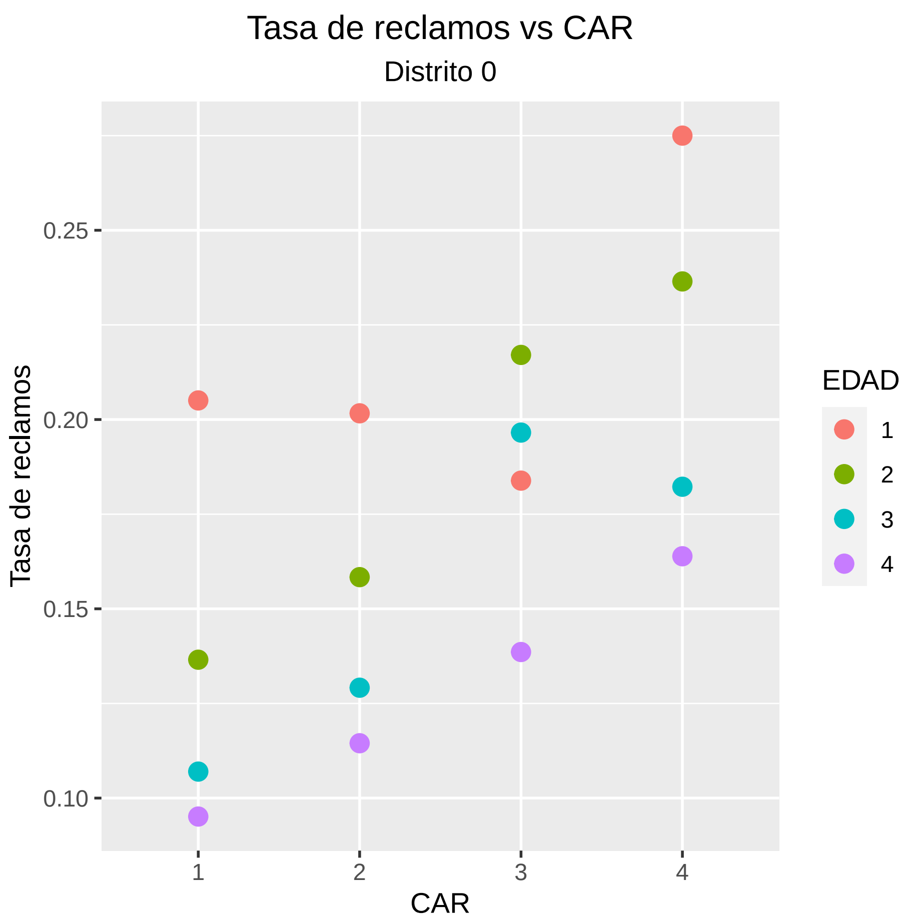
c) Basados en los resultados del inciso anterios, los autores del artículo donde aparecieron y que podían considerar que CAR y $EDAD$ fuesen tratadas como variables continuas. Ajuste un modelo incorporado estas observaciones y compárelo con el obtenido en (b) ¿Cuaáles son las conclusiones?.

Solución

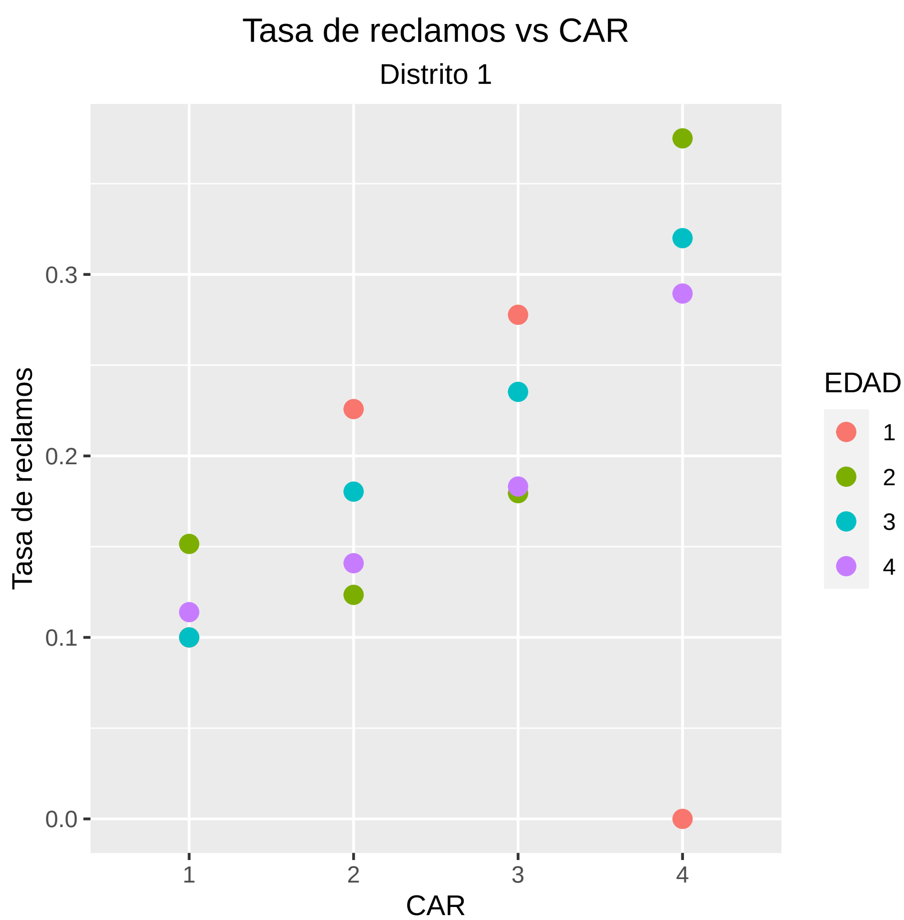
Inciso a)

```
library(ggplot2)
CAR <- rep(c('1','2','3','4'), each = 4)
EDAD <- rep(rep(c('1','2','3','4'), each = 1),4)
y_0 <- c(65,65,52,310,98,159,175,877,41,117,137,477,
         11,35,39,167)
n_0 <- c(317,476,486,3259,486,1004,1355,7660,223,539,
         697,3442,40,148,214,1019)
y_1 <- c(2,5,4,36,7,10,22,102,5,7,16,63,0,6,8,33)
```

```r
n_1 <- c(20,33,40,316,31,81,122,724,18,39,68,344,
         3,16,25,114)

tasa_0 <- y_0/n_0
tasa_1 <- y_1/n_1

dat <- data.frame('CAR'=CAR,'EDAD'=EDAD,'tasa_0'=tasa_0,
                  'tasa_1'=tasa_1)

### Realizamos los primeros gr<U+653C><U+3E31>ficos correspondientes
# CAR vs tasa_0
ggplot(data=dat, aes(x=CAR, y=tasa_0, color=EDAD )) +
  geom_point(size = 3 ,aes(col=EDAD) ) +
  labs(title = 'Tasa de reclamos vs CAR',
       subtitle='Distrito 0' , x='CAR', y='Tasa de reclamos') +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```

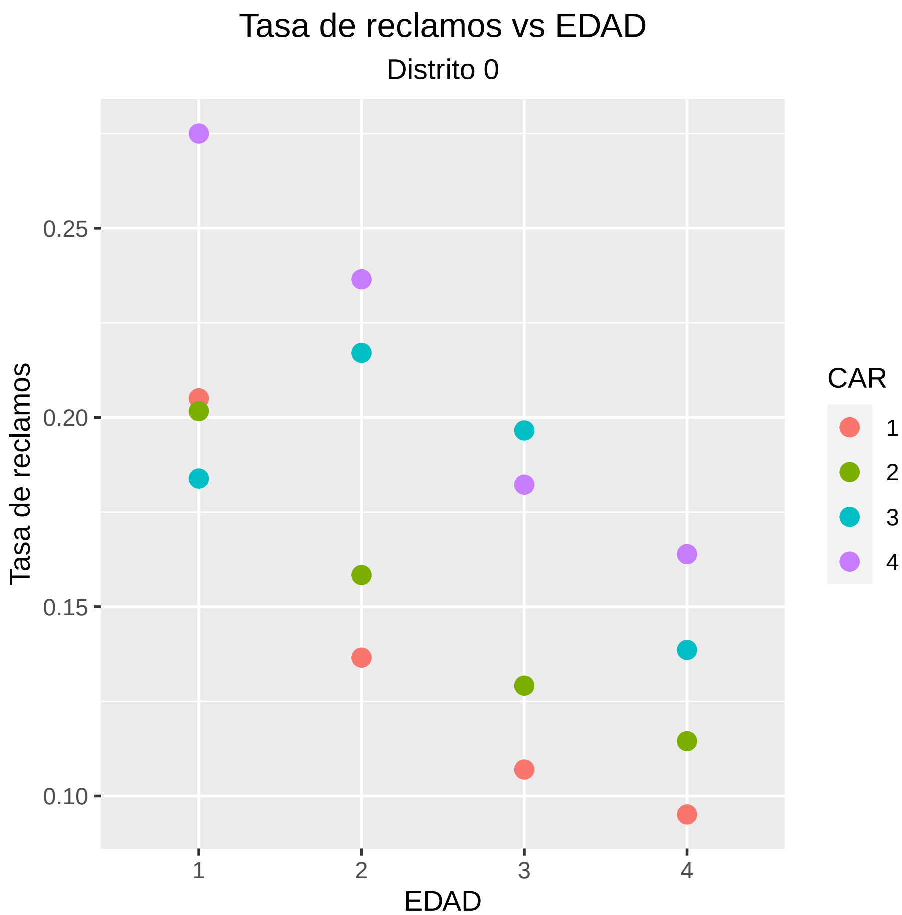



