Université de Reims Champagne Ardenne –Reims-
UFR Sciences Exactes et Naturelles
Département d'Informatique

INFO 0901 : Apprentissage statistique et Data mining

Rapport de projet INFO 0901

Filière : Informatique

Option : Intelligence Artificielle.

Thème :

Analyse du Risque d'Emprunt Bancaire via Classification Binaire et Scoring

Présenté par :
BEN ALI Dhia

Encadré par :
Mr KEIZOU Amor

2023

Table des matières

Table des figures

1 Introduction

L'analyse du risque de crédit constitue un aspect essentiel de la gestion financière pour les institutions bancaires. Dans le cadre de ce projet, nous explorons cette problématique en utilisant des données provenant du site Kaggle, une plateforme renommée pour le partage de jeux de données et de projets liés à l'apprentissage automatique. Le jeu de données, accessible à l'adresse suivante : lien Kaggle, intitulé "Bankloans.csv", offre un aperçu détaillé des détails de crédit des emprunteurs.

1.1 Origine des Données

Les données ont été collectées et mises à disposition sur Kaggle par un contributeur, offrant ainsi une opportunité d'explorer et d'analyser les tendances dans le domaine de l'analyse du risque de crédit. Kaggle sert de plateforme collaborative où la communauté mondiale des data scientists partage des connaissances et collabore sur des projets variés.

1.2 Description des Variables

1. **Age (âge du client) :** La variable "age" représente l'âge du client, offrant des indications sur la maturité financière et la stabilité.
2. **Ed (niveau d'éducation du client) :** La variable "ed" indique le niveau d'éducation du client, un facteur potentiellement lié à la stabilité financière.
3. **Employ (ancienneté avec l'employeur actuel) :** "Employ" représente la durée, en années, pendant laquelle le client est resté avec son employeur actuel, fournissant des informations sur la stabilité professionnelle.
4. **Address (nombre d'années à la même adresse) :** Cette variable, "address", renseigne sur la stabilité de résidence du client.
5. **Income (revenu du client) :** "Income" représente le revenu du client, un élément crucial pour évaluer sa capacité à rembourser un prêt.
6. **Debtinc (ratio dettes/revenu) :** La variable "debtinc" quantifie le rapport entre les dettes et le revenu du client, un indicateur significatif du niveau d'endettement.
7. **Creddebt (ratio crédit/dette) :** "Creddebt" exprime le rapport entre le crédit utilisé et la dette totale, influençant la capacité de remboursement.
8. **Othdebt (autres dettes) :** Cette variable, "othdebt", représente les autres dettes du client en dehors du crédit, contribuant à évaluer l'ensemble des obligations financières.
9. **Default (client a fait défaut dans le passé) :** La variable binaire "default" indique si le client a fait défaut dans le passé (1 pour défaut, 0 pour aucun défaut).

En explorant et en analysant ces données, nous chercherons à dégager des tendances significatives qui contribueront à une prise de décision éclairée dans le domaine du crédit.

2 Modèle de Régression Logistique Binaire

2.1 Notations et contexte

Dans le cadre de la régression logistique binaire, nous travaillons avec un ensemble d'échantillons noté Ω, comprenant n observations. Chaque observation est caractérisée par une variable à prédire Y et un ensemble de variables prédictives $X = (X_1, X_2, \ldots, X_p)$ supposées indépendantes et identiquement distribuées (iid).

La variable Y prend deux modalités, $\{1, 0\}$, ce qui correspond aux deux catégories possibles dans le problème binaire.

Pour évaluer la probabilité conditionelle d'obtenir la modalité 1 (resp. 0) de Y sachant les valeurs observées de X, nous utilisons les notations $P(Y = 1|X = x)$ et $P(Y = 0|X = x)$.

2.2 Rappels et Définitions

Définition 1 (Fonction sigmoïde)

En mathématiques, la fonction sigmoïde (dite aussi courbe en S) est définie par :

$$f(x) = \frac{1}{1 + e^{-x}}$$

pour tout réel x.Elle représente la fonction de répartition de la loi logistique.

Définition 2 (Régression Logistique Binaire)

La régression logistique binaire est un modèle statistique utilisé pour modéliser la probabilité d'un événement ayant deux issues possibles. La fonction logistique est généralement utilisée pour transformer une combinaison linéaire des variables indépendantes en une probabilité comprise entre 0 et 1.

