Scoring de crédit — Régression logistique

Axel LOUKOU _ Master 2 RAD

2024-10-26

```
options(repos = c(CRAN = "https://cloud.r-project.org"))
```

Packages & données

```
# install.packages(c("dplyr","ggplot2","MASS","margins"))
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(MASS)     # modèles logistiques
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(margins)  # effets marginaux
```

```
# Le fichier doit être à côté de ce Rmd (ou ajuster le chemin relatif)
credit <- read.csv("german_creditDV.csv", stringsAsFactors = FALSE)

# petit aperçu
glimpse(credit)
```

```
## Rows: 1,000
## Columns: 21
## $ status                  <int> 1, 1, 2, 1, 1, 1, 1, 1, 4, 2, 1, 1, 1, 2, 1, 1~
## $ duration                <int> 18, 9, 12, 12, 12, 10, 8, 6, 18, 24, 11, 30, 6~
## $ credit_history          <int> 4, 4, 2, 4, 4, 4, 4, 4, 4, 2, 4, 4, 4, 3, 2, 2~
## $ purpose                 <int> 2, 0, 9, 0, 0, 0, 0, 0, 3, 3, 0, 1, 3, 10, 3, ~
## $ amount                  <int> 1049, 2799, 841, 2122, 2171, 2241, 3398, 1361,~
## $ savings                 <int> 1, 1, 2, 1, 1, 1, 1, 1, 1, 3, 1, 2, 1, 2, 5, 3~
## $ employment_duration     <int> 2, 3, 4, 3, 3, 2, 4, 2, 1, 1, 3, 4, 4, 1, 4, 3~
## $ installment_rate        <int> 4, 2, 2, 3, 4, 1, 1, 2, 4, 1, 2, 1, 1, 2, 2, 2~
## $ personal_status_sex     <int> 2, 3, 2, 3, 3, 3, 3, 3, 2, 2, 3, 4, 2, 3, 4, 3~
## $ other_debtors           <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ present_residence       <int> 4, 2, 4, 2, 4, 3, 4, 4, 4, 4, 2, 4, 4, 4, 4, 3~
## $ property                <int> 2, 1, 1, 1, 2, 1, 1, 1, 3, 4, 1, 3, 3, 4, 3, 1~
## $ age                     <int> 21, 36, 23, 39, 38, 48, 39, 40, 65, 23, 36, 24~
## $ other_installment_plans <int> 3, 3, 3, 3, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3~
## $ housing                 <int> 1, 1, 1, 1, 2, 1, 2, 2, 2, 1, 1, 1, 2, 2, 1, 1~
## $ number_credits          <int> 1, 2, 1, 2, 2, 2, 2, 1, 2, 1, 2, 2, 1, 1, 2, 1~
## $ job                     <int> 3, 3, 2, 2, 2, 2, 2, 2, 1, 1, 3, 3, 3, 4, 2, 3~
## $ people_liable           <int> 2, 1, 2, 1, 2, 1, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1~
## $ telephone               <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1~
## $ foreign_worker          <int> 2, 2, 2, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2~
## $ credit_risk             <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
```

```
summary(credit$credit_risk)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     0.0     0.0     1.0     0.7     1.0     1.0 
```