```r
# CAR vs tasa_1
ggplot(data=dat, aes(x=CAR, y=tasa_1, color=EDAD )) +
  geom_point(size = 3 ,aes(col=EDAD) ) +
  labs(title = 'Tasa de reclamos vs CAR',
       subtitle='Distrito 1' , x='CAR', y='Tasa de reclamos') +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```



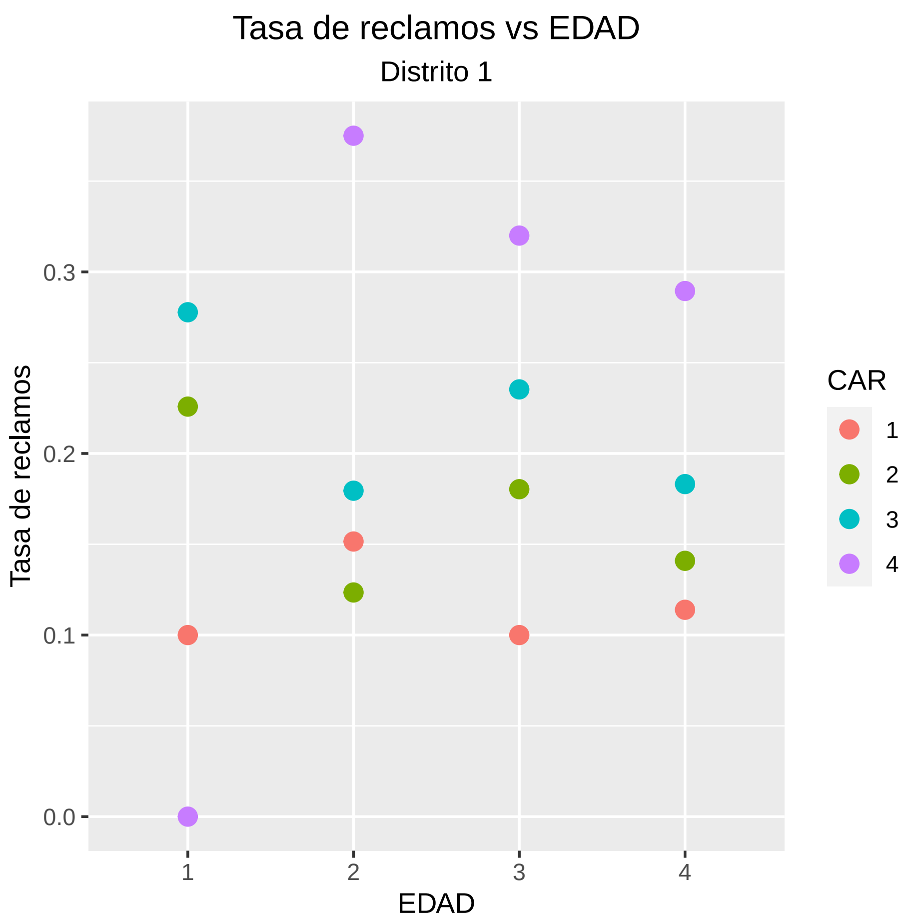

```
#EDAD vs tasa_0
ggplot(data=dat, aes(x=EDAD, y=tasa_0, color=CAR )) +
  geom_point(size = 3 ,aes(col=CAR) ) +
  labs(title = 'Tasa de reclamos vs EDAD',
       subtitle='Distrito 0' , x='EDAD', y='Tasa de reclamos') +
```

```r
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```



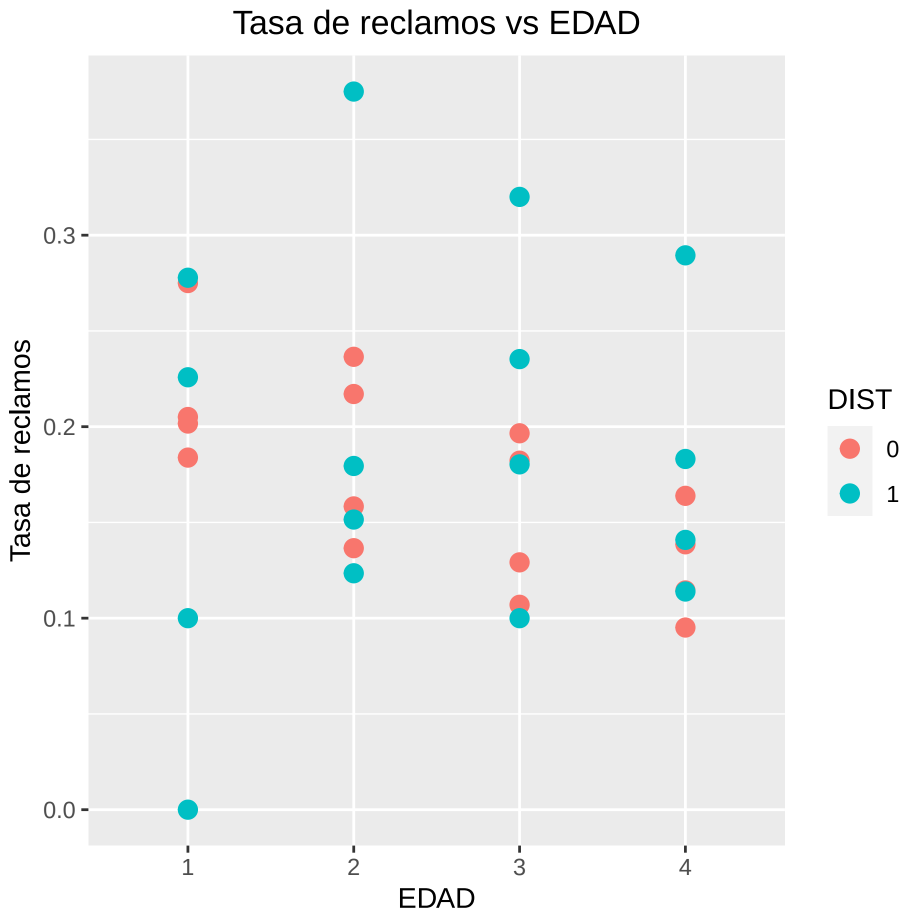