La forme générale d'un modèle de régression logistique binaire pour p variables explicatives est donnée par l'équation suivante :

$$P(Y = 1|X = x) = \frac{1}{1 + e^{-(\omega_0 + \omega_1 x_1 + \omega_2 x_2 + \ldots + \omega_p x_p)}} + \epsilon$$

où :

— ω_0 est l'intercept.

— $\omega_1, \omega_2, \ldots, \omega_p$ sont les coefficients associés aux variables indépendantes $X_1, X_2, \ldots, X_p$, respectivement.

La fonction logistique $\frac{1}{1+e^{-z}}$ transforme la combinaison linéaire $z = \omega_0 + \omega_1 x_1 + \omega_2 x_2 + \ldots + \omega_p x_p$ en une probabilité.

Définition 3 (Loi de Bernoulli)

Dans le contexte de la Régression Logistique Binaire, la variable à prédire suit une loi de Bernoulli paramétrée par la fonction de probabilité conditionnelle $P(Y = 1|X = x) = f(x, w) + \epsilon$, où w représente les paramètres du modèle.

Définition 4 (Log-Vraisemblance)

La log-vraisemblance $\ell_n(w)$ est le logarithme naturel de la vraisemblance conditionnelle, exprimé comme $\ell_n(w) = \sum_{i=1}^{n} y_i \log f(x_i, w) + (1 - y_i) \log(1 - f(x_i, w))$.

Définition 5 (Probabilités ou Valeurs Prédites)

Les probabilités prédites de $p_1(x)$ et $p_0(x)$ pour une observation $X = x$ sont définies comme $\hat{p}_1(x) = f(x, \hat{w})$ et $\hat{p}_0(x) = 1 - f(x, \hat{w})$.

Définition 6 (Modèle Logit "Linéaire")

En incorporant les probabilités prédites dans le modèle logit, on obtient une forme "linéaire" z_i définie par $z_i := \log \frac{p_1(x_i)}{1-p_1(x_i)} = w_0 + w_1 x_{i,1} + \ldots + w_p x_{i,p} + \varepsilon_i$.

2.3 Hypothèse Fondamentale

L'hypothèse fondamentale de la régression logistique repose sur l'idée que la relation entre les variables indépendantes (prédictives) et la variable dépendante (à prédire) est logistique. Autrement dit, la logit (le logarithme des cotes) de la variable dépendante est une combinaison linéaire des coefficients des variables indépendantes. Mathématiquement, cela s'exprime comme suit pour une observation

$$\ln\left(\frac{p(Y = 1|X = x)}{p(Y = 0|X = x)}\right) = w_0 + w_1 x_1 + \ldots + w_p x_p + \epsilon$$

où :
— $x_1, x_2, ..., x_p$ représentent les valeurs prises respectivement par les variables $X_1, X_2, ..., X_p$.
— ω_0 est l'intercept.
— $\omega_1, \omega_2, \ldots, \omega_p$ sont les coefficients associés aux variables indépendantes $X_1, X_2, \ldots, X_p$, respectivement.

2.4 Modèle LOGIT

La spécification précédente peut être reformulée en introduisant le terme LOGIT de $p(1|X)$ avec l'équation suivante :

$$\ln\left(\frac{p(Y = 1|X = x)}{1 - p(Y = 1|X = x)}\right) = w_0 + w_1 x_1 + \ldots + w_p x_p + \epsilon$$

Il s'agit d'une "régression" car elle explore une relation de dépendance entre une variable à expliquer et des variables explicatives.
Cette "régression" est "logistique" car elle modélise la loi de probabilité à partir d'une loi logistique.
En transformant cette équation, on obtient :

$$p(Y=1|X=x) = \frac{e^{w_0+w_1x_1+\ldots+w_px_p}}{1+e^{w_0+w_1x_1+\ldots+w_px_p}} + \epsilon$$

2.5 Estimation des paramètres

Soit (x_i, y_i), $i = 1, \ldots, n$, un échantillon représentatif de (X, Y),
et soit $z_i := \log \frac{p_1(x_i)}{1-p_1(x_i)} = w_0 + w_1 x_{i,1} + \ldots + w_p x_{i,p} + \varepsilon_i$..

