Solution Exercice #2, Série 3

Francis Duval

11 février 2020

Pour les énoncés des exercices, cliquer sur ce lien: https://nbviewer.jupyter.org/github/nmeraihi/ACT6100/blob/master/exercices_3.ipynb

Activer les librairies utiles.

```
library(here)
library(tidyverse)
library(magrittr)
library(RcmdrMisc)
```

Lire la base de données `credit.csv`.

```
credit <- read_delim(here("0_data", "credit.csv"), delim = ";")
```

Pour voir rapidement à quoi ressemble la base de données, on peut utiliser la fonction `glimpse`.

```
glimpse(credit)
```

```
## Observations: 1,500
## Variables: 8
## $ Statut   <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Age      <dbl> 38, 45, 39, 20, 61, 45, 47, 37, 18, 26, 19, 46, 52, 2...
## $ Revenu   <dbl> 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134...
## $ Tendett  <chr> "19,1", "10,2", "9,2", "13,5", "10,4", "5", "15,4", "...
## $ Nexp     <dbl> 7, 9, 9, 0, 7, 19, 8, 1, 0, 4, 0, 2, 8, 4, 6, 31, 9, ...
## $ Rabanque <dbl> 5, 11, 5, 0, 22, 7, 13, 10, 0, 4, 0, 14, 6, 2, 11, 14...
## $ Prof     <dbl> 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,...
## $ Genre    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

Partie 1

La régression logistique est mieux adaptée que la régression linéaire car nous faisons face à un problème de classification (en fait, la régression logistique a un nom trompeur: elle devrait plutôt s'appeler la classification logisitque). En effet, la variable que nous essayons de prédire est la variable `Statut`, qui est catégorielle puisqu'elle ne peut prendre que les valeurs 0 ou 1. Le but est d'estimer pour chaque observation la probabilité que la variable `Statut` prenne la valeur 1. Or, si on utilisait la régression linéaire, on risquerait de se retrouver avec des probabilités inférieures à 0 ou supérieures à 1, ce qui n'a pas de sens.

Partie 2

Premièrement, il faut arranger un peu la base de données. On remarque que la variable `Tendett` est une variable de type « chaine de caractères », mais devrait plutôt être une variable numérique. Il faut donc la convertir en variable numérique. Deuxièmement, les variables `Prof` et `Genre` sont numériques, mais devraient plutôt être catégorielle. On va changer ça aussi.

```
credit %<>%
  mutate(
    Tendett = as.numeric(str_replace(Tendett, ",", ".")),
    Prof = factor(Prof),
    Genre = factor(Genre)
  )
```

Maintenant, tout est comme on veut:

```
glimpse(credit)
```

```
## Observations: 1,500
## Variables: 8
## $ Statut   <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Age      <dbl> 38, 45, 39, 20, 61, 45, 47, 37, 18, 26, 19, 46, 52, 2...
## $ Revenu   <dbl> 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134...
## $ Tendett  <dbl> 19.1, 10.2, 9.2, 13.5, 10.4, 5.0, 15.4, 1.8, 9.9, 0.8...
## $ Nexp     <dbl> 7, 9, 9, 0, 7, 19, 8, 1, 0, 4, 0, 2, 8, 4, 6, 31, 9, ...
## $ Rabanque <dbl> 5, 11, 5, 0, 22, 7, 13, 10, 0, 4, 0, 14, 6, 2, 11, 14...
## $ Prof     <fct> 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,...
## $ Genre    <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

On peut maintenant ajuster la régression logistique avec toutes les variables. Le modèle complet est celui qui estime la probabilité que la variable `Statut` prenne la valeur 1 à l'aide de toutes les autres variables.

```
glm_logit_fit <- glm(Statut ~ ., family = binomial(link = "logit"), data = credit)
summary(glm_logit_fit)
```

```
##
## Call:
## glm(formula = Statut ~ ., family = binomial(link = "logit"),
##     data = credit)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.8059  -0.6016   0.2407   0.6431   2.9395
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.016887   0.412009  -0.041  0.96731
## Age          0.033536   0.016484   2.034  0.04191 *
## Revenu       0.004046   0.001971   2.053  0.04011 *
## Tendett     -0.162879   0.012240 -13.307  < 2e-16 ***
## Nexp         0.097315   0.017309   5.622 1.89e-08 ***
## Rabanque    -0.024105   0.035313  -0.683  0.49485
## Prof2       -0.758109   0.240756  -3.149  0.00164 **
## Prof3        0.748335   0.245004   3.054  0.00226 **
## Prof4       -0.452249   0.262196  -1.725  0.08455 .
## Genre1       2.095176   0.168239  12.454  < 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: 1969.3  on 1499  degrees of freedom
## Residual deviance: 1225.6  on 1490  degrees of freedom
## AIC: 1245.6
##
## Number of Fisher Scoring iterations: 6
```

On utilise une méthode de sélection de variables de type « backward » avec le BIC comme critère.

```
summary(stepwise(glm_logit_fit, direction = "backward", criterion = "BIC", trace = F))
```

```
##
## Direction:  backward
## Criterion:  BIC
```

```
##
## Call:
## glm(formula = Statut ~ Age + Tendett + Nexp + Prof + Genre, family = binomial(link = "logit"),
##     data = credit)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.4780  -0.6094   0.2380   0.6541   2.9310
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.203129   0.317113   0.641  0.52181
## Age          0.025956   0.008543   3.038  0.00238 **
## Tendett     -0.162385   0.012193 -13.318  < 2e-16 ***
## Nexp         0.106166   0.016819   6.312 2.75e-10 ***
## Prof2       -0.703060   0.238934  -2.942  0.00326 **
## Prof3        0.861149   0.239323   3.598  0.00032 ***
## Prof4       -0.280987   0.248728  -1.130  0.25860
## Genre1       2.051620   0.166810  12.299  < 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: 1969.3  on 1499  degrees of freedom
## Residual deviance: 1230.1  on 1492  degrees of freedom
## AIC: 1246.1
##
## Number of Fisher Scoring iterations: 6
```

Le modèle final est donc celui qui estime la probabilité que la variable `Statut` prenne la valeur 1 à l'aide des variables `Age`, `Revenu`, `Tendett`, `Nexp` et `Genre`.

Partie 3

Bien que la valeur p associée a `Prof4` soit supérieure à 0.05, on ne peut la retirer du modèle. En effet, il s'agit d'une modalité de la variable `Prof` et on ne peut retirer une seule modalité: soit on retire toutes les modalités, soit on garde toutes les modalités. On pourrait éventuellement tenter de la regrouper avec une autre modalité.

Partie 4

Bien que la valeur p associée à l'ordonnée à l'origine soit supérieure à 0.05, on ne peut la retirer du modèle. En effet, l'élément Intercept du modèle comprend non seulement l'ordonnée à l'origine (le « vrai » beta_0), mais également l'effet de référence pour chacune des variables catégorielles (homme pour la variable `Genre` et libérale pour la variable `Prof`). Ceci est dû au fait que l'encodage est réalisé à l'aide de 0/1 plutôt que de -1/1.

Partie 5

À venir

Partie 6

À venir