

Modèles de Régression Régularisée

Logistic regression

October 2023

Aim of the Practical session

- logistic regression, ℓ_1 penalized logistic regression.

Remarks

- The work has to be carried out by a team of 2 students and **R studio** is used to perform the practical sessions.
- A report should be written **only for exercice II**, automatically generated using a **R markdown** file format for 'R studio'.
- The 'R markdown file' and the corresponding pdf file have to be uploaded **before next project session on the ENSIIE project web site in the folder MRR2022tp3**.

I. Health application: diagnosis of a heart attack

In this application, the target variable Y is a binary variable with values in $\{0, 1\}$. The goal is here to build a machine able to predict the value of the target variable Y given the values of p co-variables X_1, \dots, X_p . We note the probability $\pi(\beta) = P(Y = 1|X = x)$ with $x = (x_1, \dots, x_p) \in R^p$ where β are the parameters of the model (machine). The generalized linear model (GLM) is here defined with $Z(\beta) = \log\left(\frac{\pi(\beta)}{1-\pi(\beta)}\right) = \beta^t X$ où $\beta \in R^p$ where β are the model parameters. The parameters β are estimated by the Maximum Likelihood Method with an historical data set. This model is called the **logistic regression model** and is very used for medical applications or for Scoring applications.

Applications: The file "SAHeart.txt" contains a set of data concerning a medical study of cardiac disease in South Africa for $n = 462$ individuals. For information, these data are also available on the web site: <http://www-stat.stanford.edu/~tibs/ElemStatLearn>. The file "SAHeart.info" describes the variables and the file "SaHeart.txt" contains the numerical values. Analyse briefly the two files. The "chd" variable is here the target variable Y , indicating if the person has already have an heart attack ($chd = 1$) or not ($chd = 0$). The other variables are potential co-variables use to explain Y .

- Load and visualize the set of data in the R studio environment.
- Visualize the data with a scatterplot graph where the binary value of the target variable is represented using a given color with the following instruction.

```
pairs(tab, pch=22, bg=c("red", "blue")[unclass(factor(tab[, "chd"]))])
```

- What can you say on the various joint distributions ?

A. Logistic regression model.

- a) The R function `glm()` is used to estimate the parameters of a generalized linear model.
 - Load the “glm help” file to study the inputs, outputs and the parameters of the function.
 - Then, use the `res=glm(...,family=binomial)` instruction with appropriate parameters to perform a logistic regression on the set of Heart data. With the help of the `glm` function, explain the use of the `family=binomial` option.
 - Execute the instruction: `summary(res)`.
- b) Study (briefly) the different fields of the R object returned by the function `glm()`, where the object `res`, is computed by using instructions `res=glm()` and `attributes(res)`.
 - Analyze the values of the coefficients estimated by the `glm()` procedure.
 - Using the previous results can we deduce the most significant coefficients ? the less ? Justify your answer.
 - Compute for each observation i the values predicted by the calibrated model $\hat{\beta}^t x_i$ using the `predict()` function (more precisely `predict.glm()`) with the correct parameters. Please note the differences using the options `type="link"` or `type="response"`.
- b) Odd-ratios. With the help of the slides of the MRR lecture, compute the **odds-ratio** for the different parameters then comment the obtained results. What are the limit of this approach? Compute then comment the value of the odd-ratio for the “tobacco” variable.

B. Performances of the classification model

Confusion matrix, TP, TN, FP, FN.

- a) Compute for all the observations of the data set, the binary response using the MAP criteria (MAP: Maximum A Posteriori) for a model using all the variables.
- b) Compute the confusion matrix with the help of the instruction `table()`. Compute the global performance and the error for the previous model, the False Positive $P(\hat{Y} = 1/Y = 0)$ and the False Negative $P(\hat{Y} = 0/Y = 1)$ rate. Conclusion.

K-fold.

Use a K-fold procedure to estimate the predictive power of the model. Use the `boxplot()` function to compute and visualize the different values of the performances (or the errors) computed for each fold. What information does this graph bring ? Compared the results for $K = 5$ and $K = 10$.