On veut estimer les paramètres w_j de z_i qui modélise le logit $\log \frac{p_1(x_i)}{1-p_1(x_i)}$. Alors dans le cas de la régression logistique, les paramètres ω_j sont estimés par la méthode du maximum de vraisemblance.

L'estimateur du maximum de vraisemblance $\hat{w}$ est obtenu en maximisant la log-vraisemblance, c'est-à-dire $\hat{w} = \arg\sup_w \ell_n(w)$.

avec $\ell_n(w) = \sum_{i=1}^n y_i \log f(x_i, w) + (1 - y_i) \log(1 - f(x_i, w))$.

2.6 Sélection de modèles en Régression Logistique

On considère une liste de p variables $X_1, \ldots, X_p$ susceptibles d'expliquer une variable binaire Y via un modèle de régression logistique. L'objectif est de choisir le meilleur sous-ensemble de variables conduisant au modèle optimal de classification de Y.

2.6.1 Critères AIC et BIC

On a le choix entre plusieurs modèles, notamment ceux utilisant une seule variable, deux variables, etc. Au total, il y a $m = 2^p - 1$ modèles possibles, notés $M_1, \ldots, M_m$.

Critère AIC : L'**AIC** d'un modèle M_k, avec $k = 1, \ldots, m$, est défini par

$$AIC(M_k) = -2 \sup_{w \in \mathbb{R}^{1+n_k}} \ell_n(w) + 2(1 + n_k),$$

où n_k est le nombre de variables explicatives de M_k. La sélection basée sur l'AIC consiste à choisir le modèle M_{k^*}

avec $k^* = \arg\min_{k \in \{1,\ldots,m\}} AIC(M_k)$.

Critère BIC : Le **BIC** d'un modèle M_k est défini par

$$BIC(M_k) = -2 \sup_{w \in \mathbb{R}^{1+n_k}} \ell_n(w) + \log(n)(1 + n_k),$$

et la sélection basée sur le BIC consiste à choisir le modèle M_{k^*}

avec $k^* = \arg\min_{k \in \{1,\ldots,m\}} BIC(M_k)$.

2.6.2 Les méthodes de choix des modèles par les critères AIC ou BIC

La sélection de variables est une étape cruciale dans le la regression logistique, afin de trouver le meilleur modèle . on trouve la recherche exhaustive, l'algorithme génétique, la méthode ascendante, la méthode descendante, et la méthode bidirectionnelle. Chacune de ces méthodes a ses propres avantages et inconvénients, adaptés à des problèmes spécifiques.

Recherche exhaustive : L'algorithme de recherche exhaustive évalue toutes les combinaisons possibles de variables, ce qui peut être coûteux en termes de calculs. Pour un ensemble de p variables, cela implique la comparaison de $2^p - 1$ modèles.

Algorithme génétique : L'utilisation de l'algorithme génétique est suggérée pour les cas où le nombre de variables (p) est élevé.

Méthode ascendante : La méthode ascendante commence avec un modèle trivial (aucune variable) et ajoute séquentiellement les variables qui améliorent le modèle.

Méthode descendante : La méthode descendante commence avec un modèle global et élimine séquentiellement les variables qui ont le moins d'impact sur le modèle.

Méthode bidirectionnelle : La méthode bidirectionnelle combine les approches ascendante et descendante. Elle ajoute ou élimine séquentiellement les variables.

Remarques
- Les modèles optimaux obtenus par AIC et BIC peuvent différer.
- Les critères AIC et BIC ne garantissent pas le modèle d'erreur théorique minimale.
- Une alternative consiste à utiliser la méthode K-fold CV pour estimer l'erreur théorique de chaque modèle.

2.7 Test de significativité des estimateurs

2.7.1 Test de Wald

Il s'agit de tester $H_0 : w_1 = 0$ contre $H_1 : w_1 \neq 0$.

La statistique de test pour le test de Wald dans le contexte de la régression logistique est définie comme suit :

$$Z_n = \sqrt{n} \left(\frac{\hat{w}_{b1}}{\sigma_{b1}} \right)$$

Cette statistique suit approximativement une distribution normale ($N(0, 1)$) sous l'hypothèse nulle $H_0 : w_1 = 0$.

On peut utiliser cette statistique pour calculer la P-valeur du test de Wald en comparant son module à la distribution normale standard.

