Regression Models Assignment 2

Author: Marta Cortés
Master in Statistics for Data Science
Universidad Carlos III de Madrid

```r
rm(list=ls())
```

Installation of Packages

```r
library("MASS")
library("car")
library("ISLR")
library("corrplot")
library("glmnet")
library("biglasso")
library("biglm")
library("leaps")
library("dplyr")
library("utils")
library(faraway)
```

1. (0.75 points) Fit a logistic regression model to predict the probability self-perceived health using the predictors sex and weight without including the interaction between them.

• Interpret the coefficients in terms of odds ratios

• Plot the predicted probabilities for males and females

• Given that a person has weight = 95, what is the relative risk and odds ratio of self-perceived good health of a female compared with a male?

```r
health <- read.table("/Users/cortesocanamarta/Documents/Marta/MÁSTER DATA SCIENC
E/Regression Models/datasets/health(Chapter5).txt",header=TRUE)
health <- dplyr::select(health, -g01)
```

The variables of the data are:

- age: Age in years
- height: height in cms
- weight: weight in kgs
- year: year in which the data
- drink: categorical variable (to be defined as a factor) with levels 0, 1 and 2 (no drink, occasional, frequent)
- sex: categorical variable (to be defined as a factor) with levels 1 and 2 (male, female)
- con_tab: categorical variable (to be defined as a factor) with levels 1 and 2 (non or occasional smoker, frequent smoker)
- educa: categorical variable (to be defined as a factor) with levels 1, 2, 3 and 4 indicating the level of education (low, low-median, median-high, high)
- imc: body mass index
- g02: the response variable, takes values 1 and 0 (good health or no good health)

First of all, we need to define the categorical variables as factor. After that, in order to apply the Logistic Regression, we need to relate the response to the predictor sex and weigth, and model the relationship of the predictors to the probability p_i, then, we get the linear predictor:

$$\eta_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots \beta_k x_{ik}$$

As the linear relation $\eta_i = p_i$ doesn't work here, we use a link function "g", such that $\eta_i = g(p_i)$, instead. The most popular choice of link function is the logit:

$$\eta = \log\left(\frac{p}{1-p}\right) \Rightarrow p = \frac{e^\eta}{1+e^\eta}$$

The ratio $p/(1-p)$ in the logit transformation is called the odds.

In R, the function to use logistic regression is the glm() whose main arguments are:

- formula: similar to linear models, response and predictors you want to use for the prediction
- family: probability distribution of the response (normal by default, but you can choose binomial, poisson, gamma, etc.)
- link: link function you want to use
- data: data frame you want to use

```
#Categorical values as factors
health$sex<-factor(health$sex)
health$g02<-factor(health$g02)
health$con_tab<-factor(health$con_tab)
health$educa<-factor(health$educa)
health$drink<-factor(health$drink)

#Logistic Regression using sex and weight
lrmod1 <- glm(g02 ~ sex + weight, family=binomial, health)
summary(lrmod1)

##
## Call:
## glm(formula = g02 ~ sex + weight, family = binomial, data = health)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2849   0.5139   0.6055   0.6773   1.7336
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.712536   0.218700  16.975   <2e-16 ***
## sex2        -0.825937   0.076422 -10.808   <2e-16 ***
## weight      -0.026188   0.002671  -9.803   <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: 7177.9  on 7356  degrees of freedom
## Residual deviance: 7042.3  on 7354  degrees of freedom
## AIC: 7048.3
##
## Number of Fisher Scoring iterations: 4
```

The estimated coefficients means that when you change one unit in the predictor the logit of feeling healthy changes in the number of units that are showed. In other words, as male is the reference level, if you are female, the log odds of the self-perceived good health will change **0.826 units**.

The same, the log odds of the self perceived good health will change in **0.026 units** if you increase the weight in one kg.

We can translate the change in log odds to the change in odds with the exp function.

```
#Interpretation of predictors
exp(coef(lrmod1))

## (Intercept)        sex2      weight
##  40.9575330   0.4378247   0.9741522
```

Since the reference level is male, we can conclude that females are **1/0.438 times** less likely to have a self-perceived good health (1)

On the other hand, when you increase one kg the weight of the person, then, the self-perceived health change 0.974 units in its odds, in other words, for one more kg, the odds of the percepcion of good health will be **1/0.974 times** smaller.

We can calculate the probabilities using the relation: $p = odds/(1 + odds)$

Then, since the predictor "sex" is categorical with two levels (0 and 1), the difference in the logit for an individual such that X=0 or X=1, is β_1, then:

$$p = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \quad 1 - p = \frac{1}{1 + e^{\beta_0+\beta_1 X}}$$

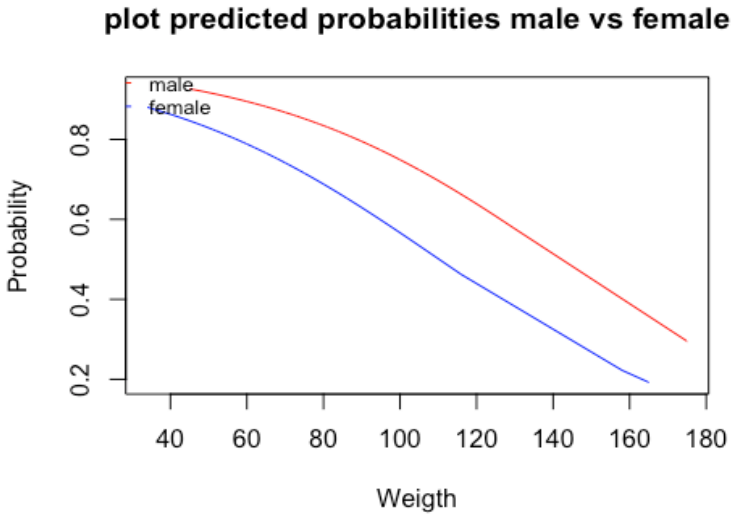
```
#Calculate the odds ratio and relative risk of good health in a female compared
with a male for 95kg of weigth
p1<-predict(lrmod1, newdata=data.frame(sex="1",weight=95),type="response")
p2<-predict(lrmod1, newdata=data.frame(sex="2",weight=95),type="response")

OR<-(p2/(1-p2))/(p1/(1-p1))
RR<-p2/p1
```

As we have detailed before, the odds ratio of a self-perceived good health in female compared to a male is **0.438**, as this value is less than 1, a good health in a female is less frequent than in a male.

The relative risk of a self-perceived good health in a female compared to a male is **0.774**, that means the likelihood (probability) of a female having a good health is less than in a male.

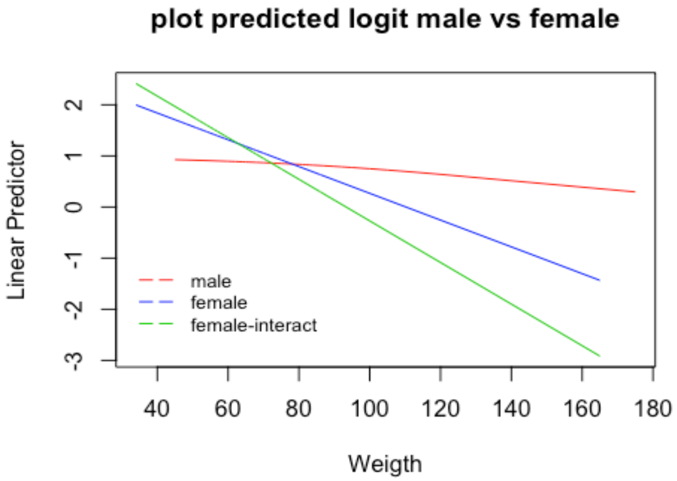
```
#Representation of the predicted probabilities for males and females
fittedlrmod<-predict(lrmod1,type="response")
plot(health$weight,fittedlrmod,type="n",main="plot predicted probabilities male
vs female",xlab="Weigth",ylab="Probability")

weight1<-health$weight[health$sex==1]
p1<-fittedlrmod[health$sex==1]
o<-order(weight1)
lines(weight1[o],p1[o],col=2)

weight2<-health$weight[health$sex==2]
```

```
p2<-fittedlrmod[health$sex==2]
o<-order(weight2)
lines(weight2[o],p2[o],col=4)

legend(15,1,col=c(2,4),c("male","female"),lty=1,bty="n",cex=0.8)
```