C. Model selection

1. Statistical approach: forward, backward, stepwise selection.

a) Model selection. Use the following instruction, to select the variables of the following models using the forward, backward and stepwise methods. Conclusion.

```
#Régression logistique Forward.
resall<-glm(chd~.,data=tab,family=binomial);
res0<-glm(chd~1,data=tab,family=binomial);
resfor<-step(res0,list(upper=resall),direction='forward')
```

```
##Régression logistique Backward
resback<-step(res,direction='backward')
print(resback)
```

```
#Régression logistique Stepwise
resstep<-step(res,direction='both');
print(resstep)
```

Use the instruction `formula()` to retrieve the final computed model.

2. Logistic regression with ℓ_1 or ℓ_2 penalizations.

In this part we are interested in the regularised methods **ridge** and **lasso** in order to constrain the variance of our estimator and control the variance of our estimator and -eventually- improve our prediction error. To generate these models we shall use the **glmnet** package. You will mostly need the **glmnet**, **predict**, **cv.glmnet** and **plot** functions of this package. Type `help(glmnet)`, `help(plot.glmnet)`, `help(cv.glmnet)` and `help(predict.glmnet)` to get help on these functions.

Use the **sample()** function to split your data in two sets: - the train test for model calibration and model selection (80% of data), and - the validation test, to evaluate your model (20% of data)

Transform the **familyhist** variable into a binary variable for ridge or lasso.

a) Ridge Regression Generate the Ridge regression model on the 10 variable ensemble. Trace the obtained regularisation path and comment on it. Select a λ through 10-fold cross-validation with the minimum and a “1 standard error” rules (the most penalized model with a 1 std distance from the model with the least error). We shall name the corresponding models **ridge.min** and **ridge.1se**

Take care with the input format for the **glmnet** function.

We can represent the regularisation path in function of different measurements:

The results are interchangeable for both methods of cross-validation.

We can access the generated models through the **predict** function. Take care as the new value of the predictors used for the prediction must be formatted as a matrix.

b) Lasso Regression Generate the Lasso regression model on the covariable ensemble. Trace the obtained regularisation path and comment on it. Select a λ through 10-fold cross-validation with the minimum and a “1 standard error” rules (the most penalized model with a 1 std distance from the model with the least error).

We name **lasso.min**, **lasso.1se**, **lasso.BIC** and **lasso.mBIC** the corresponding models.

Again the results remain close with both methods of cross-validation.

D. Conclusion. comparison of the model performances.

For each of your models, i.e. :

- The model corresponding to a constant (**null**),
- The model with all the predictors (**full**),
- The models obtained by using the stepwise methodology ,
- The models obtained by using the ridge methodology (**ridge.min**),
- The models obtained by using the lasso methodology (**lasso.min**),

estimate its precision errors with the help of the test dataset with a K-fold procedure. You can use the **predict** functions associated to the different objects you are manipulating.

II Cookies Study

As a data scientist, you are now asked to study the **Cookies.csv** dataset.

The dataset

Each cookie is characterized by a fat indicator (scalar value) and a vector of size $p = 700$. In the dataset, each row stores one cookie information: the fat indicator is available in the first column and the remaining values correspond to the spectra information. $n = 32$ cookies have been analysed.

Study

The aim of this study is to explain if, for one cookie, the fat is greater or not than the median of the fat values for each the cookies (binary target Ybin).

```
tab=read.csv(file="cookies.csv",sep=',',dec='.',header=TRUE);
YBin=as.numeric(tab$fat>median(tab$fat));
```

Logistic regression model using features

For this first study, the spectra of each cookie is sum-up in 5 characteristics (called features). which are the mean, the standard deviation, the slope, the minimum and the maximum of each spectra.

Study and comment a logistic model to explain if the fat is higher than the median value of all the fat indicators given the previous computed features.

Logistic regression model using the spectra

Study the use of the logistic regression (regular or ℓ_1 ℓ_2 penalized models) to explain the low or high values of the fat given the spectra (Ybin indicator).

```
tab=read.table(file="cookies.csv",header=TRUE);
YBin=as.numeric(tab$Y>median(tab$Y));
```