Soit Z une variable aléatoire suivant la loi $N(0, 1)$..

La P-valeur du test précédent est alors donnée par

$$\text{P-valeur} = P(|Z| > |Z_n|).$$

Si P-valeur $< \alpha$, on rejette H_0 et Si P-valeur $\geq \alpha$, on accepte H_0.

où α est le risque de premier espèce.

2.7.2 Test de rapport de vraisemblance

Il s'agit de tester $H_0 : w_1 = 0$ contre $H_1 : w_1 \neq 0$.

Notons $W := \mathbb{R}^{1+p}$ et $W_0 := \mathbb{R} \times \{0\} \times \mathbb{R}^{p-1}$. On peut alors écrire les deux hypothèses précédentes sous la forme équivalente suivante :

$$H_0 : w \in W_0 \quad \text{contre} \quad H_1 : w \in W/W_0.$$

Le rapport de vraisemblances associé aux hypothèses précédentes s'écrit alors

$$R_n := \frac{\sup_{w \in W} L_n(w)}{\sup_{w \in W_0} L_n(w)}.$$

La statistique du rapport de vraisemblance correspondante s'écrit

$$S_n := 2 \log R_n = -2 \sup_{w \in W_0} \lambda_n(w) - \left(-2 \sup_{w \in W} \lambda_n(w) \right) =: D_{\text{Modèle Réduit}} - D_{\text{Modèle global}}.$$

La statistique S_n , si H_0 est vraie, converge en loi, quand $n \to \infty$, vers une loi du $\chi^2(d)$ à d degrés de liberté avec $d = \dim(W) - \dim(W_0) = (1 + p) - (p) = 1$.

Soit Z une variable aléatoire suivant la loi $\chi^2(p)$.

La P-valeur du test précédent est alors donnée par

$$\text{P-valeur} = P(Z > S_n).$$

Si P-valeur $< \alpha$, on rejette H_0 et Si P-valeur $\geq \alpha$, on accepte H_0.

où α est le risque de premier espèce.

2.7.3 Test de validité du modèle global

Il s'agit de tester $H_0 : \omega_1 = \omega = \ldots = \omega_p = 0$ contre $H_1 : \exists i$ tel que $\omega_i \neq 0$.

On peut utiliser ici le test de rapport de vraisemblance.

Notons $W := \mathbb{R}^{1+p}$ et $W_0 := \mathbb{R} \times \{0\} \times \ldots \times \{0\}$.

La statistique du rapport de vraisemblance correspondante s'écrit

$$S_n = -2 \sup_{w \in W_0} \ell_n(w) - \left(-2 \sup_{w \in W} \ell_n(w) \right) = D_{\text{ModelTrivial}} - D_{\text{ModelComplet}},$$

La statistique S_n , si H_0 est vraie, converge en loi, quand $n \to \infty$, vers une loi du $\chi^2(d)$ à d degrés de liberté avec $d = \dim(W) - \dim(W_0) = (1 + p) - (1) = p$.

Soit Z une variable aléatoire suivant la loi $\chi^2(p)$.

La P-valeur du test précédent est alors donnée par

$$\text{P-valeur} = P(Z > S_n).$$

Si P-valeur $< \alpha$, on rejette H_0 et Si P-valeur $\geq \alpha$, on accepte H_0.

où α est le risque de premier espèce.

3 Scoring

Étant donnés un échantillon $(X_1, Y_1), \ldots, (X_n, Y_n)$, le travail consiste à construire une fonction $S(x)$ qui permette d'expliquer Y .

3.1 Définitions

Définition 1 : Scoring

Un score est une fonction $S : x \in \mathbb{R}^p \mapsto S(x) \in \mathbb{R}$;

La fonction de score théorique est définie par :

$$S(x) = P(Y = 1 | X = x).$$

La construction de scores s'effectue généralement avec les modèles de classification (binaire) classiques.Une fois le score construit, la décision s'effectue selon la règle

$$\hat{Y} = \begin{cases} 1 & \text{si } \hat{S}(x) > s, \\ 0 & \text{sinon,} \end{cases}$$

où s est un seuil choisi par l'utilisateur.