2. (0.25 points) Repeat the previous exercise including the interaction between weight and sex in the model, compare and comment the results. Use the LRT to test if the terms in the model are significant.

```
#Model with the interaction
lrmod2 <- glm(g02 ~ sex + weight+sex:weight, family=binomial, health)
summary(lrmod2)

##
## Call:
## glm(formula = g02 ~ sex + weight + sex:weight, family = binomial,
##     data = health)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1347   0.5501   0.5933   0.6527   2.3243
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.561521   0.306033   8.370  < 2e-16 ***
## sex2         1.236831   0.393019   3.147  0.00165 **
## weight      -0.011713   0.003836  -3.053  0.00226 **
## sex2:weight -0.028984   0.005438  -5.330 9.81e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7177.9  on 7356  degrees of freedom
## Residual deviance: 7013.5  on 7353  degrees of freedom
## AIC: 7021.5
##
## Number of Fisher Scoring iterations: 4

#Testing of significance
lrt <- 2*(lrmod2$deviance-lrmod1$deviance)
df<-lrmod2$df.residual-lrmod1$df.residual
1-pchisq(abs(lrt),abs(df))

#Anova function
anova(lrmod2,lrmod1,test="Chisq")

## Analysis of Deviance Table
##
## Model 1: g02 ~ sex + weight + sex:weight
## Model 2: g02 ~ sex + weight
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      7353     7013.5
## 2      7354     7042.3 -1  -28.858 7.79e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#Plot
lrmod1 <- glm(g02 ~ sex + weight, family=binomial, health)
lrmod2 <- glm(g02 ~ sex + weight + sex:weight, family=binomial, health)

fittedlrmod1<-predict(lrmod1)
fittedlrmod2<-predict(lrmod2)

plot(health$weight,fittedlrmod2,type="n",main="plot predicted logit male vs female",xlab="Weigth",ylab="Linear Predictor")

weight1<-health$weight[health$sex==1]
lp1<-fittedlrmod1[health$sex==1]
o<-order(weight1)
lines(weight1[o],p1[o],col=2)

weight2<-health$weight[health$sex==2]
p2<-fittedlrmod1[health$sex==2]
o<-order(weight2)
lines(weight2[o],p2[o],col=4)

weight3<-health$weight[health$sex==2]
p3<-fittedlrmod2[health$sex==2]
o<-order(weight3)
lines(weight3[o],p3[o],col=3)

legend(30,-1,col=c(2,4,3),c("male","female", "female-interact"),lty=5,bty="n",cex=0.8)
```



The estimated parameter for the interaction seems quite small compared with the parameter for sex (0.029 vs 1.237), so probably both rates of change will continue to be very similar.

However, applying the likelihood ratio test and the Anova function in R, which test if all coefficients are zero, we obtain that the p-value is, in fact, 0, then, we can reject the null hypothesis which consider no significant the terms in the model, and assume they actually are.

In addition, we can see it graphically plotting the linear predictor, in where we can see that the one for females with and without interaction is quite different, therefore, then, it doesn't behave the same.

3. (0.5 points) Calculate and interpret the confidence intervals for the coefficients in the model fitted the previous exercise and calculate the estimated expected probability of males of 80 and 165 kg and give a confidence interval for each of those predictions.

```
#Confident Intervals
confint(lrmod2)

## Waiting for profiling to be done...

##                   2.5 %       97.5 %
## (Intercept)  1.96086919  3.161216659
## sex2         0.46795815  2.008969452
## weight      -0.01919948 -0.004151157
## sex2:weight -0.03968172 -0.018361339

#Calculate the estimate expected probability
response80<-predict(lrmod2,data.frame(sex="1",weight=80), type="response")
response165<-predict(lrmod2,data.frame(sex="1",weight=165), type="response")
response80

##         1
## 0.8354128

response165

##         1
## 0.6522367
```

```
#Calculate the CI
fitted80<-predict(lrmod2,data.frame(sex="1",weight=80), se.fit=TRUE)
fitted165<-predict(lrmod2,data.frame(sex="1",weight=165), se.fit=TRUE)

L.inf80<-with(fitted80,exp(fit-1.96*se.fit)/(1+exp(fit-1.96*se.fit)))
L.sup80<-with(fitted80,exp(fit+1.96*se.fit)/(1+exp(fit+1.96*se.fit)))
L.inf80

##        1
## 0.822903

L.sup80

##        1
## 0.847203

L.inf165<-with(fitted165,exp(fit-1.96*se.fit)/(1+exp(fit-1.96*se.fit)))
L.sup165<-with(fitted165,exp(fit+1.96*se.fit)/(1+exp(fit+1.96*se.fit)))
L.inf165

##         1
## 0.4939067

L.sup165

##         1
## 0.7828148
```

As we can see with the confidential intervals for the odds ratio, the effect of weight on self-perceived health is not too different for males and females. Since the zero doesn't lie inside the interval, the confidential interval are representative.

The estimate expected probability of males with 80kg of weight is **0.835**, while the estimate probability of males with 165kg is much lower, **0.652**.

On the other hand, the confidential interval for the prediction of the probability for males with 80kgs is **0.823-0.847**, and the CI for the prediction of the probability for males with 165kgs is **0.494-0.783**.

That means in the 95% of the cases our response value for men and 80kg, and men and 165kg lie the previous interval, respectively.

4. (1 point) Use all predictors availables in the dataset health to find the best subset of predictors (and their possible interactions) using LRT, AIC and BIC. Are the chosen models the same?. If the answer is not, which one would you use as your final model?. Check the predictive accuracy of the final model.

```
#Possible subset of predictors, first way
full<-glm(g02~.,family=binomial,health,na.action=na.fail)
summary(full)

## 
## Call:
## glm(formula = g02 ~ ., family = binomial, data = health, na.action = na.fail)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.5640   0.3379   0.4743   0.6455   1.7725
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.4961319 56.4990785   0.097 0.922505
## sex2        -0.3111181  0.0924709  -3.364 0.000767 ***
## weight       0.0075375  0.0250436   0.301 0.763432
## height      -0.0015613  0.0220561  -0.071 0.943568
## con_tab2    -0.1483280  0.0656448  -2.260 0.023849 *
## year        -0.0007064  0.0281941  -0.025 0.980011
## educa2       0.3402181  0.1073338   3.170 0.001526 **
## educa3       0.8243733  0.1157786   7.120 1.08e-12 ***
## educa4       1.1822373  0.1177180  10.043  < 2e-16 ***
## imc         -0.0860625  0.0693225  -1.241 0.214428
## drink1       0.4382270  0.0672640   6.515 7.27e-11 ***
## drink2       0.4040458  0.1691012   2.389 0.016877 *
## age         -0.0348452  0.0029123 -11.965  < 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: 7177.9  on 7356  degrees of freedom
## Residual deviance: 6365.7  on 7344  degrees of freedom
## AIC: 6391.7
##
## Number of Fisher Scoring iterations: 5
```

```
library(MuMIn)

allfits<-dredge(full, beta="none")
```

```
## Fixed term is "(Intercept)"
```

```
allfits<-allfits[order(allfits[,13]),]
resultF1<-head(allfits)
resultF1
```

```
##     (Intercept)         age con_tab drink educa      height         imc sex
## 112   3.8982259 -0.03555523       +     +     +          NA -0.06596117   +
## 240   3.8214137 -0.03482738       +     +     +          NA -0.08130537   +
## 128   3.0079870 -0.03481723       +     +     + 0.004908896 -0.06536979   +
## 368   4.2250146 -0.03555523       +     +     +          NA -0.06596034   +
## 224  -0.3671258 -0.03482592       +     +     + 0.025057828          NA   +
## 110   3.7993496 -0.03548339    <NA>     +     +          NA -0.06511706   +
##         weight         year df    logLik     AICc     delta
## 112 0.21675059           NA 10 -3183.387 6386.803 0.0000000
## 240 0.13603541           NA 11 -3182.849 6387.735 0.9316642
## 128 0.13028274           NA 11 -3182.893 6387.821 1.0180806
## 368 0.07950078 -0.000163207 11 -3183.386 6388.809 2.0059610
## 224 0.06149093           NA 11 -3183.643 6389.323 2.5197153
## 110 0.05012784           NA  9 -3185.853 6389.731 2.9283414
```

```
#Possible subset of predictors other way
X.health<-dplyr::select(health,-g02)
colnames<-names(X.health)
```

```
X = paste0("x",1:9)
for (i in 1:length(X)) {
  X[i]<-colnames[i]
}
X

## [1] "sex"     "weight"  "height"  "con_tab" "year"    "educa"   "imc"
## [8] "drink"   "age"

out <- unlist(lapply(1:9, function(n) combn(X, n, FUN=function(row) paste0(row,
collapse = "+"))))
head(out)

##   [1] "sex"
##   [2] "weight"
##   [3] "height"
##   [4] "con_tab"
##   [5] "year"
##   [6] "educa"

response<-"g02"
AICs<-c(0)
tableF<-as.data.frame(matrix(nrow=1,ncol=2))
names(tableF)[1]<-c("modcomb")
names(tableF)[2]<-c("V2")
for (i in 1:length(out)) {
  f<-as.formula(paste(response,out[i],sep="~"))
  mod<-glm(f,family=binomial,health)
  AICs[i]<-summary(mod)$aic
  AICs[i]<-as.numeric(AICs[i])
  modcomb<-as.character(f)
  modcomb<-modcomb[3]
  modcomb<-paste0(modcomb,collapse="+")
  table<-cbind(modcomb,AICs[i])
  tableF<-rbind(tableF,table)
}

tableF<-na.omit(tableF)
results<-tableF[order(tableF[,2]),]
resultF2<-head(results)
resultF2

##                                                          modcomb
## 437                   sex + con_tab + educa + imc + drink + age
## 486          sex + weight + con_tab + educa + imc + drink + age
## 492          sex + height + con_tab + educa + imc + drink + age
## 494            sex + con_tab + year + educa + imc + drink + age
## 475       sex + weight + height + con_tab + educa + drink + age
## 507 sex + weight + height + con_tab + educa + imc + drink + age
##                  V2
## 437 6386.77300567252
## 486 6387.69867534187
## 492 6387.78509175896
## 494 6388.77297211738
## 475  6389.2867264299
## 507 6389.69358540758
```