```r
#EDAD vs tasa_1
ggplot(data=dat, aes(x=EDAD, y=tasa_1, color=CAR )) +
  geom_point(size = 3 ,aes(col=CAR) ) +
  labs(title = 'Tasa de reclamos vs EDAD',
```

```
      subtitle='Distrito 1' , x='EDAD', y='Tasa de reclamos') +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```

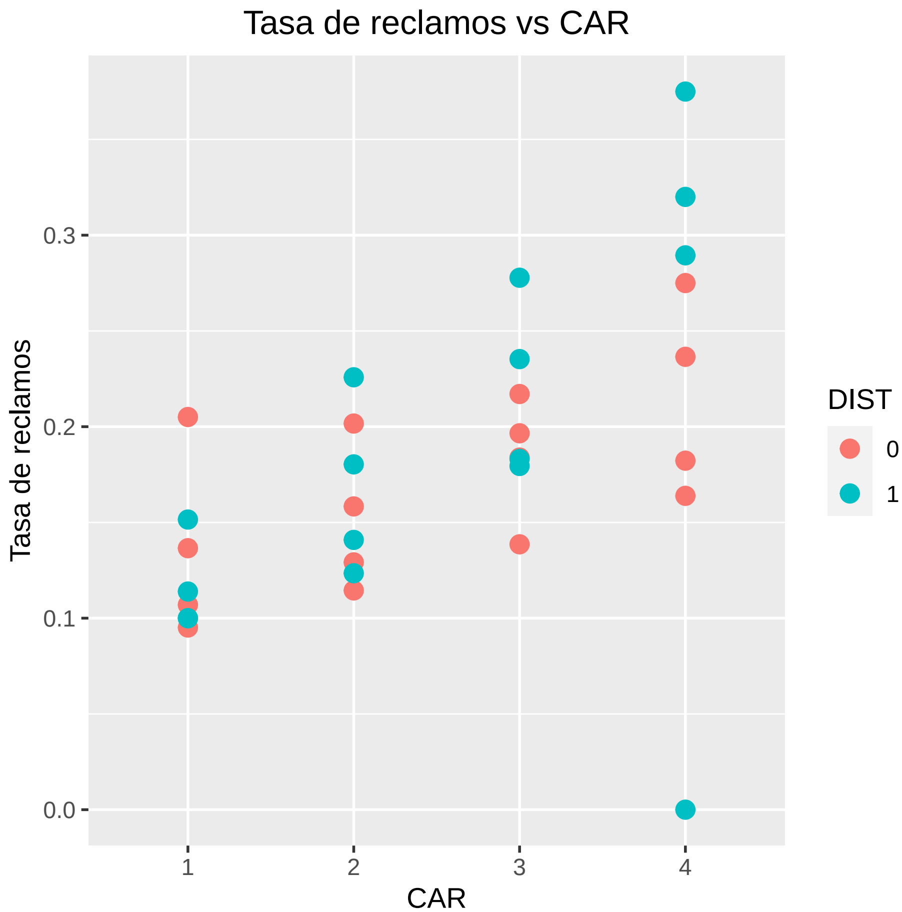



```
#### una con todos los datos

DIST <- rep(c('0','1'), each=16)
```

```r
tasas <- c(tasa_0, tasa_1)
edad_f <- c(EDAD, EDAD)
car_f <- c(CAR, CAR)

dat_2 <- data.frame('CAR'=car_f,'EDAD'=edad_f,'Tasa'=tasas,'DIST'=DIST)

ggplot(data=dat_2, aes(x=EDAD, y=Tasa, color=DIST )) +
  geom_point(size = 3 ,aes(col=DIST) ) +
  labs(title = 'Tasa de reclamos vs EDAD'
       , x='EDAD', y='Tasa de reclamos') +
  theme(plot.title = element_text(hjust = 0.5))
```




```
ggplot(data=dat_2, aes(x=CAR, y=Tasa, color=DIST )) +
  geom_point(size = 3 ,aes(col=DIST) ) +
  labs(title = 'Tasa de reclamos vs CAR'
       , x='CAR', y='Tasa de reclamos') +
  theme(plot.title = element_text(hjust = 0.5))
```



Los gráficos los dividimos en varios casos, los primeros corresponden a gráficos tomando en cuenta solo la variable con algún distrito(Distrito $=$ 0 ó Distrito $=$ 1) y el otro tomando la variable con ambos distritos.
En el gráfico correspondiente a la variable CAR respecto al distrito 0, se observa que la categoria de la variable CAR que tiene más altas tasas es la 4 y tomando como referencia la variable $EDAD$ la categoria que tiene más alto valores es la categoria 1.

En el gráfico donde ahora tomamos el distrito 1 observamos que las tasas son mayores para la categoria 4 de la variable CAR.
En el gráfico de la tasa de reclamos respecto a la EDAD para el distrito 0 nos muestra un comportamiento en descenso ya que empieza con tasas altas y a medida que cambia la categoria disminuye dicha tasa.
En ese mismo gráfico pero tomando ahora el distrito 1 ya no se observa un patrón, pero si hacemos diferencia respecto a la variable CAR nuestro valores son grandes para la categoria 4. Ya al final tomando ambos distritos para cada variable, observamos para la variable EDAD cierto comportamiento decreciente y algunos valores altos que no son muchos. Para la variable CAR es distinto el escenario ya que los valores van en forma creciente y en especifico se observa que los valores altos en su mayoria provienen del distrito 0.