Définition 2 : Courbe ROC (Receiver Operating Characteristic)

La courbe ROC permet de représenter sur un graphe 2D les deux types d'erreur pour tous les seuils s. C'est une courbe 2D paramétrée par le seuil s, avec les équations suivantes :

$$x(s) = \alpha(s) = 1 - \text{sp}(s) = P(S(X) > s | Y = 0)$$
$$y(s) = 1 - \beta(s) = \text{se}(s) = P(S(X) \geq s | Y = 1).$$

.

Définition 3 : AUC (Area Under the Curve)

C'est l'aire sous la courbe ROC d'un score $S(\cdot)$, notée AUC(S), est souvent utilisée pour mesurer sa performance.

3.2 Évaluation des modèles

De nombreux modèles d'analyse discriminante permettent d'estimer $P(Y = 1|X = x)$. L'estimation de S sera généralement basée sur les méthodes d'analyse discriminante (modèle logistique, LDA, QDA, K-NN, SVM, Arbres de décision, Forêts aléatoires, Adaboost, ...).

L'évaluation de ces modèles de scoring est une étape cruciale pour mesurer leur performance . Plusieurs métriques d'évaluation sont utilisées. Parmi ces métriques, on peut utiliser la courbe **ROC (Receiver Operating Characteristic)** et **l'AUC (Area Under the Curve)** qui sont des outils simples et efficaces pour évaluer la performance des modèles de scoring

4 Application

4.1 Prétraitement des Données

Le prétraitement des données est une étape cruciale dans le processus de préparation des données avant leur utilisation dans un modèle d'apprentissage automatique. Cette phase vise à améliorer la qualité, la cohérence et la pertinence des données, contribuant ainsi à l'efficacité et à la performance du modèle. Les étapes typiques du prétraitement incluent la gestion des valeurs manquantes, l'élimination des doublons, la normalisation des données numériques et la conversion des variables catégorielles en format approprié.

```
> str(bankloans)
'data.frame':   700 obs. of  9 variables:
 $ age     : int  41 27 40 41 24 41 39 43 24 36 ...
 $ ed      : Factor w/ 5 levels "1","2","3","4",..: 3 1 1 1 2 2 1 1 1 1 ...
 $ employ  : int  17 10 15 15 2 5 20 12 3 0 ...
 $ address : int  12 6 14 14 0 5 9 11 4 13 ...
 $ income  : int  176 31 55 120 28 25 67 38 19 25 ...
 $ debtinc : num  9.3 17.3 5.5 2.9 17.3 10.2 30.6 3.6 24.4 19.7 ...
 $ creddebt: num  11.359 1.362 0.856 2.659 1.787 ...
 $ othdebt : num  5.009 4.001 2.169 0.821 3.057 ...
 $ default : int  1 0 0 0 1 0 0 0 1 0 ...
 - attr(*, "na.action")= 'omit' Named int [1:150] 701 702 703 704 705 706 707 708 709 710 ...
  ..- attr(*, "names")= chr [1:150] "701" "702" "703" "704" ...
>
```

FIGURE 1 – Description des données après traitement.

Dans le cadre de la préparation des données pour la régression logistique, un prétraitement minutieux a été effectué afin d'optimiser la qualité du modèle. Premièrement, les valeurs manquantes ont été identifiées et supprimées du jeu de données, car elles peuvent introduire des biais et fausser les résultats du modèle. Deuxièmement, une variable qui était initialement numérique a été convertie en facteur. Cette transformation est cruciale car

elle permet de refléter la nature catégorielle de la variable en question, facilitant ainsi son interprétation dans le modèle et améliorant la précision des estimations des paramètres.

4.2 Modèle global et analyse exploratoires des données

4.2.1 Analyse exploratoires des données

Corrélation des variables explicatives : L'objectif est de repérer des corrélations potentiellement fortes entre les variables indépendantes (multicolinéarité), ce qui pourrait affecter la précision des estimations du modèle, et sélectionner des variables indépendantes qui ne sont pas fortement corrélées entre elles. Cela permet d'assurer que chaque variable apporte des informations uniques au modèle, améliorant ainsi la fiabilité des classificationss.

FIGURE 2 – Matrice de Corrélation des variables

On remarque que les corrélations entre les variables indépendantes semblent être généralement faibles à modérées, ce qui est préférable pour la régression logistique car cela réduit le risque de multicolinéarité.