Apparently, the best subsets of predictors are the next:

- age+con_tab+drink+educa+imc+sex
- age+con_tab+drink+educa+imc+sex+weight
- age+con_tab+drink+educa+height+imc+sex+weight

Let's study further these models adding interactions between the most significant predictors and calculate the measure criterias (LRT, AIC and BIC) for each of them, in order to determine which is the best.

```
mod01=glm(g02~age+con_tab+drink+educa+imc+sex+age:con_tab,family=binomial,health
)
summary(mod01)$aic

## [1] 6386.216

mod02<-glm(g02~age+con_tab+drink+educa+imc+sex+age:drink,family=binomial,health)
summary(mod02)$aic

## [1] 6385.012

mod03<-glm(g02~age+con_tab+drink+educa+imc+sex+age:educa,family=binomial,health)
summary(mod03)$aic

## [1] 6391.234

mod04<-glm(g02~age+con_tab+drink+educa+imc+sex+age:imc,family=binomial,health)
summary(mod04)$aic

## [1] 6388.738

mod05<-glm(g02~age+con_tab+drink+educa+imc+sex+age:sex,family=binomial,health)
summary(mod05)$aic

## [1] 6388.552

mod06<-glm(g02~age+con_tab+drink+educa+imc+sex+age:con_tab+age:drink,family=bino
mial,health)
summary(mod06)$aic

## [1] 6385.542

mod07<-glm(g02~age+con_tab+drink+educa+imc+sex+con_tab:drink,family=binomial,hea
lth)
summary(mod07)$aic

## [1] 6389.318

mod08<-glm(g02~age+con_tab+drink+educa+imc+sex+con_tab:educa,family=binomial,hea
lth)
summary(mod08)$aic

## [1] 6392.648

mod09<-glm(g02~age+con_tab+drink+educa+imc+sex+con_tab:imc,family=binomial,healt
h)
summary(mod09)$aic
```

```
## [1] 6385.284

mod10<-glm(g02~age+con_tab+drink+educa+imc+sex+con_tab:sex,family=binomial,healt
h)
summary(mod10)$aic

## [1] 6386.773

mod11<-glm(g02~age+con_tab+drink+educa+imc+sex+age:con_tab+con_tab:imc,family=bi
nomial,health)
summary(mod11)$aic

## [1] 6385.945

mod12<-glm(g02~age+con_tab+drink+educa+imc+sex+con_tab:drink+con_tab:imc,family=
binomial,health)
summary(mod12)$aic

## [1] 6387.717

mod13<-glm(g02~age+con_tab+drink+educa+imc+sex+age:con_tab+age:drink+con_tab:dri
nk+con_tab:imc,family=binomial,health)
summary(mod13)$aic

## [1] 6387.959

mod14<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:educa,family=binomial,healt
h)
summary(mod14)$aic

## [1] 6395.172

mod15<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc,family=binomial,health)
summary(mod15)$aic

## [1] 6385.065

mod16<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:sex,family=binomial,health)
summary(mod16)$aic

## [1] 6388.578

mod17<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:con_tab,family=bino
mial,health)
summary(mod17)$aic

## [1] 6384.521

mod18<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:drink,family=binomi
al,health)
summary(mod18)$aic

## [1] 6382.081

mod19<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:con_tab+age:drink,f
amily=binomial,health)
summary(mod19)$aic
```

```
## [1] 6382.65

mod20<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:con_tab+age:drink+c
on_tab:imc,family=binomial,health)
summary(mod20)$aic

## [1] 6381.682

mod21<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:drink+con_tab:imc,f
amily=binomial,health)
summary(mod21)$aic

## [1] 6380.186

mod22<-glm(g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:con_tab+con_tab:imc
,family=binomial,health)
summary(mod22)$aic

## [1] 6383.877

mod24<-glm(g02~age+con_tab+drink+educa+imc+sex+educa:sex,family=binomial,health)
summary(mod24)$aic

## [1] 6391.842

mod25<-glm(g02~age+con_tab+drink+educa+imc+sex+imc:sex,family=binomial,health)
summary(mod25)$aic

## [1] 6388.722
```

Regarding the AIC criteria the best models are: mod18, mod20 and mod21. Lets calculate the other measures criterias (LRT and BIC) in other to confirm if the mod21 is the best as the AIC specifies.

```
BICmod18<-2*summary(mod18)$deviance+log(nrow(health))+length(coef(mod18))
BICmod20<-2*summary(mod20)$deviance+log(nrow(health))+length(coef(mod20))
BICmod21<-2*summary(mod21)$deviance+log(nrow(health))+length(coef(mod21))
BICfull<-2*summary(full)$deviance+log(nrow(health))+length(coef(full))

ddmod18<-summary(mod18)$null.deviance-summary(mod18)$deviance
ddfmod18<-abs(7356-7343)
LRTmod18<-1-pchisq(ddmod18,ddfmod18)

ddmod20<-summary(mod20)$null.deviance-summary(mod20)$deviance
ddfmod20<-abs(7356-7341)
LRTmod20<-1-pchisq(ddmod20,ddfmod20)

ddmod21<-summary(mod21)$null.deviance-summary(mod21)$deviance
ddfmod21<-abs(7356-7342)
LRTmod21<-1-pchisq(ddmod21,ddfmod21)

ddfull<-summary(full)$null.deviance-summary(full)$deviance
ddffull<-abs(7356-7344)
LRTfull<-as.double(1-pchisq(ddfull,ddffull))
```

The next table summarise the results of the measure criterias for the best models comparing with the full model.