Préparation des variables (facteurs & dummies)

```
# Vérifier les valeurs manquantes sur les variables catégorielles utilisées
colSums(is.na(credit[, c("other_debtors","property")]))
```

```
## other_debtors      property 
##             0             0 
```

```
# Convertir en facteurs avec libellés clairs (sans accents/espaces pour des noms propres)
credit$other_debtors <- factor(credit$other_debtors,
                               levels = c(1,2,3),
                               labels = c("Aucun","Co_emprunteur","Garant"))

credit$property <- factor(credit$property,
                          levels = c(1,2,3,4),
                          labels = c("Pas_de_propriete","Voiture","Assurance_vie","Immobilier"))

# Création de dummies (si on veut des colonnes explicites)
dummies_other <- model.matrix(~ other_debtors - 1, data = credit) %>% as.data.frame()
dummies_prop  <- model.matrix(~ property - 1,      data = credit) %>% as.data.frame()
```

```
credit <- bind_cols(credit, dummies_other, dummies_prop)

# Vérification
head(credit[, c("other_debtors","property", colnames(dummies_other), colnames(dummies_prop))])
```

```
##   other_debtors         property other_debtorsAucun other_debtorsCo_emprunteur
## 1         Aucun          Voiture                  1                          0
## 2         Aucun Pas_de_propriete                  1                          0
## 3         Aucun Pas_de_propriete                  1                          0
## 4         Aucun Pas_de_propriete                  1                          0
## 5         Aucun          Voiture                  1                          0
## 6         Aucun Pas_de_propriete                  1                          0
##   other_debtorsGarant propertyPas_de_propriete propertyVoiture
## 1                   0                        0               1
## 2                   0                        1               0
## 3                   0                        1               0
## 4                   0                        1               0
## 5                   0                        0               1
## 6                   0                        1               0
##   propertyAssurance_vie propertyImmobilier
## 1                     0                  0
## 2                     0                  0
## 3                     0                  0
## 4                     0                  0
## 5                     0                  0
## 6                     0                  0
```

Partition train / test

```
n <- nrow(credit)
idx_train <- sample(seq_len(n), size = floor(0.7 * n))
train_data <- credit[idx_train, ]
test_data  <- credit[-idx_train, ]
```

Modèle logit (train) & évaluation

```
# Formule avec dummies explicites
form <- as.formula(
  paste(
    "credit_risk ~ amount + employment_duration + installment_rate +",
    "savings + number_credits +",
    # dummies other_debtors
    paste(colnames(dummies_other), collapse = " + "), "+",
    # dummies property
    paste(colnames(dummies_prop),  collapse = " + ")
  )
)
```

```r
model_logit_train <- glm(form, data = train_data, family = binomial())
summary(model_logit_train)
```

```
##
## Call:
## glm(formula = form, family = binomial(), data = train_data)
##
## Coefficients: (2 not defined because of singularities)
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 8.245e-01  6.631e-01   1.244 0.213670
## amount                     -1.616e-04  3.686e-05  -4.385 1.16e-05 ***
## employment_duration         2.512e-01  7.404e-02   3.392 0.000693 ***
## installment_rate           -3.523e-01  8.832e-02  -3.989 6.63e-05 ***
## savings                     3.700e-01  6.620e-02   5.588 2.29e-08 ***
## number_credits              2.847e-01  1.632e-01   1.744 0.081103 .
## other_debtorsAucun         -5.239e-01  4.246e-01  -1.234 0.217232
## other_debtorsCo_emprunteur -6.305e-01  5.752e-01  -1.096 0.273020
## other_debtorsGarant                NA         NA      NA       NA
## propertyPas_de_propriete    4.334e-01  3.059e-01   1.417 0.156601
## propertyVoiture             6.960e-02  3.007e-01   0.231 0.816971
## propertyAssurance_vie       1.242e-01  2.784e-01   0.446 0.655427
## propertyImmobilier                 NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 858.57  on 699  degrees of freedom
## Residual deviance: 766.42  on 689  degrees of freedom
## AIC: 788.42
##
## Number of Fisher Scoring iterations: 4
```

```r
# Prédictions proba
p_train <- predict(model_logit_train, newdata = train_data, type = "response")
p_test  <- predict(model_logit_train, newdata = test_data,  type = "response")

# Seuil
threshold <- 0.5
yhat_train <- ifelse(p_train > threshold, 1, 0)
yhat_test  <- ifelse(p_test  > threshold, 1, 0)

# Accuracy & matrices de confusion
acc_train <- mean(yhat_train == train_data$credit_risk)
acc_test  <- mean(yhat_test  == test_data$credit_risk)

acc_train; acc_test
```

```
## [1] 0.7214286
```

```
## [1] 0.7166667
```

```r
tab_train <- table(Predicted = yhat_train, Actual = train_data$credit_risk)
tab_test  <- table(Predicted = yhat_test,  Actual = test_data$credit_risk)

tab_train; tab_test
```

```
##          Actual
## Predicted   0   1
##         0  54  37
##         1 158 451
```

```
##          Actual
## Predicted   0   1
##         0  23  20
##         1  65 192
```

