Régression logistique

Julien JACQUES

07/01/2020

Régression logistique

Régression logistique binaire

On cherche à expliquer :

$$Y = (y_1, \ldots, y_n)^t \qquad \text{où } y_i \in \{0, 1\}$$

à partir de p variables explicatives, que l'on supposera quantitatives :

$$X = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}$$

Régression logistique binaire

On souhaiterait effectuer des prédictions à l'aide d'un modèle linéaire que l'on maitrise bien :

$$\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}$$

Or ce modèle n'est pas adapté à $Y \in \{0, 1\}$.

Régression logistique binaire

On souhaiterait effectuer des prédictions à l'aide d'un modèle linéaire que l'on maitrise bien :

$$\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}$$

Or ce modèle n'est pas adapté à $Y \in \{0, 1\}$. Plutôt que de prédire Y, on va chercher à prédire la probabilité que $Y = 1$, où plus exactement :

$$logit(\pi(x)) = \ln \frac{\pi(x)}{1 - \pi(x)}$$

où

$$\pi(x) = P(Y = 1 | X = x)$$

qui prend ses valeurs sur $\mathbb{R}$ est pour lequel le modèle linéaire est bien adapté

Régression logistique binaire

Le modèle de régression logistique binaire s'écrit

$$logit(\pi(x)) = \beta_0 + \sum_{j=1}^{p} \beta_j x_j.$$

ou encore

$$\pi(x) = \frac{\exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right)}{1 + \exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right)}.$$

Odds-Ratio

On définit

$$\text{odds}(x) = \frac{\pi(x)}{1 - \pi(x)}$$

qui représente le rapport entre la probabilité d'avoir $Y = 1$ sur la probabilité d'avoir $Y = 0$ lorsque $X = x$.

On définit également

$$\text{odds-ratio}(x^1, x^2) = \frac{\text{odds}(x^1)}{\text{odds}(x^2)}$$

qui représente combien de fois on a plus de chance d'avoir $Y = 1$ au lieu d'avoir $Y = 0$ lorsque $X = x^1$ au lieu de $X = x^2$.

Les **odds-ratio permettent de quantifier l'impact des variables explicatives**.

Sélection de variables

Comme en régression linéaire :

- des tests de significativité des variables sont disponibles ($H_0 : \beta_j = 0$ contre $H_1 : \beta_j \neq 0$). On supprimera les variables non significatives.
- des algorithmes de sélection de variables sont disponibles

Régression logistique sous R

```r
reglog=glm((Species=="versicolor")~.,data=iris,family=binomial)
summary(reglog)
```

```
##
## Call:
## glm(formula = (Species == "versicolor") ~ ., family = binomia
##     data = iris)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1280  -0.7668  -0.3818   0.7866   2.1202
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    7.3785     2.4993   2.952 0.003155 **
## Sepal.Length  -0.2454     0.6496  -0.378 0.705634
## Sepal.Width   -2.7966     0.7835  -3.569 0.000358 ***
## Petal.Length   1.3136     0.6838   1.921 0.054713 .
## Petal.Width   -2.7783     1.1731  -2.368 0.017868 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

Régression logistique sous R

Effectuons une sélection de variables

```
library(MASS)
reglog2=stepAIC(reglog,trace = F)
summary(reglog2)
```

```
##
## Call:
## glm(formula = (Species == "versicolor") ~ Sepal.Width + Petal
##     Petal.Width, family = binomial, data = iris)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1262  -0.7731  -0.3984   0.8063   2.1562
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    6.9506     2.2261   3.122  0.00179 **
## Sepal.Width   -2.9565     0.6668  -4.434 9.26e-06 ***
## Petal.Length   1.1252     0.4619   2.436  0.01484 *
## Petal.Width   -2.6148     1.0815  -2.418  0.01562 *
```

Classification multi-classe : rég. log. polytomique

Cette fois $y_i \in \{1, \ldots, K\}$.

Dans cette situation, on se fixe une modalité de référence ($Y = K$ par exemple), et on réalise $K - 1$ régressions logistiques de $\pi_k(x)$ versus $\pi_K(x)$:

$$\ln \frac{\pi_k(x)}{\pi_K(x)} = \beta_{0k} + \sum_{j=1}^{p} \beta_{jk} x_j \qquad \forall 1 \leq k \leq K - 1.$$

Classification ordinale : rég. log. ordinale

Cette fois $y_i \in \{1, \ldots, K\}$, où un ordre existe entre les modalités 1 à K.

Dans cette situation, on modélise généralement des *logits cumulatifs* :

$$\ln \frac{\pi_{k+1}(x) + \ldots + \pi_K(x)}{\pi_1(x) + \ldots + \pi_k(x)} = \beta_{0k} + \sum_{j=1}^{p} \beta_{jk} x_j \qquad \forall 1 \leq k \leq K-1.$$

Ce dernier modèle comportant un grand nombre de paramètres, les β_{jk} sont souvent supposés constants par classe
$\beta_{jk} = \beta_j \quad \forall 1 \leq k \leq K-1.$

Régression logistique polytomique sous R