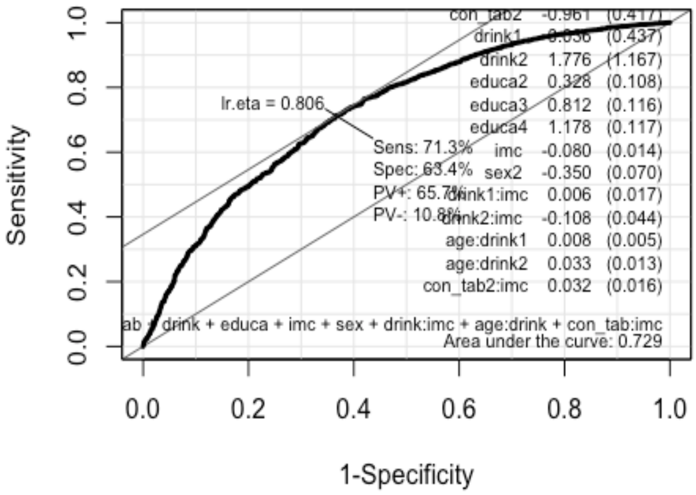
	Model	AIC	BIC	LRT
1	full	6391.693	12753.289	0
2	mod18	6382.081	12731.065	0.000655
3	mod20	6381.682	12724.267	0.001128
4	mod21	6380.186	12724.275	0.000429

The best model, regarding the AIC, is the mod21, but regarding BIC, the best model is mod20, although the values of BIC are really close. Therefore, I would choose the mod21 because it has the best performance with less interactions and also has the smallest LRT:

mod21: g02 ~ age + con_tab + drink + educa + imc + sex + drink:imc + age:drink + con_tab:imc

Finally, let's represent the ROC plot and calculate the accuracy for that model:

```
library(Epi)

ROC(form=g02~age+con_tab+drink+educa+imc+sex+drink:imc+age:drink+con_tab:imc, da
ta=health, plot="ROC", lwd=3, cex=1.5)
```



The AUC for the model is **0.81** which determines that the model is good in terms of predicting.

5. (1 point) In a hospital in New York a sample of size 100 was taken among alcoholics, and another sample among non-alcoholics of size 500. For each patient is was recorded whether he/she suffered from cirrhosis of the liver. A similar investigation was carried out in Philadelphia with samples of 228 alcoholics and 3772 non-alcoholics.

Use a logistic regression model to analyze the dependence of disease prevalence on site and patient status.

First of all we have to translate the information to a data frame in R. In order to do so, we consider the next factor variables: State (0 = NYC, 1 = PHIL), Drinker (0 = Not alcoholic, 1 = Alcoholic) and Cirrhosis suffered (0 = not sick, 1 = sick).

```r
data5=as.data.frame(matrix(nrow=1,ncol=3))
names(data5)[1]<-c("State")
names(data5)[2]<-c("Alcoholic")
names(data5)[3]<-c("Sick")

for (i in 1:600) {
  data5<-rbind(data5,c(0,0,0))
}

for (j in 1:4000) {
  data5<-rbind(data5, c(1,0,0))
}

data5<-na.omit(data5)

for (i in 1:4600) {
  if (data5[i,1]==0 & length(which(data5[,2]==1))<100) {
    data5[i,2]=1
  }
}

for (i in 1:4600) {
  if (data5[i,1]==0 & data5[i,2]==0 & length(which(data5[,3]==1))<25) {
    data5[i,3]=1
  }
}

for (i in 1:4600) {
  if (data5[i,1]==0 & data5[i,2]==1 & length(which(data5[,3]==1))<60) {
    data5[i,3]=1
  }
}


for (i in 1:4600) {
  if (data5[i,1]==1 & length(which(data5[,2]==1))<328) {
    data5[i,2]=1
  }
}

for (i in 1:4600) {
  if (data5[i,1]==1 & data5[i,2]==0 & length(which(data5[,3]==1))<165) {
    data5[i,3]=1
  }
}

for (i in 1:4600) {
  if (data5[i,1]==1 & data5[i,2]==1 & length(which(data5[,3]==1))<210) {
    data5[i,3]=1
  }
}

data5$State<-factor(data5$State)
data5$Alcoholic<-factor(data5$Alcoholic)
```

```
data5$Sick<-factor(data5$Sick)

mod1ex5<-glm(Sick~State,family=binomial,data5)
summary(mod1ex5)

##
## Call:
## glm(formula = Sick ~ State, family = binomial, data = data5)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4590  -0.2765  -0.2765  -0.2765   2.5626
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.1972     0.1361  -16.15  < 2e-16 ***
## State1       -1.0480     0.1595   -6.57 5.04e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1706.7  on 4599  degrees of freedom
## Residual deviance: 1669.4  on 4598  degrees of freedom
## AIC: 1673.4
##
## Number of Fisher Scoring iterations: 6

mod2ex5<-glm(Sick~Alcoholic,family=binomial,data5)
summary(mod2ex5)

##
## Call:
## glm(formula = Sick ~ Alcoholic, family = binomial, data = data5)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7478  -0.2486  -0.2486  -0.2486   2.6428
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.46140    0.08907  -38.86   <2e-16 ***
## Alcoholic1   2.33000    0.15642   14.90   <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: 1706.7  on 4599  degrees of freedom
## Residual deviance: 1528.4  on 4598  degrees of freedom
## AIC: 1532.4
##
## Number of Fisher Scoring iterations: 6

mod3ex5<-glm(Sick~State+Alcoholic,family=binomial,data5)
summary(mod3ex5)
```

```
##
## Call:
## glm(formula = Sick ~ State + Alcoholic, family = binomial, data = data5)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9045  -0.2360  -0.2360  -0.2360   2.6814
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.8859     0.1638 -17.614  < 2e-16 ***
## State1       -0.6814     0.1716  -3.971 7.17e-05 ***
## Alcoholic1    2.2035     0.1605  13.732  < 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: 1706.7  on 4599  degrees of freedom
## Residual deviance: 1513.9  on 4597  degrees of freedom
## AIC: 1519.9
##
## Number of Fisher Scoring iterations: 6
```

```
mod4ex5<-glm(Sick~State+Alcoholic+State:Alcoholic,family=binomial,data5)
summary(mod4ex5)
```

```
##
## Call:
## glm(formula = Sick ~ State + Alcoholic + State:Alcoholic, family = binomial,
##     data = data5)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9282  -0.2376  -0.2376  -0.2376   2.6763
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.9444     0.2052 -14.349  < 2e-16 ***
## State1             -0.6087     0.2278  -2.672  0.00754 **
## Alcoholic1          2.3254     0.2934   7.927 2.25e-15 ***
## State1:Alcoholic1  -0.1751     0.3515  -0.498  0.61846
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1706.7  on 4599  degrees of freedom
## Residual deviance: 1513.7  on 4596  degrees of freedom
## AIC: 1521.7
##
## Number of Fisher Scoring iterations: 6
```

```
#The best model is the one with both predictors but without the interaction
exp(coef(mod3ex5))
```

```
## (Intercept)      State1  Alcoholic1
##  0.05580478  0.50593261  9.05654116

#Another way to calculate the regression with a grouped data in binary variables
state<-c(0,1,0,1)
alcoholic<-c(0,0,1,1)
sick<-c(1,1,1,1)
proportions<-c(0.05,0.0278,0.35,0.1974)
weights<-c(500,3772,100,228)

data5_2<-as.data.frame(cbind(state,alcoholic,sick,proportions,weights))

mod5ex5<-glm(alcoholic/sick~state, family=binomial, weights=weights, data=data5_
2)
summary(mod5ex5)

##
## Call:
## glm(formula = alcoholic/sick ~ state, family = binomial, data = data5_2,
##     weights = weights)
##
## Deviance Residuals:
##      1       2       3       4
## -13.50  -21.04   18.93   36.14
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.6094     0.1095 -14.692   <2e-16 ***
## state        -1.1966     0.1290  -9.273   <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: 2364.4  on 3  degrees of freedom
## Residual deviance: 2289.7  on 2  degrees of freedom
## AIC: 2293.7
##
## Number of Fisher Scoring iterations: 6
```

The best model doing the regular fitted regression (without grouped data) is the one with both predictors but without the interaction (AIC=1519.9).

Then, the odds of the coefficients obtained through the summary of the model are the next:

- State1 (1=Philadelphia): 0.5059

- Alcoholic1 (1=Alcoholic): 9.0565

Calculating the probabilities with them, we have:

- State1 (1=Philadelphia): 0.3359

- Alcoholic1 (1=Alcoholic): 0.9000

These values mean that the possibility that a sick user is from Philadelphia is only a 0.3359, so New York is more vulnerable to have sick people.

On the other hand, the possibility that a sick user is alcoholic is 0.9, so alcoholic people is much more likely to suffer a cirrhosis.

Finally, evaluating the model of the grouped data in order to know if the location influences in the presence of the illness, we can conclude the same than before since the coefficient of the model summary is -1.1966 and that means as higher the level of the State, less likely to be sick, so, in Philadelphia is less common to suffer cirrhosis.