Inciso b)

```
library(MASS)
library(tidyverse)

y_0 <- c(65,2,65,5,52,4,310,36,98,7,159,10,175,22,877,102,41,5,117,7
      ,137,16,477,63,11,0,35,6,39,8,167,33)

n_0 <- c(317,20,476,33,486,40,3259,316,486,31,1004,81,1355,122,7660,724,
      223,28,539,39,697,68,3442,344,40,3,148,16,214,25,1019,114)

dat <- data.frame(expand.grid(
DIST = factor( c("0","1"),levels=c("1","0") ),
EDAD = factor( c('1','2','3','4'),levels=c('4','3','2','1') ),
CAR = factor( c('1','2','3','4'),levels=c('4','3','2','1') )),
frec = c(0.20504,0.1,0.1365, 0.1515, 0.1069, 0.1, 0.0951, 0.1139,
         0.2016, 0.2258, 0.1583, 0.1234, 0.12915, 0.18032, 0.1144,
         0.14088, 0.18385, 0.2777, 0.21706, 0.17948, 0.19655, 0.2352,
         0.1385, 0.1831, 0.2750, 0.00, 0.2364, 0.3750, 0.1822, 0.3200,
```

```
         0.1638, 0.2894),
y = c(65,2,65,5,52,4,310,36,98,7,159,10,175,22,877,102,41,5,117,7
      ,137,16,477,63,11,0,35,6,39,8,167,33),
n = c(317,20,476,33,486,40,3259,316,486,31,1004,81,1355,122,7660,724,
      223,28,539,39,697,68,3442,344,40,3,148,16,214,25,1019,114),
n_y = n_0-y_0)


dat


   DIST EDAD CAR    frec   y    n  n_y
1     0    1   1 0.20504  65  317  252
2     1    1   1 0.10000   2   20   18
3     0    2   1 0.13650  65  476  411
4     1    2   1 0.15150   5   33   28
5     0    3   1 0.10690  52  486  434
6     1    3   1 0.10000   4   40   36
7     0    4   1 0.09510 310 3259 2949
8     1    4   1 0.11390  36  316  280
9     0    1   2 0.20160  98  486  388
10    1    1   2 0.22580   7   31   24
11    0    2   2 0.15830 159 1004  845
12    1    2   2 0.12340  10   81   71
13    0    3   2 0.12915 175 1355 1180
14    1    3   2 0.18032  22  122  100
15    0    4   2 0.11440 877 7660 6783
16    1    4   2 0.14088 102  724  622
17    0    1   3 0.18385  41  223  182
18    1    1   3 0.27770   5   28   23
19    0    2   3 0.21706 117  539  422
20    1    2   3 0.17948   7   39   32
21    0    3   3 0.19655 137  697  560
22    1    3   3 0.23520  16   68   52
```

```
23     0    4   3 0.13850 477 3442 2965
24     1    4   3 0.18310  63  344  281
25     0    1   4 0.27500  11   40   29
26     1    1   4 0.00000   0    3    3
27     0    2   4 0.23640  35  148  113
28     1    2   4 0.37500   6   16   10
29     0    3   4 0.18220  39  214  175
30     1    3   4 0.32000   8   25   17
31     0    4   4 0.16380 167 1019  852
32     1    4   4 0.28940  33  114   81
```

```r
dat %>%
  mutate(EDAD = paste("EDAD", EDAD, sep = "_"),
         valor_EDAD = 1)
```

```
   DIST   EDAD CAR    frec   y    n  n_y valor_EDAD
1     0 EDAD_1   1 0.20504  65  317  252          1
2     1 EDAD_1   1 0.10000   2   20   18          1
3     0 EDAD_2   1 0.13650  65  476  411          1
4     1 EDAD_2   1 0.15150   5   33   28          1
5     0 EDAD_3   1 0.10690  52  486  434          1
6     1 EDAD_3   1 0.10000   4   40   36          1
7     0 EDAD_4   1 0.09510 310 3259 2949          1
8     1 EDAD_4   1 0.11390  36  316  280          1
9     0 EDAD_1   2 0.20160  98  486  388          1
10    1 EDAD_1   2 0.22580   7   31   24          1
11    0 EDAD_2   2 0.15830 159 1004  845          1
12    1 EDAD_2   2 0.12340  10   81   71          1
13    0 EDAD_3   2 0.12915 175 1355 1180          1
14    1 EDAD_3   2 0.18032  22  122  100          1
15    0 EDAD_4   2 0.11440 877 7660 6783          1
16    1 EDAD_4   2 0.14088 102  724  622          1
17    0 EDAD_1   3 0.18385  41  223  182          1
18    1 EDAD_1   3 0.27770   5   28   23          1
```

```
19     0 EDAD_2   3 0.21706 117  539  422          1
20     1 EDAD_2   3 0.17948   7   39   32          1
21     0 EDAD_3   3 0.19655 137  697  560          1
22     1 EDAD_3   3 0.23520  16   68   52          1
23     0 EDAD_4   3 0.13850 477 3442 2965          1
24     1 EDAD_4   3 0.18310  63  344  281          1
25     0 EDAD_1   4 0.27500  11   40   29          1
26     1 EDAD_1   4 0.00000   0    3    3          1
27     0 EDAD_2   4 0.23640  35  148  113          1
28     1 EDAD_2   4 0.37500   6   16   10          1
29     0 EDAD_3   4 0.18220  39  214  175          1
30     1 EDAD_3   4 0.32000   8   25   17          1
31     0 EDAD_4   4 0.16380 167 1019  852          1
32     1 EDAD_4   4 0.28940  33  114   81          1

m <- dat %>%
  mutate(CAR = paste("CAR", CAR, sep = "_"),
         valor_CAR = 1,
         EDAD = paste("EDAD", EDAD, sep = "_"),
         valor_EDAD = 1
         )%>%
  spread(key = CAR, value = valor_CAR, fill = 0)%>%
  spread(key = EDAD, value = valor_EDAD, fill = 0)

m

   DIST    frec   y    n  n_y CAR_1 CAR_2 CAR_3 CAR_4 EDAD_1 EDAD_2 EDAD_3
1     1 0.00000   0    3    3     0     0     0     1      1      0      0
2     1 0.10000   2   20   18     1     0     0     0      1      0      0
3     1 0.10000   4   40   36     1     0     0     0      0      0      1
4     1 0.11390  36  316  280     1     0     0     0      0      0      0
5     1 0.12340  10   81   71     0     1     0     0      0      1      0
6     1 0.14088 102  724  622     0     1     0     0      0      0      0
7     1 0.15150   5   33   28     1     0     0     0      0      1      0
```

```
8     1 0.17948   7   39   32      0     0     1     0      0      1      0
9     1 0.18032  22  122  100      0     1     0     0      0      0      1
10    1 0.18310  63  344  281      0     0     1     0      0      0      0
11    1 0.22580   7   31   24      0     1     0     0      1      0      0
12    1 0.23520  16   68   52      0     0     1     0      0      0      1
13    1 0.27770   5   28   23      0     0     1     0      1      0      0
14    1 0.28940  33  114   81      0     0     0     1      0      0      0
15    1 0.32000   8   25   17      0     0     0     1      0      0      1
16    1 0.37500   6   16   10      0     0     0     1      0      1      0
17    0 0.09510 310 3259 2949      1     0     0     0      0      0      0
18    0 0.10690  52  486  434      1     0     0     0      0      0      1
19    0 0.11440 877 7660 6783      0     1     0     0      0      0      0
20    0 0.12915 175 1355 1180      0     1     0     0      0      0      1
21    0 0.13650  65  476  411      1     0     0     0      0      1      0
22    0 0.13850 477 3442 2965      0     0     1     0      0      0      0
23    0 0.15830 159 1004  845      0     1     0     0      0      1      0
24    0 0.16380 167 1019  852      0     0     0     1      0      0      0
25    0 0.18220  39  214  175      0     0     0     1      0      0      1
26    0 0.18385  41  223  182      0     0     1     0      1      0      0
27    0 0.19655 137  697  560      0     0     1     0      0      0      1
28    0 0.20160  98  486  388      0     1     0     0      1      0      0
29    0 0.20504  65  317  252      1     0     0     0      1      0      0
30    0 0.21706 117  539  422      0     0     1     0      0      1      0
31    0 0.23640  35  148  113      0     0     0     1      0      1      0
32    0 0.27500  11   40   29      0     0     0     1      1      0      0
   EDAD_4
1       0
2       0
3       0
4       1
5       0
6       1
7       0
```

```
8       0
9       0
10      1
11      0
12      0
13      0
14      1
15      0
16      0
17      1
18      0
19      1
20      0
21      0
22      1
23      0
24      1
25      0
26      0
27      0
28      0
29      0
30      0
31      0
32      0

##### Realizamos un primer analisis solo tomando en cuenta la
## la variable CAR y DIST pero con todos los datos

mm_1 <- glm(cbind(SI=y, NO=n_y) ~ DIST + CAR_1 + CAR_2 + CAR_3,
            data = m, family = binomial(link = logit))
summary(mm_1)

Call:
```

```
glm(formula = cbind(SI = y, NO = n_y) ~ DIST + CAR_1 + CAR_2 +
    CAR_3, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0876  -0.6404   0.4438   1.4216   5.0969

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.24553    0.08572 -14.529  < 2e-16 ***
DIST0       -0.23354    0.06398  -3.650 0.000262 ***
CAR_1       -0.64337    0.07882  -8.162 3.30e-16 ***
CAR_2       -0.47437    0.07014  -6.763 1.35e-11 ***
CAR_3       -0.19843    0.07423  -2.673 0.007516 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.22  on 31  degrees of freedom
Residual deviance: 127.25  on 27  degrees of freedom
AIC: 299.3

Number of Fisher Scoring iterations: 4
```

```
mm_2 <- glm(cbind(SI=y, NO=n_y) ~ DIST + EDAD_1 + EDAD_2 + EDAD_3,
            data = m, family = binomial(link = logit))
summary(mm_2)
```

```
Call:
glm(formula = cbind(SI = y, NO = n_y) ~ DIST + EDAD_1 + EDAD_2 +
    EDAD_3, family = binomial(link = logit), data = m)

Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-4.4617  -1.4236  -0.3305   1.5437   4.1465

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.73738    0.06188 -28.078  < 2e-16 ***
DIST0       -0.25799    0.06398  -4.033 5.52e-05 ***
EDAD_1       0.58605    0.07755   7.557 4.12e-14 ***
EDAD_2       0.41033    0.05957   6.888 5.67e-12 ***
EDAD_3       0.24213    0.05615   4.312 1.62e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.22  on 31  degrees of freedom
Residual deviance: 133.58  on 27  degrees of freedom
AIC: 305.62

Number of Fisher Scoring iterations: 4

#### Ahora realizamos los mismos modelos pero por separado
D_t1 <- m[1:16,]
D_t0 <- m[17:32,]
## CAR: modelos de car
modelo_CAR_1 <- glm(cbind(SI=y, NO=n_y) ~ CAR_1 + CAR_2 + CAR_3,
                    data = D_t1,family = binomial(link = logit))
