ETH Zürich, D-MATH
HS 2021
Prof. Dr. Mario V. Wüthrich

Non-Life Insurance: Mathematics and Statistics

Exercise sheet 11

Exercise 11.1 Claim Frequency Modeling with GLM (R Exercise)
Suppose that a motorbike insurance portfolio of an insurance company has been divided according to three tariff criteria

- vehicle class: {weight over 60 kg and more than two gears, other},
- vehicle age: {at most one year, more than one year},
- geographic zone: {large cities, middle-sized towns, smaller towns and countryside}.

Assume that we observed the following claim frequencies:

class	age	zone	volume	number of claims	claim frequency
1	1	1	100	25	0.250
1	1	2	200	15	0.075
1	1	3	500	15	0.030
1	2	1	400	60	0.150
1	2	2	900	90	0.100
1	2	3	7'000	210	0.030
2	1	1	200	45	0.225
2	1	2	300	45	0.150
2	1	3	600	30	0.050
2	2	1	800	80	0.100
2	2	2	1'500	120	0.080
2	2	3	5'000	90	0.018

Table 1: Observed volumes, numbers of claims and claim frequencies in the $2 \cdot 2 \cdot 3 = 12$ risk classes.

(a) Perform a GLM analysis for the claim frequencies using the Poisson model. Comment on the results.

(b) Plot the observed and the fitted claim frequencies against the vehicle class, the vehicle age and the geographic zone.

(c) Create a Tukey-Anscombe plot of the deviance residuals versus the fitted expected numbers of claims.

(d) Is there statistical evidence that the classification into the geographic zones could be omitted?

Exercise 11.2 Claim Frequency Modeling with Neural Networks (R Exercise)
In this exercise we consider the French motor third-party liability insurance data set prepared in Listing 1. We model the claim frequencies using the three continuous but categorized tariff criteria

power of the car, age of the car and age of the driver.

(a) Write an R code that performs a GLM analysis for the claim frequencies on the data `trainset` using the Poisson model. Calculate the deviance statistics of the resulting GLM model on both the data sets `trainset` and `testset`.

(b) Write an R code that models the claim frequencies on the data `trainset` using a neural network with two hidden layers with $(r_1, r_2) = (20, 10)$ hidden neurons. Choose the hyperbolic tangent activation function and 100 gradient descent steps. Calculate the deviance statistics of the resulting neural network model on both the data sets `trainset` and `testset`. Compare the results to the values obtained in part (a).

(c) Repeat the neural network fitting procedure of part (b) with 1'000 gradient descent steps instead of 100. Calculate the deviance statistics of the resulting neural network model on both the data sets `trainset` and `testset`. What do you observe?

Listing 1: R code for Exercise 11.2.

```
### Dataset preparation
install.packages("CASdatasets",repos="http://dutangc.free.fr/pub/RRepos/", type="source")
lapply(c("CASdatasets", "keras", "plyr"), require, character.only=TRUE)
data("freMTPL2freq")
data <- freMTPL2freq[,c(1:6)]
data$VehPower <- relevel(as.factor(data$VehPower), ref="6")
VehAgeCat <- cbind(c(0:100), c(1,rep(2,3),rep(3,2),rep(4,2),rep(5,3),rep(6,2),rep(7,2),rep(8,3),
                               rep(9,83)))
data$VehAge <- relevel(as.factor(VehAgeCat[data$VehAge+1,2]), ref="2")
DrivAgeCat <- cbind(c(18:100), c(rep(1,21-18),rep(2,26-21),rep(3,31-26),rep(4,41-31),
                                 rep(5,51-41),rep(6,71-51),rep(7,101-71)))
data$DrivAge <- relevel(as.factor(DrivAgeCat[data$DrivAge-17,2]), ref="6")

### Training set and test set
set.seed(100)
train <- sample(1:nrow(data),round(0.5*nrow(data)))
trainset <- ddply(data[train,], .(VehPower,VehAge,DrivAge), summarise, ClaimNb=sum(ClaimNb),
                  Exposure=sum(Exposure))[,c(4:5,1:3)]
testset <- ddply(data[-train,], .(VehPower,VehAge,DrivAge), summarise, ClaimNb=sum(ClaimNb),
                 Exposure = sum(Exposure))[,c(4:5,1:3)]
```

Exercise 11.3 Claim Severity Modeling with GLM (R Exercise)
In this exercise we consider the French motor third-party liability insurance data set prepared in Listing 2. This time we model the claim severities using the three (categorical) tariff criteria

- area code: {A, B, C, D, E, F},
- brand of the vehicle: {B1, B10, B11, B12, B13, B14, B2, B3, B4, B5, B6},
- diesel/fuel: {diesel, regular fuel}.

(a) Perform a GLM analysis for the claim severities using the gamma model with log-link function. Comment on the results.

(b) Is there statistical evidence that the area code could be omitted as tariff criterion?

Listing 2: R code for Exercise 11.3.

```
# install.packages("CASdatasets",repos="http://dutangc.free.fr/pub/RRepos/", type="source")
lapply(c("CASdatasets", "plyr"), require, character.only=TRUE)
data("freMTPL2freq")
data("freMTPL2sev")
data <- freMTPL2sev[is.element(freMTPL2sev$IDpol,freMTPL2freq$IDpol),]
data <- ddply(data, .(IDpol), summarize, ClaimAmount=sum(ClaimAmount))
data <- cbind(freMTPL2freq[is.element(freMTPL2freq$IDpol,data$IDpol),-3],data[,2])
colnames(data)[12]<- "ClaimAmount"
data <- ddply(data, .(Area, VehBrand, VehGas), summarize, ClaimNb=sum(ClaimNb),
              ClaimAmount=sum(ClaimAmount))
data$ClaimAmount <- data$ClaimAmount/data$ClaimNb
```

Exercise 11.4 Neural Networks and Gradient Descent
We assume that we have M independent large claims $\boldsymbol{Y} = (Y_1, \ldots, Y_M)$ with corresponding covariates $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_M \in \mathcal{Z} \subset \mathbb{R}^{r_0+1}$, where $r_0 = 1$ and $\boldsymbol{z}_m = (1, z_m)$, for all $m = 1, \ldots, M$. Let $\alpha : \mathcal{Z} \to \mathbb{R}_+$ be a given (but unknown) regression function. We assume that Y_m is Pareto distributed with threshold $\theta > 0$ and tail index parameter $\alpha(\boldsymbol{z}_m) > 0$, for all $m = 1, \ldots, M$. The goal is to model the regression function $\alpha : \mathcal{Z} \to \mathbb{R}_+$ with a neural network.

(a) Set up a single hidden layer neural network with $r_1 \in \mathbb{N}$ hidden neurons for this regression problem using the hyperbolic tangent activation function. How many parameters does this model have?

(b) Calculate the deviance statistics for this regression problem.

(c) Assume that we have a large number of hidden neurons. Why are we in this situation in general not interested in finding the maximum likelihood estimator? What alternative solution do you propose?

(d) Calculate one step of the gradient descent optimization algorithm explicitly for the single hidden layer neural network defined in part (a) and the deviance statistics loss function derived in part (b).