6. (0.5 points) Show that the Deviance in the case of Poisson regression is: $2\left[\sum_{i=1}^{n} y_i\, log\left(\frac{y}{\hat{\mu}}\right) - (y - \hat{\mu})\right]$

We know the deviance is defined as following:

$$D = -2log\left[\frac{\text{likelihood of the fitted model}}{\text{likelihood of the saturated model}}\right]$$

And the maximum likelihood for a Poisson distribution is:

$$\mathcal{L}(y_1, \dots, y_n) = \prod_{i=1}^{n} \frac{e^{-\mu}\mu^{y_i}}{y_i}$$

Then the log-likelihood for the saturated and fitted model is:

$$log\mathcal{L}(y_1, \dots, y_n) = \sum_{i=1}^{n} -\hat{\mu} + y_i \log(\hat{\mu}) - \log(y_i)$$

$$log\mathcal{L}(y_1, \dots, y_n) = \sum_{i=1}^{n} -y + y_i \log(y) - \log(y_i)$$

If we replace both likelihoods in the previous deviance expression, then, we have:

$$\text{Deviance} = 2\left[[\sum_{i=1}^{n} -y + y_i \log(y) - \log(y_i)] - [\sum_{i=1}^{n} -\hat{\mu} + y_i \log(\hat{\mu}) - \log(y_i)] \right]$$

So, finally:

$$= 2\left[\sum_{i=1}^{n} y_i \, log\left(\frac{y}{\hat{\mu}}\right) - (y - \hat{\mu})\right]$$

7. (1 point) Find the best model for the property crime rates used in chapter 6 and interpret the parameters.

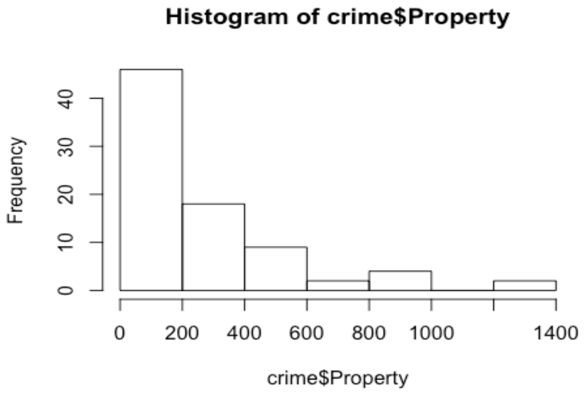
```
crime <- read.table("/Users/cortesocanamarta/Documents/Marta/MÁSTER DATA SCIENCE
/Regression Models/datasets/Campus_Crime.txt",header=TRUE)
```

The dataset contains the following variables:

- Type = college (C) or university (U)
- Region = region of the country (C = Central, MW = Midwest, NE = Northeast, SE = Southeast, SW = Southwest, and W = West)
- Violent = the number of violent crimes for that institution for the given year
- Property = the number of property crimes for that institution for the given year
- Enrollment = enrollment at the school

Let's plot the data first in order to obtain the distribution:

```
hist(x=crime$Property)
```



We can see that our data seems to follow a Poisson distribution.

For modelling this data we need to take into a account that it is not the same the impact of the number of crimes when the number of the enrollment is small than when the enrollment is big, as it is not comparable. We use the command "offset" for the regression model fit, then, we apply the log to the Enrollment because the term log(t) is referred to as an offset in the Poisson regression model for the expected rate of the occurrence of event:

$$\log(\mu) = \beta_0 + \beta_1 x + \log(t)$$

```
mod1ex7<-glm(Property~Type+Region,family=poisson,offset=log(Enrollment), data=cr
ime)
summary(mod1ex7)
```

```
##
## Call:
## glm(formula = Property ~ Type + Region, family = poisson, data = crime,
##     offset = log(Enrollment))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -14.638   -5.840   -1.714    2.700   21.361
##
## Coefficients:
##             Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -4.85393    0.02741 -177.105  < 2e-16 ***
## TypeU        0.64707    0.02311   28.002  < 2e-16 ***
## RegionMW     0.19275    0.02427    7.941 2.01e-15 ***
## RegionNE     0.31429    0.02356   13.340  < 2e-16 ***
## RegionSE     0.48617    0.02314   21.014  < 2e-16 ***
## RegionSW     0.11158    0.02908    3.837 0.000124 ***
## RegionW      0.31874    0.02624   12.145  < 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: 5979.2  on 80  degrees of freedom
## Residual deviance: 4585.5  on 74  degrees of freedom
## AIC: 5140.5
##
## Number of Fisher Scoring iterations: 5
```

The Region coefficients each compare the mean rate for that region to the Central region (the reference level). According to the p-values, all regions differ significantly from the Central region. The estimated coefficient of 0.48617 translates to the property crime rate (per 1,000), in the Southeast, being nearly 1.63 ≈ exp(0.48617) times than in the Central region for that type of school.

In order to check the effect of the type of the school (University or Collegue) we can include the interaction between Type and Region in the fit model.

```
mod2ex7<-glm(Property~Type+Region+Type:Region,family=poisson,offset=log(Enrollme
nt), data=crime)
summary(mod2ex7)
```

```
##
## Call:
## glm(formula = Property ~ Type + Region + Type:Region, family = poisson,
##     data = crime, offset = log(Enrollment))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -14.674   -5.951   -1.880    3.173   20.835
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -4.90062    0.06202 -79.020  < 2e-16 ***
## TypeU           0.69782    0.06460  10.802  < 2e-16 ***
## RegionMW       -0.42732    0.10189  -4.194 2.74e-05 ***
## RegionNE        1.12897    0.06947  16.252  < 2e-16 ***
## RegionSE        0.18179    0.08666   2.098 0.035914 *  
## RegionSW       -0.18535    0.11989  -1.546 0.122089    
## RegionW        -0.15011    0.08187  -1.833 0.066730 .  
## TypeU:RegionMW  0.66068    0.10493   6.296 3.05e-10 ***
## TypeU:RegionNE -0.99001    0.07412 -13.357  < 2e-16 ***
## TypeU:RegionSE  0.32943    0.08992   3.664 0.000249 ***
## TypeU:RegionSW  0.31548    0.12359   2.553 0.010689 *  
## TypeU:RegionW   0.57304    0.08640   6.633 3.30e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for poisson family taken to be 1)
## 
##     Null deviance: 5979.2  on 80  degrees of freedom
## Residual deviance: 3735.4  on 69  degrees of freedom
## AIC: 4300.5
## 
## Number of Fisher Scoring iterations: 5
```

As we can see, the AIC is much lower in the model with the interaction than in the first model, so it confirms the significance of the contribution of the interaction to this model and then, the last model is the best.

Particulary, we can interpret the coefficients related to the type of school in such a way that the only region where the crime is higher in collegues than in universities is Northeast, since the value of the exp(0.69782-0.99001) is less than 1 (0.7466). That means the crime is 1.34 times higher in collegues than universities in the Northeast, in the rest of the regions this value is higher than 1, then, they crime in universities is more likely than in collegues.

8. (2 points) The dataset ships2 concern a type of damage caused by waves to the forward section of cargo-carrying vessels. The variables are:

- incidents: the number of damage incidents (you need to remo)
- service: aggregate month of service
- period: period of operation (to be defined as a factor): 1960-74, 75-79
- year: year of construction (to be defined as a factor): 1960-64, 65-69, 70-74, 75-79
- type: A to E

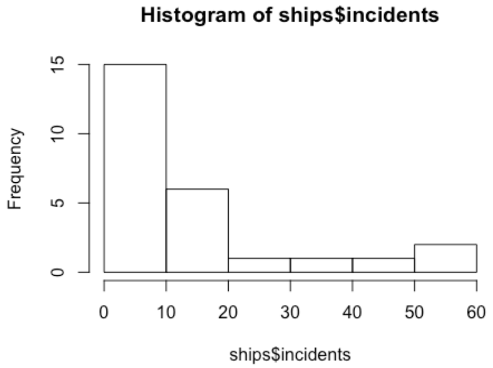
Develop a model for the rate of incidents per aggregate months of service. Check and correct for overdispersion (if necessary). Given the final model, answer the following questions:

- Which type of ship has the lowest and the highest risk of incidents?
- By how much does the incident rate increases after 1974?
- In which year where built the safest ships?

```
ships <- read.table("/Users/cortesocanamarta/Documents/Marta/MÁSTER DATA SCIENCE
/Regression Models/datasets/Ships2.txt",header=TRUE)
```

Let's first plot and check the distribution followed by the response and calculate the variance and the mean for the response in order to check if there is overdispersion.

```
hist(x=ships$incidents)
```



```
respmean<-mean(ships$incidents)
respvar<-var(ships$incidents)
```

The distribution seems a Poisson one and, in fact, we have overdispersion since the variance in the response is 280.94 and the mean is 13.69. So, we need to include a dispersion parameter (ϕ).