summary(modelo_CAR_1)

Call:
glm(formula = cbind(SI = y, NO = n_y) ~ CAR_1 + CAR_2 + CAR_3,
    family = binomial(link = logit), data = D_t1)

Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-1.4555  -0.3075  -0.1617   0.6403   1.1616

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.8594     0.1740  -4.938 7.88e-07 ***
CAR_1        -1.1821     0.2331  -5.072 3.94e-07 ***
CAR_2        -0.8975     0.1965  -4.568 4.92e-06 ***
CAR_3        -0.5908     0.2094  -2.821  0.00479 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 36.6853  on 15  degrees of freedom
Residual deviance:  7.1799  on 12  degrees of freedom
AIC: 76.88

Number of Fisher Scoring iterations: 4
```

```
modelo_CAR_0 <- glm(cbind(SI=y, NO=n_y) ~ CAR_1 + CAR_2 + CAR_3,
                   data = D_t0,family = binomial(link = logit))
summary(modelo_CAR_0)
```

```
Call:
glm(formula = cbind(SI = y, NO = n_y) ~ CAR_1 + CAR_2 + CAR_3,
    family = binomial(link = logit), data = D_t0)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1023  -0.3598   1.2903   2.8393   5.0026

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
```

```
(Intercept) -1.53447    0.06945 -22.094  < 2e-16 ***
CAR_1       -0.57253    0.08428  -6.793 1.10e-11 ***
CAR_2       -0.41503    0.07547  -5.499 3.82e-08 ***
CAR_3       -0.14233    0.07976  -1.785   0.0743 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 191.67  on 15  degrees of freedom
Residual deviance: 113.48  on 12  degrees of freedom
AIC: 221.83

Number of Fisher Scoring iterations: 4
```

```
## EDAD: modelos de la edad
modelo_EDA_1 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + EDAD_3,
                    data = D_t1,
                    family = binomial(link = logit))
summary(modelo_EDA_1)
```

```
Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + EDAD_3,
    family = binomial(link = logit), data = D_t1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1575  -1.0603  -0.0559   0.9333   3.5869

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.68672    0.07117 -23.701   <2e-16 ***
EDAD_1       0.10627    0.30199   0.352    0.725
EDAD_2       0.07016    0.21879   0.321    0.748
```

```
EDAD_3       0.27573    0.17304   1.593    0.111
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 36.685  on 15  degrees of freedom
Residual deviance: 34.186  on 12  degrees of freedom
AIC: 103.89

Number of Fisher Scoring iterations: 4
```

```
modelo_EDAD_0 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + EDAD_3,
                    data = D_t0,family = binomial(link = logit))
summary(modelo_EDAD_0)
```

```
Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + EDAD_3,
    family = binomial(link = logit), data = D_t0)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.3526  -1.4196   0.0732   2.1028   4.2125

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.00145    0.02490 -80.383  < 2e-16 ***
EDAD_1       0.62568    0.08029   7.793 6.55e-15 ***
EDAD_2       0.44051    0.06195   7.111 1.15e-12 ***
EDAD_3       0.23864    0.05939   4.018 5.86e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 191.665  on 15  degrees of freedom
Residual deviance:  93.716  on 12  degrees of freedom
AIC: 202.07

Number of Fisher Scoring iterations: 4

##########################   Posibles
########################## Combinaciones

modelo_1 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + EDAD_3 + CAR_1,
                data = m,
                family = binomial(link = logit))
summary(modelo_1)

Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + EDAD_3 +
    CAR_1, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.8513  -0.6950   0.4398   1.7072   4.5118

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.90986    0.02510 -76.102  < 2e-16 ***
EDAD_1       0.60641    0.07772   7.802 6.08e-15 ***
EDAD_2       0.40801    0.05960   6.846 7.60e-12 ***
EDAD_3       0.23044    0.05620   4.101 4.12e-05 ***
CAR_1       -0.31422    0.05049  -6.223 4.87e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 242.22  on 31  degrees of freedom
Residual deviance: 108.34  on 27  degrees of freedom
AIC: 280.39

Number of Fisher Scoring iterations: 4

modelo_2 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + EDAD_3 + CAR_1 + CAR_2,
                data = m, family = binomial(link = logit))
summary(modelo_2)

Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + EDAD_3 +
    CAR_1 + CAR_2, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3008  -0.5930   0.2703   1.0413   3.7170

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.71752    0.03512 -48.904  < 2e-16 ***
EDAD_1       0.60957    0.07784   7.831 4.83e-15 ***
EDAD_2       0.39937    0.05970   6.690 2.24e-11 ***
EDAD_3       0.22134    0.05629   3.932 8.42e-05 ***
CAR_1       -0.50471    0.05606  -9.003  < 2e-16 ***
CAR_2       -0.32001    0.04280  -7.476 7.67e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.22  on 31  degrees of freedom
Residual deviance:  53.03  on 26  degrees of freedom
```

```
AIC: 227.08

Number of Fisher Scoring iterations: 4

modelo_3 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + EDAD_3 + CAR_1 + CAR_2 + CAR
                data = m, family = binomial(link = logit))
summary(modelo_3)

Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + EDAD_3 +
    CAR_1 + CAR_2 + CAR_3, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0171  -0.8352  -0.1288   1.2140   3.0177

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.55400    0.06577 -23.626  < 2e-16 ***
EDAD_1       0.61551    0.07787   7.904 2.69e-15 ***
EDAD_2       0.40031    0.05971   6.704 2.03e-11 ***
EDAD_3       0.22071    0.05631   3.920 8.86e-05 ***
CAR_1       -0.66886    0.07914  -8.451  < 2e-16 ***
CAR_2       -0.48393    0.07031  -6.883 5.87e-12 ***
CAR_3       -0.21509    0.07441  -2.891  0.00384 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.219  on 31  degrees of freedom
Residual deviance:  44.861  on 25  degrees of freedom
AIC: 220.91
```

```
Number of Fisher Scoring iterations: 4

modelo_4 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + CAR_1 + CAR_2 + CAR_3,
              data = m, family = binomial(link = logit))
summary(modelo_4)

Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + CAR_1 +
    CAR_2 + CAR_3, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0370  -0.8063   0.1422   1.3086   3.2706

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.51404    0.06486 -23.344  < 2e-16 ***
EDAD_1       0.57989    0.07726   7.505 6.13e-14 ***
EDAD_2       0.36412    0.05890   6.182 6.32e-10 ***
CAR_1       -0.67705    0.07908  -8.561  < 2e-16 ***
CAR_2       -0.48832    0.07027  -6.949 3.68e-12 ***
CAR_3       -0.21621    0.07438  -2.907  0.00365 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.219  on 31  degrees of freedom
Residual deviance:  59.715  on 26  degrees of freedom
AIC: 233.76

Number of Fisher Scoring iterations: 4

modelo_5 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + CAR_1 + CAR_2 + CAR_3,
```

```
                data = m, family = binomial(link = logit))
summary(modelo_5)

Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + CAR_1 + CAR_2 +
    CAR_3, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9365  -0.3269   0.4838   1.7067   3.6983

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.47116    0.06434 -22.865  < 2e-16 ***
EDAD_1       0.53657    0.07687   6.981 2.94e-12 ***
CAR_1       -0.67455    0.07901  -8.538  < 2e-16 ***
CAR_2       -0.49029    0.07020  -6.984 2.86e-12 ***
CAR_3       -0.21355    0.07430  -2.874  0.00405 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.219  on 31  degrees of freedom
Residual deviance:  95.719  on 27  degrees of freedom
AIC: 267.77

Number of Fisher Scoring iterations: 4

modelo_6 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + EDAD_2 + CAR_1 + CAR_2,
                data = m, family = binomial(link = logit))
summary(modelo_6)

Call:
```

```
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + EDAD_2 + CAR_1 +
    CAR_2, family = binomial(link = logit), data = m)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0709  -0.5862   0.4527   1.3386   3.5513

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.67830    0.03348 -50.134  < 2e-16 ***
EDAD_1       0.57380    0.07723   7.430 1.09e-13 ***
EDAD_2       0.36307    0.05888   6.166 7.00e-10 ***
CAR_1       -0.51206    0.05601  -9.142  < 2e-16 ***
CAR_2       -0.32354    0.04278  -7.563 3.95e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.219  on 31  degrees of freedom
Residual deviance:  67.975  on 27  degrees of freedom
AIC: 240.02

Number of Fisher Scoring iterations: 4
```

```
modelo_7 <- glm(cbind(SI=y, NO=n_y) ~ EDAD_1 + CAR_2,
                data = m, family = binomial(link = logit))
summary(modelo_7)
```

```
Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD_1 + CAR_2, family = binomial(link = log
    data = m)

Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-7.6394  -0.4919   0.6587   2.2377   4.8962

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.82233    0.02678 -68.037  < 2e-16 ***
EDAD_1       0.49279    0.07652   6.440 1.19e-10 ***
CAR_2       -0.13636    0.03845  -3.546 0.000391 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.22  on 31  degrees of freedom
Residual deviance: 190.83  on 29  degrees of freedom
AIC: 358.88

Number of Fisher Scoring iterations: 4
```