Modèle logit sur l’ensemble & effets marginaux

```r
model_logit <- glm(form, data = credit, family = binomial())
summary(model_logit)
```

```
## 
## Call:
## glm(formula = form, family = binomial(), data = credit)
## 
## Coefficients: (2 not defined because of singularities)
##                              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)                 6.833e-01  5.541e-01   1.233 0.217462    
## amount                     -1.324e-04  2.815e-05  -4.701 2.59e-06 ***
## employment_duration         2.140e-01  6.197e-02   3.454 0.000553 ***
## installment_rate           -2.841e-01  7.212e-02  -3.940 8.15e-05 ***
## savings                     3.110e-01  5.327e-02   5.839 5.26e-09 ***
## number_credits              1.875e-01  1.318e-01   1.422 0.154940    
## other_debtorsAucun         -5.496e-01  3.743e-01  -1.468 0.142092    
## other_debtorsCo_emprunteur -8.820e-01  5.011e-01  -1.760 0.078429 .  
## other_debtorsGarant                NA         NA      NA       NA    
## propertyPas_de_propriete    7.554e-01  2.437e-01   3.100 0.001937 ** 
## propertyVoiture             3.394e-01  2.359e-01   1.439 0.150267    
## propertyAssurance_vie       4.509e-01  2.168e-01   2.080 0.037522 *  
## propertyImmobilier                 NA         NA      NA       NA    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1221.7  on 999  degrees of freedom
## Residual deviance: 1113.2  on 989  degrees of freedom
## AIC: 1135.2
## 
## Number of Fisher Scoring iterations: 4
```

```r
# Effets marginaux (margins sur glm binomial)
me <- margins(model_logit)
summary(me)
```

```
##                       factor     AME SE  z  p lower upper
##                       amount -0.0000 NA NA NA    NA    NA
##          employment_duration  0.0402 NA NA NA    NA    NA
##             installment_rate -0.0533 NA NA NA    NA    NA
##               number_credits  0.0352 NA NA NA    NA    NA
##           other_debtorsAucun -0.1031 NA NA NA    NA    NA
##   other_debtorsCo_emprunteur -0.1655 NA NA NA    NA    NA
##          other_debtorsGarant  0.0000 NA NA NA    NA    NA
##        propertyAssurance_vie  0.0846 NA NA NA    NA    NA
##           propertyImmobilier  0.0000 NA NA NA    NA    NA
##     propertyPas_de_propriete  0.1417 NA NA NA    NA    NA
##              propertyVoiture  0.0637 NA NA NA    NA    NA
##                      savings  0.0584 NA NA NA    NA    NA
```