We can estimate a dispersion parameter, by dividing the model deviance by its corresponding degrees of freedom:

$$\hat{\phi} = \frac{\text{Deviance}}{n - (k+1)}$$

Or we can calculate it directly through the function of the fit model using the quasipoisson family in R. But first of all, we should transform the type of the element of the year and period variable into factors.

```
ships$year<-as.factor(ships$year)
ships$period<-as.factor(ships$period)

mod1ex8 = glm(incidents~type+year+period,family=quasipoisson, offset=log(service
),data=ships)
summary(mod1ex8)

## 
## Call:
## glm(formula = incidents ~ type + year + period, family = quasipoisson, 
##     data = ships, offset = log(service))
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.8793  -0.7254   0.0063   0.7267   2.6054  
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -6.35173    0.31263 -20.317 2.31e-13 ***
## typeB       -0.58015    0.25467  -2.278  0.03593 *  
## typeC       -0.58755    0.46955  -1.251  0.22777    
## typeD        0.08396    0.41787   0.201  0.84315    
## typeE        0.31729    0.33628   0.944  0.35863    
## year65       0.70468    0.21376   3.297  0.00426 ** 
## year70       0.77731    0.24442   3.180  0.00548 ** 
## year75       0.39347    0.33801   1.164  0.26047    
## period75     0.36989    0.16882   2.191  0.04267 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 2.03308)
##
##     Null deviance: 138.221  on 25  degrees of freedom
## Residual deviance:  29.664  on 17  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 5
```

The estimated dispersion parameter here is much larger than 1 (3.066) indicating overdispersion (extra variance) which has to be taken into consideration.

The models fitted using the quasipoisson distribution should be compared using a F test instead of a χ^2.

Other way to treat with overdispersion is applying a Negative Binomial model to the response instead of a Poisson. Let's model it in that way:

```
library(MASS)
mod2ex8 = glm.nb(incidents~type+year+period, weights=offset(log(service)),data=s
hips)
summary(mod2ex8)
```

```
##
## Call:
## glm.nb(formula = incidents ~ type + year + period, data = ships,
##     weights = offset(log(service)), init.theta = 6.124651285,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.2985  -1.7704  -0.7857   1.6894   4.7444
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.89858    0.14359  13.222  < 2e-16 ***
## typeB        1.46021    0.10659  13.700  < 2e-16 ***
## typeC       -1.45238    0.15916  -9.125  < 2e-16 ***
## typeD       -0.38423    0.15978  -2.405  0.01618 *
## typeE       -0.34726    0.13085  -2.654  0.00796 **
## year65       0.23432    0.11956   1.960  0.05001 .
## year70       0.26318    0.11803   2.230  0.02576 *
## year75      -0.44304    0.14633  -3.028  0.00246 **
## period75     0.25655    0.08168   3.141  0.00168 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(6.1247) family taken to be 1)
##
##     Null deviance: 1043.39  on 25  degrees of freedom
## Residual deviance:  198.24  on 17  degrees of freedom
## AIC: 1213
##
```

```
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  6.125
##           Std. Err.:  1.000
##
##  2 x log-likelihood:  -1193.050

#AIC=1241
```

These results differ from the quasipoisson model. Some coefficients change the direction although the figures are similar in size and negative binomial standard errors are smaller. Regarding the significance, now, all of the predictors are significant with level 0.05.

Let's try different models with other subset of predictors and interactions in order to obtain the best one:

```
mod3ex8 = glm.nb(incidents~type+year, weights=offset(log(service)),data=ships)
summary(mod3ex8)

##
## Call:
## glm.nb(formula = incidents ~ type + year, data = ships, weights = offset(log(
service)),
##     init.theta = 5.476794175, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.8497  -1.9529  -0.5936   1.7782   3.9288
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.0033     0.1434  13.969  < 2e-16 ***
## typeB         1.4558     0.1104  13.182  < 2e-16 ***
## typeC        -1.4595     0.1631  -8.951  < 2e-16 ***
## typeD        -0.3835     0.1636  -2.343  0.01911 *
## typeE        -0.3594     0.1346  -2.670  0.00758 **
## year65        0.2540     0.1250   2.031  0.04226 *
## year70        0.3396     0.1229   2.762  0.00574 **
## year75       -0.2860     0.1459  -1.960  0.04997 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(5.4768) family taken to be 1)
##
##     Null deviance: 969.50  on 25  degrees of freedom
## Residual deviance: 194.29  on 18  degrees of freedom
## AIC: 1220.1
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  5.477
##           Std. Err.:  0.858
##
##  2 x log-likelihood:  -1202.051
```

```
#(AIC=1220)

mod4ex8 = glm.nb(incidents~type+period, weights=offset(log(service)),data=ships)
summary(mod4ex8)
```

```
##
## Call:
## glm.nb(formula = incidents ~ type + period, data = ships, weights = offset(lo
g(service)),
##     init.theta = 4.659012395, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.2292  -1.9069  -0.9829   0.9654   4.8634
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.06254    0.10825  19.054   <2e-16 ***
## typeB        1.46988    0.11184  13.143   <2e-16 ***
## typeC       -1.43811    0.16313  -8.816   <2e-16 ***
## typeD       -0.41218    0.16691  -2.469   0.0135 *
## typeE       -0.26173    0.14062  -1.861   0.0627 .
## period75     0.14077    0.08448   1.666   0.0957 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(4.659) family taken to be 1)
##
##     Null deviance: 868.36  on 25  degrees of freedom
## Residual deviance: 201.10  on 20  degrees of freedom
## AIC: 1242.5
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  4.659
##           Std. Err.:  0.698
##
##  2 x log-likelihood:  -1228.503
```

```
# (AIC=1243)

mod5ex8 = glm.nb(incidents~type+year+period+type:year, weights=offset(log(servic
e)),data=ships)
summary(mod5ex8)
```

```
##
## Call:
## glm.nb(formula = incidents ~ type + year + period + type:year,
##     data = ships, weights = offset(log(service)), init.theta = 23.38397095,
##     link = log)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -5.630  -1.738   0.000   1.381   3.994
```

```
## 
## Coefficients: (4 not defined because of singularities)
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)   2.23605    0.53097   4.211 2.54e-05 ***
## typeB         1.16163    0.53338   2.178  0.02942 *  
## typeC        -2.39790    0.45066  -5.321 1.03e-07 ***
## typeD        -1.01160    0.23604  -4.286 1.82e-05 ***
## typeE        -2.39790    0.42775  -5.606 2.07e-08 ***
## year65       -1.15485    0.55184  -2.093  0.03637 *  
## year70        0.07631    0.53775   0.142  0.88715    
## year75       -0.15665    0.51411  -0.305  0.76059    
## period75      0.31849    0.05572   5.716 1.09e-08 ***
## typeB:year65  1.62796    0.55770   2.919  0.00351 ** 
## typeC:year65  0.99820    0.62228   1.604  0.10869    
## typeD:year65       NA         NA      NA       NA    
## typeE:year65  3.10392    0.47017   6.602 4.06e-11 ***
## typeB:year70 -0.33545    0.54498  -0.616  0.53821    
## typeC:year70  1.27242    0.48188   2.641  0.00828 ** 
## typeD:year70  0.43087    0.28034   1.537  0.12430    
## typeE:year70  2.05380    0.44947   4.569 4.89e-06 ***
## typeB:year75 -0.66915    0.52750  -1.269  0.20461    
## typeC:year75       NA         NA      NA       NA    
## typeD:year75       NA         NA      NA       NA    
## typeE:year75       NA         NA      NA       NA    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for Negative Binomial(23.384) family taken to be 1)
## 
##     Null deviance: 2077.99  on 25  degrees of freedom
## Residual deviance:  155.84  on  9  degrees of freedom
## AIC: 1069.2
## 
## Number of Fisher Scoring iterations: 1
## 
## 
##               Theta:  23.38 
##           Std. Err.:  5.67 
## 
##  2 x log-likelihood:  -1033.169
```

```r
#the best so far (AIC=1069)

mod6ex8 = glm.nb(incidents~type+year+period+type:period+type:year, weights=offset(log(service)),data=ships)
summary(mod6ex8)
```

```
## 
## Call:
## glm.nb(formula = incidents ~ type + year + period + type:period + 
##     type:year, data = ships, weights = offset(log(service)), 
##     init.theta = 40.51420085, link = log)
## 
## Deviance Residuals: 
##       1        2        3        4        5        6        7        8  
##  1.5628  -1.1586  -0.9675   0.6229   0.0000   2.9921  -3.3526   1.9387 
##       9       10       11       12       13       14       15       16 
## -2.0369  -7.3078   5.2519   0.0000  -0.8864   1.2260   0.0000   0.4460 
##      17       18       19       20       21       22       23       24 
## -0.6463   0.0000   0.0000   0.0000   0.0000   1.5298  -1.4313  -1.5173 
##      25       26 
##  1.1055   0.0000 
## 
## Coefficients: (4 not defined because of singularities)
##                Estimate Std. Error z value Pr(>|z|)    
## (Intercept)      0.9788     0.5806   1.686   0.0918 .  
## typeB            2.4793     0.5836   4.248 2.16e-05 ***
## typeC           -0.6585     0.5436  -1.211   0.2257    
## typeD           -1.8334     0.4268  -4.296 1.74e-05 ***
## typeE           -1.9959     0.4756  -4.196 2.71e-05 ***
## year65          -0.2529     0.5801  -0.436   0.6629    
## year70           0.9664     0.5670   1.704   0.0883 .  
## year75           0.5361     0.5449   0.984   0.3252    
## period75         0.8829     0.1588   5.559 2.71e-08 ***
## typeB:period75  -0.7142     0.1694  -4.217 2.48e-05 ***
## typeC:period75  -1.7394     0.3130  -5.558 2.73e-08 ***
## typeD:period75   0.8218     0.3621   2.269   0.0232 *  
## typeE:period75  -0.4020     0.2202  -1.825   0.0679 .  
## typeB:year65     0.7316     0.5842   1.252   0.2104    
## typeC:year65     0.7890     0.6160   1.281   0.2003    
## typeD:year65         NA         NA      NA       NA    
## typeE:year65     2.9662     0.4687   6.329 2.47e-10 ***
## typeB:year70    -1.1807     0.5722  -2.063   0.0391 *  
## typeC:year70     0.4287     0.5087   0.843   0.3994    
## typeD:year70     0.5813     0.2772   2.097   0.0360 *  
## typeE:year70     1.9219     0.4476   4.294 1.76e-05 ***
## typeB:year75    -1.2727     0.5563  -2.288   0.0222 *  
## typeC:year75         NA         NA      NA       NA    
## typeD:year75         NA         NA      NA       NA    
## typeE:year75         NA         NA      NA       NA    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for Negative Binomial(40.5142) family taken to be 1)
## 
##     Null deviance: 2483.98  on 25  degrees of freedom
## Residual deviance:  125.01  on  5  degrees of freedom
## AIC: 1016.4
## 
## Number of Fisher Scoring iterations: 1
## 
## 
##               Theta:  40.5 
##           Std. Err.:  12.6 
## 
##  2 x log-likelihood:  -972.441

#better than the previous one (AIC=1016)
```