En cada uno de los modelos observamos valores altos para el AIC, dicho valor del cual nos vamos a apoyar para hacer la diferencia entre los distintos modelos.
Los primeros modelos tomando solo una variable CAR ó $EDAD$ nos dicen que cada categoria son significativas pero para el modelo donde tomamos la variable CAR nos dice que la que tiene un mayor peso en el pronostico es la categoria 1 y en el modelo donde tomamos la $EDAD$ nos dice que la categoria con mayor peso al momento de realizar el pronostico es la categoria 1.
Seguido de estos modelos realizamos los mismo pero ahora dividiendo nuestros datos, es decir solo realizar el pronostico tomando en cuenta un distrito. En estos modelos el que mejor nos dio el AIC fue cuando solo se toma el distrito 1 y la variable CAR, los demás nos arrojaron valores altos de AIC.
Finalmente realizamos posibles combinaciones entre las distintas catego-

rias que tenemos, los valores de acuerdo al AIC no fueron menores a las 200 unidades por lo que ninguno se tomo el mejor. Los resultados de cada uno de ellos se muestra así como la combinación que se tomo.

Inciso c)

```
dat_2 <- data.frame(expand.grid(
DIST = factor( c("0","1"), levels=c("1","0") ),
EDAD = factor( c('1','2','3','4'), levels=c('4','3','2','1') ),
CAR = factor( c('1','2','3','4'), levels=c('4','3','2','1') )),
frec = c(0.20504,0.1,0.1365, 0.1515, 0.1069, 0.1, 0.0951, 0.1139,
         0.2016, 0.2258, 0.1583, 0.1234, 0.12915, 0.18032, 0.1144,
         0.14088, 0.18385, 0.2777, 0.21706, 0.17948, 0.19655, 0.2352,
         0.1385, 0.1831, 0.2750, 0.00, 0.2364, 0.3750, 0.1822, 0.3200,
         0.1638, 0.2894),
y = c(65,2,65,5,52,4,310,36,98,7,159,10,175,22,877,102,41,5,117,7
      ,137,16,477,63,11,0,35,6,39,8,167,33),
n = c(317,20,476,33,486,40,3259,316,486,31,1004,81,1355,122,7660,724,
      223,28,539,39,697,68,3442,344,40,3,148,16,214,25,1019,114),
n_y = n_0-y_0)

dat_2

   DIST EDAD CAR    frec   y    n  n_y
1     0    1   1 0.20504  65  317  252
2     1    1   1 0.10000   2   20   18
3     0    2   1 0.13650  65  476  411
4     1    2   1 0.15150   5   33   28
5     0    3   1 0.10690  52  486  434
6     1    3   1 0.10000   4   40   36
7     0    4   1 0.09510 310 3259 2949
8     1    4   1 0.11390  36  316  280
9     0    1   2 0.20160  98  486  388
```

```
10    1    1    2 0.22580   7    31   24
11    0    2    2 0.15830 159 1004  845
12    1    2    2 0.12340  10    81   71
13    0    3    2 0.12915 175 1355 1180
14    1    3    2 0.18032  22   122  100
15    0    4    2 0.11440 877 7660 6783
16    1    4    2 0.14088 102   724  622
17    0    1    3 0.18385  41   223  182
18    1    1    3 0.27770   5    28   23
19    0    2    3 0.21706 117   539  422
20    1    2    3 0.17948   7    39   32
21    0    3    3 0.19655 137   697  560
22    1    3    3 0.23520  16    68   52
23    0    4    3 0.13850 477 3442 2965
24    1    4    3 0.18310  63   344  281
25    0    1    4 0.27500  11    40   29
26    1    1    4 0.00000   0     3    3
27    0    2    4 0.23640  35   148  113
28    1    2    4 0.37500   6    16   10
29    0    3    4 0.18220  39   214  175
30    1    3    4 0.32000   8    25   17
31    0    4    4 0.16380 167 1019  852
32    1    4    4 0.28940  33   114   81
```

```
modelo_con <- glm(cbind(SI=y, NO=n_y) ~ EDAD + CAR,
                data = dat_2, family = binomial(link = logit))
summary(modelo_con)
```

```
Call:
glm(formula = cbind(SI = y, NO = n_y) ~ EDAD + CAR, family = binomial(link = logit),
    data = dat_2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
```

```
-2.0171  -0.8352  -0.1288   1.2140   3.0177

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.55400    0.06577 -23.626  < 2e-16 ***
EDAD3        0.22071    0.05631   3.920 8.86e-05 ***
EDAD2        0.40031    0.05971   6.704 2.03e-11 ***
EDAD1        0.61551    0.07787   7.904 2.69e-15 ***
CAR3        -0.21509    0.07441  -2.891  0.00384 **
CAR2        -0.48393    0.07031  -6.883 5.87e-12 ***
CAR1        -0.66886    0.07914  -8.451  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 242.219  on 31  degrees of freedom
Residual deviance:  44.861  on 25  degrees of freedom
AIC: 220.91

Number of Fisher Scoring iterations: 4
```