```
library(VGAM)
reglog=vglm(Species~.,data=iris,family=multinomial)
summary(reglog)
```

```
##
## Call:
## vglm(formula = Species ~ ., family = multinomial, data = iris
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept):1      35.490  22666.953      NA       NA
## (Intercept):2      42.638     25.708   1.659   0.0972 .
## Sepal.Length:1      9.495   6729.217      NA       NA
## Sepal.Length:2      2.465      2.394   1.030   0.3032
## Sepal.Width:1      12.300   3143.611      NA       NA
## Sepal.Width:2       6.681      4.480   1.491   0.1359
## Petal.Length:1    -22.975   4799.227  -0.005   0.9962
## Petal.Length:2     -9.429      4.737      NA       NA
## Petal.Width:1     -33.843   7583.502      NA       NA
## Petal.Width:2     -18.286      9.743      NA       NA
## ---
```

Régression logistique polytomique sous R

Comme l'espèce setosa est très bien séparée des autres, on a une infinité de modèle qui permettent de séparer parfaitement les setosa des autres, et par conséquent on ne peut réaliser de test de significativité pour cette modalité.

Régression logistique pénalisée

Comme en régression pénalisée, il est possible d'introduire des pénalités :

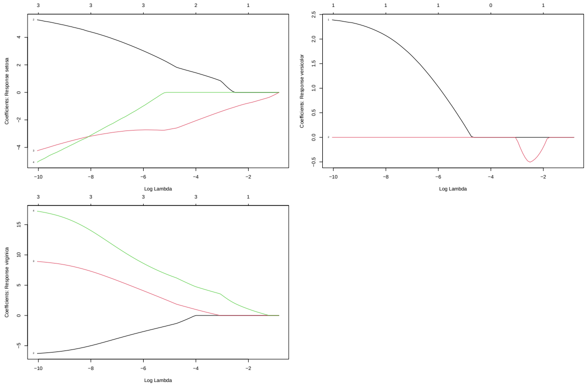
- LASSO
- ridge
- elasticnet

sur les coefficients afin de régulariser les situations en grande dimension où en présence de corrélation.

Régression logistique pénalisée

Avec pénalité LASSO

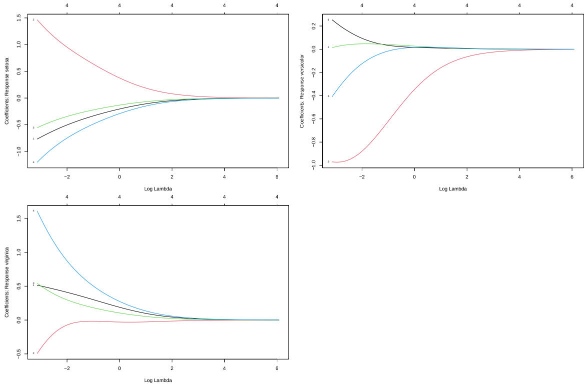
```
library(glmnet)
fit=glmnet(as.matrix(iris[,1:4]),iris[,5],alpha=1,
           family="multinomial")
plot(fit, xvar = "lambda", label = TRUE, type.coef = "coef")
```



Régression logistique pénalisée

Avec pénalité Ridge

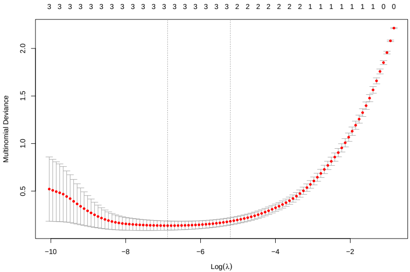
```
library(glmnet)
fit=glmnet(as.matrix(iris[,1:4]),iris[,5],alpha=0,
           family="multinomial")
plot(fit, xvar = "lambda", label = TRUE, type.coef = "coef")
```



Régression logistique pénalisée

Pour la pénalité LASSO, le paramètre de pénalisation peut-être choisi par CV

```
require(doMC)
registerDoMC(cores=2)
cvfit=cv.glmnet(as.matrix(iris[,1:4]),iris[,5],
                family="multinomial",parallel=TRUE)
plot(cvfit)
```



Régression logistique pénalisée

Une fonction predict permet d'effectuer des prédictions

```
p=predict(cvfit, newx = as.matrix(iris[,1:4]),
        s="lambda.min",type ="class")
table(p,iris[,5])
```

```
##
## p            setosa versicolor virginica
##   setosa         50          0         0
##   versicolor      0         48         1
##   virginica       0          2        49
```

L'option lambda.1se permet de choisir le lambda pénalisant le plus, tout en donnant un erreur CV pas significativement différente de celle minimale (obtenue avec lambda.min).

Bilan sur la régression logistique

Avantages :

- bonnes performances en prédiction
- sélection de variables à l'aide de test statistique
- prise en compte des prédicteurs quantitatives et catégorielles
- version pénalisées disponibles pour la grande dimension