We can conclude that the best model is **mod6ex8**, the one with the next subset of predictors: incidents ~ type + year + period + type:period + type:year.

Let's now calculate the predictions in order to answer the questions:

```
pred<-predict(mod6ex8,type="response")
pred

##          1          2          3          4          5          6          7
##  2.0667660  4.9974652  6.9953596 16.9148641 11.0000000 31.7587235 37.5971539
##          8          9         10         11         12         13         14
## 51.2612982 60.6850245 25.6338069 30.3462506 18.0000000  1.3775926  0.5850241
##         15         16         17         18         19         20         21
##  1.0000000  5.5592071  2.3608360  1.0000000  2.0000000 11.0000000  4.0000000
##         22         23         24         25         26
##  5.4532079  8.8214607  6.4960292 10.5083957  1.0000000

which.min(pred)

## 14
## 14

#the one of the position 14

which.max(pred)

## 9
## 9

#the one of the position 9

newdata<-as.data.frame(cbind(ships$type,ships$year,ships$period,ships$service,pr
ed))
names(newdata)[1]<-c("type")
names(newdata)[2]<-c("year")
names(newdata)[3]<-c("period")
names(newdata)[4]<-c("service")

above74<-newdata[newdata$year==4,]
mean_above74<-mean(above74$pred)
at7074<-newdata[newdata$year==3,]
mean_at7074<-mean(at7074$pred)
mean_above74-mean_at7074

## [1] -4.781475

newdata[,6]<-as.double(newdata$pred/newdata$service)
a<-newdata %>% group_by(year) %>% summarise_all(mean)
#year 1 (60-64)
```

The ship which has the lowest risk of incidents is the **type C** (the one in the position 14), since it has the lowest coefficient in the prediction.

Then, the ship which has the highest risk of incidents is the **type B**.

On the other hand, the incident rate decrease in 4.78 units in ships built after 1974.

Finally, the safest ships where built in **1960-1964**.

9. (0.5 points) In the zero-inflated model, show why: $Pr(Y = 0) = \phi + (1 - \phi) * e^{-\mu}$.

Since Y is a mixture of random variables following a Poisson distribution and Y takes value x with probability ϕ and 0 with probability $1 - \phi$.

Then,

$$P(Y = 0) = \phi + (1 - \phi)P(X = 0)$$

$$P(X = x|\mu) = \frac{e^{-\mu}\mu^{x}}{x!}$$

$$P(Y = 0) = \phi + (1 - \phi)\frac{e^{-\mu}\mu^{0}}{0!}$$

$$= \phi + (1 - \phi)e^{-\mu}$$

10. (2.5 points) The aim of this task is to examine the relationship between the number of physician office visits for a person (ofp) and a set of explanatory variables for individuals on Medicare. Their data are contained in the file dt.csv.

The explanatory variables are:

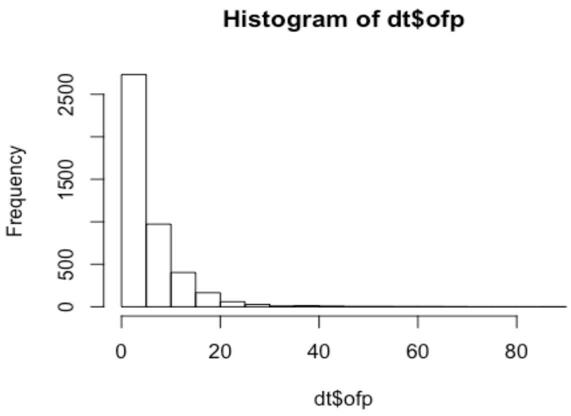
• number of hospital stays (hosp)
• number of chronic conditions (numchron)
• gender (male = 1, female = 0) • number of years of education (school)
• private insurance (yes = 1, no = 0)
• health excellent and health poor: these two are self-perceived health status indicators that take on a value of yes = 1 or no = 0, and they cannot both be 1 (both equal to 0 indicates “average” health).