Finalmente nos dicen que observemos que sucede al tomar las variables como continuas, en dicho proceso los resultados nos dicen que el AIC fue cerca de 220 por lo que comparado con las demás interacciones no hay mucha diferencia, en este caso coincide con el más bajo de los valores AIC del inciso b) por lo que en conclusión podemos decir que las interacciones ninguna interacción es importante o tiene un mayor impacto que tratar a las variables de forma continua.

8. Problema 8

A lo largo del curso hemos enfatizado el uso del método de Máxima Verosimilitud para todo lo relacionado con estimación. Consideremos ahora una alternativa: El método de la Mínima Ji-cuadrada. Suponga que las celdad de una multinomial están parametrizadas en términos de un vector $\theta = (\theta_1, ..., \theta_s)^t$. El método de la Mínima Ji-cuadrada consiste en estimar θ mediante aquel valor que minimice el estadístico de Pearson.

$$x^2 = \sum \frac{(obs - esp)^2}{esp} = \sum \frac{(y_j - n\pi_j(\theta))^2}{n\pi_j(\theta)}$$

Considere el siguiente problema. Suponga una población muy grande de objetos que pueden clasificarse en tres categorías, A, B y C. Para estimar las proporciones π_1, π_2 y π_3 correspondientes a cada una de esas categorías, se efectuó un estudio; se obtuvieron tres muestras de tamaños n_1, n_2 y n_3 tomadas de la población global, sin embargo, en vez de registrar la frecuencia observada de A's, B's y C's de cada muestra, lo que se hizo fue anotar:

1. Número de A's en la muestra de tamaño $n_1 = y_1$.
2. Número de B's en la muestra de tamaño $n_2 = y_2$
3. Número de A's en la muestra de tamaño $n_3 = y_3$

Estime π_1, π_2 y π_3 usando el método de la mínima ji-cuadrada: suponga que $n_1 = 100, y_1 = 22, n_2 = 150, y_2 = 52, n_3 = 200.y_2 = 77$. Esto es, encuentre π_1, π_2 y π_3 que minimizen:

$$\frac{(y_1 - n_1\pi_1)^2}{n_1\pi_1} + \frac{[(n_1 - y_1) - n_1(1 - \pi_1)]^2}{n_1(1 - \pi_1)} + ... + \frac{(y_3 - n_3\pi_3)^2}{n_3\pi_3} + \frac{[(n_3 - \pi_3) - n_3(1 - \pi_3)]^2}{n_3(1 - \pi_3)}$$

con la restricción $\pi_3 = 1 - \pi_1 - \pi_2$(sugerimos usar directamente nlminb de R).

Solución

```
suma1 <- function(p1)
  {
    (22 - (100*p1) )^2/(100*p1) +
    ( (100-22) - 100*(1-p1) )^2/( 100*(1 - p1) )
  }

suma2 <- function(p2)
  {
    (52 - (150*p2) )^2/(150*p2) +
    ( (150-52) - 150*(1-p2) )^2/( 150*( 1 - p2) )
  }

suma3 <- function(p3)
  {
    (77 - (200*p3) )^2/(200*p3) +
    ( (200-77) - 200*(1-p3) )^2/( 200*(1 - p3) )
  }

Manu_G<-function(p){
    p1 <- p[1]
    p2 <- p[2]
    suma1(p1) + suma2(p2) + suma3(1 - p[1] - p[2])}

valores <- nlminb(c(1/3,1/3),
                  Manu_G, lower = c(0,0), upper = c(1,1))
a <- valores$par
x1 <- a[1]
x2 <- a[2]
x3 <- 1 - x2 - x1
x1;x2;x3
```

```
[1] 0.2395447
[1] 0.3628983
[1] 0.397557
```

Al final obtenemos los valores y los mandamos a imprimir en pantalla, para $\pi_1 = 0.239544$, $\pi_2 = 0.36289$ y $\pi_3 = 0.397557$, en donde observamos que se cumple la restricción $\pi_3 = 1 - \pi_1 - \pi_2$.

9. Problema 9

Se toman los datos relacionados con el hundimiento del Titanic en abril de 1912. El resultado se puede expresar en una tabla de dimensión 4.
Las variables son *Classs* de los pasejeros(1,2,3, Tripulación), *Sex* de los pasajeros(Male, Femele), *Age* de los pasajeros(Child, Adult), y *Survived* si los pasajeros sobrevivieron o no (No, Yes). Usar libreria en R ”titanicz los datos se encuentran en la variable ”Titanic”.