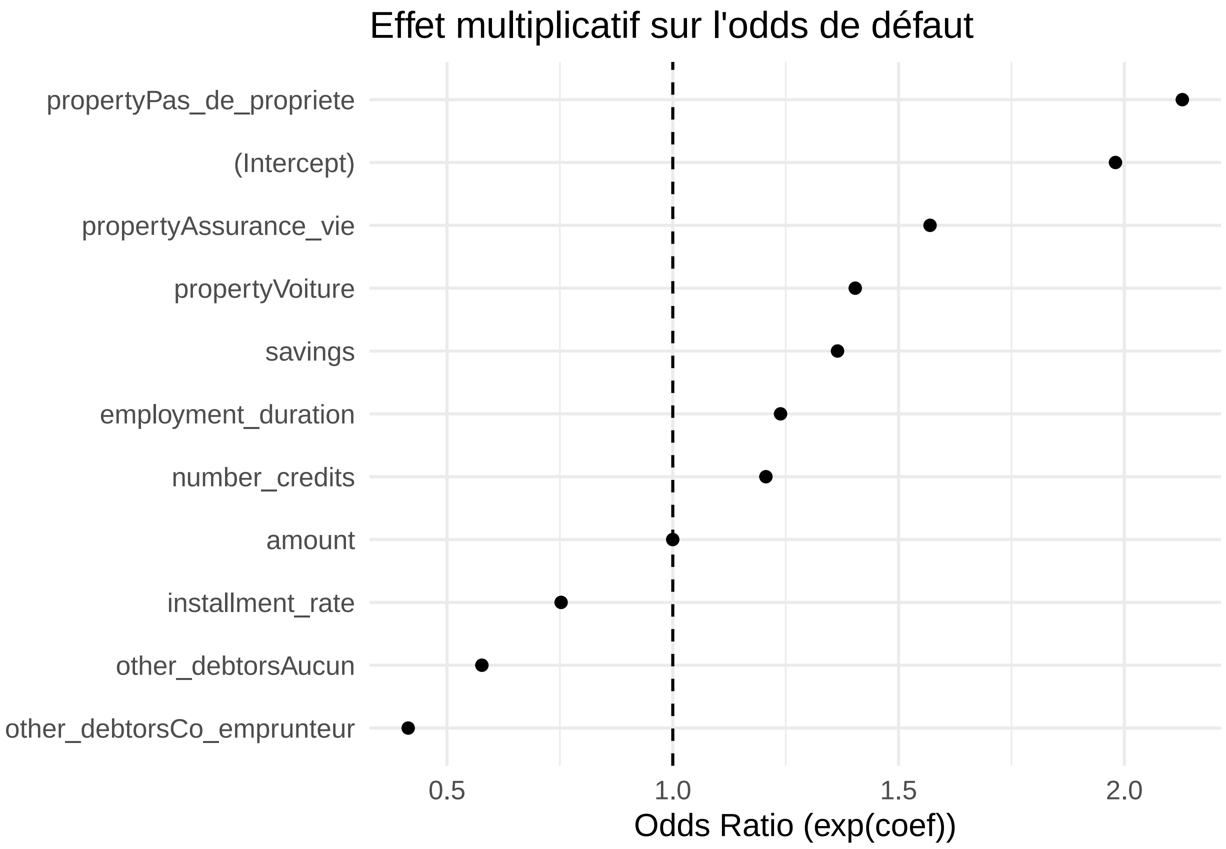
Odds ratios & interprétation rapide

```r
or <- exp(coef(model_logit))
OR <- data.frame(
  variable = names(or),
  odds_ratio = unname(or)
) %>%
  arrange(desc(abs(odds_ratio - 1)))

head(OR, 12)
```

```
##                       variable odds_ratio
## 1    propertyPas_de_propriete  2.1284699
## 2                 (Intercept)  1.9804316
## 3  other_debtorsCo_emprunteur  0.4139728
## 4       propertyAssurance_vie  1.5697636
## 5          other_debtorsAucun  0.5772045
## 6             propertyVoiture  1.4040745
## 7                     savings  1.3648393
## 8            installment_rate  0.7526579
## 9         employment_duration  1.2386266
## 10             number_credits  1.2062362
## 11                     amount  0.9998677
## 12        other_debtorsGarant         NA
```

```r
OR %>%
  filter(!is.na(odds_ratio)) %>%
  mutate(variable = reorder(variable, odds_ratio)) %>%
  ggplot(aes(x = variable, y = odds_ratio)) +
  geom_point() +
  geom_hline(yintercept = 1, linetype = "dashed") +
  coord_flip() +
  labs(x = NULL, y = "Odds Ratio (exp(coef))",
       title = "Effet multiplicatif sur l’odds de défaut") +
  theme_minimal(base_size = 12)
```


Effet multiplicatif sur l'odds de défaut
propertyPas_de_propriete
(Intercept)
propertyAssurance_vie
propertyVoiture
savings
employment_duration
number_credits
amount
installment_rate
other_debtorsAucun
other_debtorsCo_emprunteur
Odds Ratio (exp(coef))