Using these data, complete the following:

• Estimate the Poisson regression model to predict the number of physician office visits and interpret the coefficients
• Compare the number of zero-visit counts in the data to the number predicted by the model and comment. Can you think of a possible explanation for why there are so many zeroes in the data?
• Estimate the zero-inflated Poisson regression model to predict the number of physician office visits. Use all of the explanatory variables for the $\log(\mu)$ part of the model and no explanatory variables in the ϕ part of the model. Interpret the model fit results • Do the previous item again, but now use all of the explanatory variables to estimate ϕ. Interpret the model fit results and compare this model to the previous ZIP model using a LRT • Examine how well each model estimates the number of 0 counts

```
dt<-read.csv("/Users/cortesocanamarta/Documents/Marta/MÁSTER DATA SCIENCE/Regres
sion Models/datasets/dt.csv")

dt$gender<-as.factor(dt$gender)
dt$privins<-as.factor(dt$privins)
dt$health_excellent<-as.factor(dt$health_excellent)
dt$health_poor<-as.factor(dt$health_poor)
```

```
hist(dt$ofp)
```



```
mod1ex10 <- glm(ofp~.,family=poisson,dt)
summary(mod1ex10)

##
## Call:
## glm(formula = ofp ~ ., family = poisson, data = dt)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.4055  -1.9962  -0.6737   0.7049  16.3620
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)         1.028874   0.023785  43.258   <2e-16 ***
## hosp                0.164797   0.005997  27.478   <2e-16 ***
## numchron            0.146639   0.004580  32.020   <2e-16 ***
## gender1            -0.112320   0.012945  -8.677   <2e-16 ***
## school              0.026143   0.001843  14.182   <2e-16 ***
## privins1            0.201687   0.016860  11.963   <2e-16 ***
## health_excellent1 -0.361993   0.030304 -11.945   <2e-16 ***
## health_poor1        0.248307   0.017845  13.915   <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: 26943  on 4405  degrees of freedom
## Residual deviance: 23168  on 4398  degrees of freedom
## AIC: 35959
##
## Number of Fisher Scoring iterations: 5

# Goodness of fit test
gof.pvalue = 1 - pchisq(mod1ex10$deviance, mod1ex10$df.residual)
gof.pvalue

## [1] 0
```

```
exp(coef(mod1ex10))

##       (Intercept)              hosp          numchron           gender1
##         2.7979142         1.1791542         1.1579362         0.8937583
##            school          privins1 health_excellent1      health_poor1
##         1.0264877         1.2234649         0.6962871         1.2818534

predofp<-predict(mod1ex10)
predofp<-sum(1/exp(exp(predofp)))

length(which(dt$ofp==0))

## [1] 683

#683 zeros

predofp

## [1] 46.71402

#47 zeros
```

The Poisson regression model estimated doesn't fit the data very well (the AIC is too large) and the goodness-of-fit tell the test is statistically significant (value 0) which remains lack-of-fit.

In any case, the coefficients means that the higher the number of hospital stays and chronic illness, the more physician office visits (1.18 and 1.16 times more, respectively).

In the same way, if the person has private insurance and feel poor health, they also are more likely to have office visits (1.22 and 1.28 times more, respectively).

On the contrary, the males have less opfs than females (1.12 times less) and the people with excellent health are less likely to go to the physician office (1.44 times less).

This model predicts 47 zero visits to the physician office while in the original data there are 683 zeros.

The high number of zeros in the data can be because many of the people is treated by the doctor at their own houses or because many of these people has private insurance and they prefer go to the private doctor.

```
#Zero inflated regression model
library(pscl)

## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis

mod2ex10<-zeroinfl(ofp~ hosp+numchron+gender+school+privins+health_excellent+hea
lth_poor | 1, data=dt)
summary(mod2ex10)

##
## Call:
```

```
## zeroinfl(formula = ofp ~ hosp + numchron + gender + school + privins + 
##     health_excellent + health_poor | 1, data = dt)
## 
## Pearson residuals:
##     Min      1Q  Median      3Q     Max 
## -2.1641 -1.2121 -0.4525  0.5880 26.2757 
## 
## Count model coefficients (poisson with log link):
##                    Estimate Std. Error z value Pr(>|z|)    
## (Intercept)        1.391755   0.024536  56.723  < 2e-16 ***
## hosp               0.159132   0.006059  26.265  < 2e-16 ***
## numchron           0.103486   0.004738  21.843  < 2e-16 ***
## gender1           -0.065252   0.013122  -4.973 6.60e-07 ***
## school             0.019712   0.001887  10.447  < 2e-16 ***
## privins1           0.085868   0.017328   4.955 7.22e-07 ***
## health_excellent1 -0.319184   0.031852 -10.021  < 2e-16 ***
## health_poor1       0.254342   0.017720  14.354  < 2e-16 ***
## 
## Zero-inflation model coefficients (binomial with logit link):
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -1.74045    0.04327  -40.22   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## Number of iterations in BFGS optimization: 14 
## Log-likelihood: -1.63e+04 on 9 Df
```

```
exp(coef(mod2ex10))
```

```
##       count_(Intercept)              count_hosp          count_numchron 
##               4.0219035               1.1724922               1.1090304 
##           count_gender1            count_school          count_privins1 
##               0.9368313               1.0199078               1.0896624 
## count_health_excellent1      count_health_poor1        zero_(Intercept) 
##               0.7267416               1.2896123               0.1754406
```

```
mod3ex10<-zeroinfl(ofp~ hosp+numchron+gender+school+privins+health_excellent+health_poor |  hosp+numchron+gender+school+privins+health_excellent+health_poor, data=dt)
summary(mod3ex10)
```

```
## 
## Call:
## zeroinfl(formula = ofp ~ hosp + numchron + gender + school + privins + 
##     health_excellent + health_poor | hosp + numchron + gender + school + 
##     privins + health_excellent + health_poor, data = dt)
## 
## Pearson residuals:
##     Min      1Q  Median      3Q     Max 
## -5.4092 -1.1579 -0.4769  0.5435 25.0380 
## 
## Count model coefficients (poisson with log link):
##                    Estimate Std. Error z value Pr(>|z|)    
## (Intercept)        1.405812   0.024175  58.152  < 2e-16 ***
## hosp               0.159011   0.006060  26.239  < 2e-16 ***
## numchron           0.101836   0.004721  21.571  < 2e-16 ***
## gender1           -0.062332   0.013054   -4.775 1.80e-06 ***
## school             0.019144   0.001873   10.221  < 2e-16 ***
## privins1           0.080557   0.017145    4.699 2.62e-06 ***
## health_excellent1 -0.304134   0.031151   -9.763  < 2e-16 ***
## health_poor1       0.253454   0.017705   14.315  < 2e-16 ***
##
## Zero-inflation model coefficients (binomial with logit link):
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -0.08102    0.14233   -0.569 0.569219
## hosp              -0.30330    0.09158   -3.312 0.000927 ***
## numchron          -0.53117    0.04601 -11.545  < 2e-16 ***
## gender1            0.41527    0.08919    4.656 3.22e-06 ***
## school            -0.05677    0.01223   -4.640 3.49e-06 ***
## privins1          -0.75294    0.10257   -7.341 2.12e-13 ***
## health_excellent1  0.23786    0.14990    1.587 0.112550
## health_poor1       0.02166    0.16170    0.134 0.893431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of iterations in BFGS optimization: 24
## Log-likelihood: -1.613e+04 on 16 Df
```

```
exp(coef(mod3ex10))
```

```
##       count_(Intercept)              count_hosp          count_numchron
##               4.0788391               1.1723503               1.1072014
##           count_gender1            count_school          count_privins1
##               0.9395709               1.0193280               1.0838910
## count_health_excellent1      count_health_poor1        zero_(Intercept)
##               0.7377620               1.2884684               0.9221782
##               zero_hosp           zero_numchron            zero_gender1
##               0.7383796               0.5879169               1.5147765
##             zero_school           zero_privins1  zero_health_excellent1
##               0.9448131               0.4709786               1.2685321
##       zero_health_poor1
##               1.0218989
```

```
lrt <- 2*(mod3ex10$loglik-mod2ex10$loglik)
df=mod3ex10$df.residual-mod2ex10$df.residual
1-pchisq(lrt,abs(df))
```

```
## [1] 0
```

```
pred_zeroinf1<-sum(predict(mod2ex10, type="prob")[,1])

pred_zeroinf2<-sum(predict(mod3ex10, type="prob")[,1])
```

The fitted results of the model, with no explanatory variable in the ϕ part of the model, consider significant all the predictors. The zero_(Intercept) coefficient means that the model only predict zeros with 0.175 of accuracy.

Then, comparing it with the model with all the explanatory variables in both parts, we can say, since the LRT is 0, that the model with all explanatory variables in both sides of the fit function (the completed model) is the one we should consider since it fits better the data.

Finally, the model with all explanatory variables predicts 682.8 zeros while the model without the explanatory variables in the binomial part, predicts only 672 zeros which is lower than the number of zeros of the original data (683).