Solución

```
library(titanic)
library(MASS)
Titanic

, , Age = Child, Survived = No

      Sex
Class  Male Female
  1st     0      0
  2nd     0      0
  3rd    35     17
  Crew    0      0
```

```
, , Age = Adult, Survived = No

      Sex
Class  Male Female
  1st   118      4
  2nd   154     13
  3rd   387     89
  Crew  670      3

, , Age = Child, Survived = Yes

      Sex
Class  Male Female
  1st     5      1
  2nd    11     13
  3rd    13     14
  Crew    0      0

, , Age = Adult, Survived = Yes

      Sex
Class  Male Female
  1st    57    140
  2nd    14     80
  3rd    75     76
  Crew  192     20
```

```
datos <- data.frame(expand.grid(
  Survived = factor(c('no', 'yes'), levels = c('yes','no') ),
  Age = factor( c('Child','Adult'), levels = c('Adult', 'Child') ),
  Sex = factor( c('M','F'), levels = c('F','M') ),
  Class = factor( c('1','2','3','4'), levels = c('4','3','2','1') )),
  frec = c(0,5,48,57 ,0,1,4,140, 0,11,154,14, 0,13,13,80, 35,13,387,75,
           17,14,89,76, 0,0,670,192, 0,0,3,20))
```

```
datos

   Survived   Age Sex Class frec
1        no Child   M     1    0
2       yes Child   M     1    5
3        no Adult   M     1   48
4       yes Adult   M     1   57
5        no Child   F     1    0
6       yes Child   F     1    1
7        no Adult   F     1    4
8       yes Adult   F     1  140
9        no Child   M     2    0
10      yes Child   M     2   11
11       no Adult   M     2  154
12      yes Adult   M     2   14
13       no Child   F     2    0
14      yes Child   F     2   13
15       no Adult   F     2   13
16      yes Adult   F     2   80
17       no Child   M     3   35
18      yes Child   M     3   13
19       no Adult   M     3  387
20      yes Adult   M     3   75
21       no Child   F     3   17
22      yes Child   F     3   14
23       no Adult   F     3   89
24      yes Adult   F     3   76
25       no Child   M     4    0
26      yes Child   M     4    0
27       no Adult   M     4  670
28      yes Adult   M     4  192
29       no Child   F     4    0
30      yes Child   F     4    0
31       no Adult   F     4    3
```

```
32      yes Adult   F     4   20
```

```
ff <- c("Class+Age+Sex+Survived")
gg <- c(
    "Class","Age",'Sex','Survived',

    "Class*Age*Sex",  ## XYZ
    "Class*Age*Survived", ## XYW
    "Class*Sex*Survived", ## XZW
    "Age*Sex*Survived", ##YZW

    "Class*Age*Sex+Class*Age*Survived", ## XYZ + XYW
    "Class*Age*Sex+Class*Sex*Survived", ## XYZ+ XZW
    "Class*Age*Sex+Age*Sex*Survived", ## XYZ + YZW
    "Class*Age*Survived+Class*Sex*Survived", ##XYW+XZW
    "Class*Age*Survived+Age*Sex*Survived", ## xYW+YZW
    "Class*Sex*Survived+Age*Sex*Survived", ## XZW+YZW

    "Class*Age*Sex+Class*Age*Survived+Class*Sex*Survived",
    "Class*Age*Sex+Class*Age*Survived+Age*Sex*Survived",
    "Class*Age*Survived+Class*Sex*Survived+Age*Sex*Survived",
    "Class*Age*Sex+Class*Sex*Survived+Age*Sex*Survived",
    "Class*Age*Sex+Class*Age*Survived+Class*Sex*Survived+Age*Sex*Survived",

    "Class*Age*Sex+Class*Age*Survived+Class*Sex*Survived+Age*Sex*Survived+Class*Age*


tt <- matrix(0,21,4)
colnames(tt) <- c("G2","X2","gl","p-valor")
out <- loglm(frec~Class+Age+Sex+Survived,
             data=Titanic, param=T,fit=T)
tt[1,] <- c(out$lrt,out$pearson,out$df,1-pchisq(out$lrt,out$df))
for(j in 1:20){
  if(j < 4){
```

```r
    fmla <- as.formula(paste("frec ~","+",gg[j]))
    out <- loglm(fmla, data=Titanic, param=T,fit=T)
    tt[j+1,] <- c(out$lrt,out$pearson,out$df,1-pchisq(out$lrt,out$df))
  }
  else{
    fmla <- as.formula(paste("frec ~",ff,"+",gg[j]))
    out <- loglm(fmla, data=Titanic, param=T,fit=T)
    tt[j+1,] <- c(out$lrt,out$pearson,out$df,1-pchisq(out$lrt,out$df))
  }}

tt
```

```
                G2        X2 gl      p-valor
 [1,] 1.243663e+03 1637.4455 25 0.000000e+00
 [2,] 4.477324e+03 5901.0840 28 0.000000e+00
 [3,] 2.769572e+03 3647.2745 30 0.000000e+00
 [4,] 4.184815e+03 5513.2576 30 0.000000e+00
 [5,] 1.243663e+03 1637.4455 25 0.000000e+00
 [6,] 6.719622e+02       NaN 15 0.000000e+00
 [7,] 8.596453e+02       NaN 15 0.000000e+00
 [8,] 2.253383e+02  242.1268 15 0.000000e+00
 [9,] 7.636986e+02  755.1860 21 0.000000e+00
[10,] 4.362715e+02       NaN  8 0.000000e+00
[11,] 6.623848e+01       NaN  8 2.744627e-11
[12,] 2.152813e+02       NaN 12 0.000000e+00
[13,] 2.222167e+01       NaN  8 4.521358e-03
[14,] 3.992412e+02       NaN 12 0.000000e+00
[15,] 1.798425e+02  173.8851 12 0.000000e+00
[16,] 1.685503e+00       NaN  4 7.933493e-01
[17,] 6.501320e+01       NaN  6 4.287237e-12
[18,] 9.783359e+00       NaN  6 1.340767e-01
[19,] 3.726253e+01       NaN  6 1.565079e-06
[20,] 3.683794e-04       NaN  3 9.999981e-01
```

```
[21,] 0.000000e+00       NaN  0 1.000000e+00

#### probando otros modelos
sat.model <- loglm(frec~Class*Sex*Age*Survived , data = Titanic,
                   param=T, fit=T )
sat.model

Call:
loglm(formula = frec ~ Class * Sex * Age * Survived, data = Titanic,
    param = T, fit = T)

Statistics:
                 X^2 df P(> X^2)
Likelihood Ratio   0  0        1
Pearson          NaN  0        1

stepAIC(sat.model, direction = 'backward', trace = 0)

Call:
loglm(formula = frec ~ Class + Sex + Age + Survived + Class:Sex +
    Class:Age + Sex:Age + Class:Survived + Sex:Survived + Age:Survived +
    Class:Sex:Age + Class:Sex:Survived + Class:Age:Survived,
    data = Titanic, param = T, fit = T, evaluate = FALSE)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.685479  4 0.7933536
Pearson               NaN  4       NaN
```

Al realizar la tabla con cada uno de los posibles combinaciones tomando en cuenta lo que plantea el ejercicio llegamos a la conclución que realmente todos los modelos propuestos son rechazados, basandonos en ese valor en el p-value.
Finalmente con el comando StepAIC se trata de encontrar el mejor modelo

comenzando desde la cuádriple interacción.

10. Problema 10

Se ha ralizado un análisis sobre el valor terapéutico del ácido ascórbico(vitamina C) en la relación a su efecto sobre la gripe común. Se tiene una tabla 2x2 con los recuentos correspondientes para una muestra de 279 personas:
Aplicar un modelo lineal para determinar si existe evidencia suficiente para asegurar que el ácido ascórbico ayuda tener menos gripe.

Solución

```
datos2 <- data.frame( expand.grid(
  Gripe_V = factor(c('Gripe', 'no-Gripe'), levels = c('no-Gripe','Gripe') ),
  Aspirina = factor( c('Placebo','A.Ascorbico'),
                     levels = c('A.Ascorbico', 'Placebo') )),
  frec = c(31,109,17,122) )

datos2

   Gripe_V    Aspirina frec
1    Gripe     Placebo   31
2 no-Gripe     Placebo  109
3    Gripe A.Ascorbico   17
4 no-Gripe A.Ascorbico  122

model_datos2_10 <- loglm(frec~Gripe_V + Aspirina ,
                         data = datos2,
                         param=T, fit=T)
model_datos2_10
```

```
Call:
loglm(formula = frec ~ Gripe_V + Aspirina, data = datos2, param = T,
    fit = T)

Statistics:
                      X^2 df   P(> X^2)
Likelihood Ratio 4.871697  1 0.02730064
Pearson          4.811413  1 0.02827186

glmTcoef <- coef(model_datos2_10)

glmTcoef

$`(Intercept)`
[1] 3.963656

$Gripe_V
  no-Gripe      Gripe
 0.7856083 -0.7856083

$Aspirina
 A.Ascorbico      Placebo
-0.003584245  0.003584245
```

Al observar los coeficientes que regresa el modelo encontramos que no hay evidencia suficiente para asegurar que el ácido ascórbico ayuda a tener menos gripe.