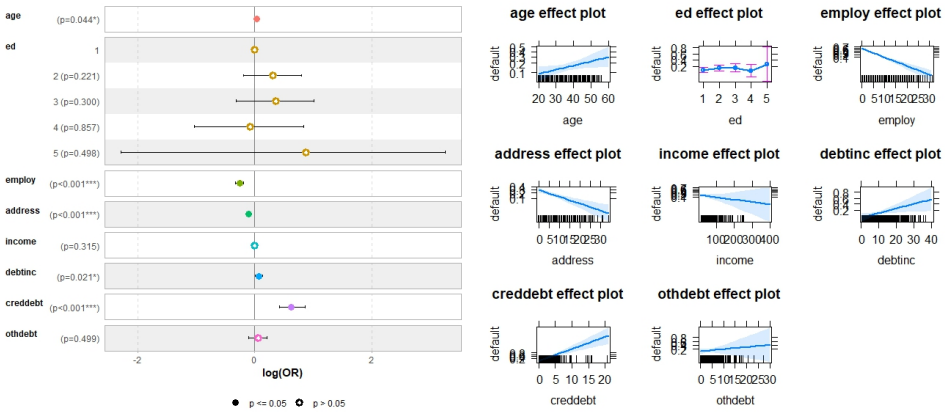
Modèle global et significativité des variables

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.590424   0.605329  -2.627   0.0086 **
age          0.035497   0.017586   2.018   0.0435 *
ed2          0.307643   0.251628   1.223   0.2215
ed3          0.352652   0.339949   1.037   0.2996
ed4         -0.085158   0.472917  -0.180   0.8571
ed5          0.876191   1.293810   0.677   0.4983
employ      -0.260676   0.033407  -7.803 6.04e-15 ***
address     -0.105429   0.023263  -4.532 5.84e-06 ***
income      -0.007823   0.007785  -1.005   0.3149
debtinc      0.070704   0.030596   2.311   0.0208 *
creddebt     0.624980   0.112917   5.535 3.11e-08 ***
othdebt      0.052999   0.078467   0.675   0.4994
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 804.36  on 699  degrees of freedom
Residual deviance: 549.56  on 688  degrees of freedom
AIC: 573.56
```

FIGURE 3 – Modèle de régression logistique global.

Le modèle de régression logistique global a été élaboré pour évaluer la signification statistique des différentes variables. Les résultats indiquent clairement que les variables "employ", "creddebt", et "address" sont les plus significatives, avec des valeurs de p extrêmement basses. En revanche, les variables "debtinc" et "age" montrent une signification statistique moindre, bien que leur impact reste notable.
En synthèse, les résultats mettent en lumière l'importance prépondérante de la stabilité de l'emploi, du montant total des dettes de crédit, et de la durée à la même adresse dans la classification de la variable dépendante, tandis que le ratio dette sur revenu et l'âge présentent des influences moins marquées mais néanmoins significatives.

Effet de chaque variable explicative sur la variable cibe



(a) Forest plot (b) Effect plot

FIGURE 4 – Effet de chaque variable explicative sur la variable cibe.

— Le premier graphique est un forest plot montrant les rapports de cotes logarithmiques (log(OR)) pour différentes variables dans le modèle de régression logistique. Les points représentent l'estimation du log(OR), et les lignes horizontales représentent les intervalles de confiance. Un log(OR) supérieur à zéro suggère une probabilité accrue de l'issue (défaut) étant donné une augmentation d'une unité dans le prédicteur, tandis qu'un log(OR) inférieur à zéro suggère une probabilité diminuée. Les points qui ne croisent pas la ligne verticale à zéro indiquent des effets statistiquement significatifs au niveau $p < 0,05$.

— Le deuxième graphique est une série de graphiques montrant l'effet de chaque prédicteur sur la probabilité de défaut. Ces graphiques sont probablement basés sur les probabilités prédites à partir du modèle de régression logistique, montrant comment la probabilité de défaut change avec différentes valeurs des prédicteurs.

4.2.2 Validation de modèle global

Pour valider le modèle global, on utilise le test de rapport de vraisemblance. On teste l'hypothèse de nullité de tous les coefficients. w_p pour $i = 1, \ldots, p$ (à l'exception de l'intercept w_0),L'hypothèse nulle correspond à l'idée que toutes les variables explicatives n'ont pas d'effet significatif sur la variable dépendante.

Il s'agit de tester $H_0 : \omega_1 = \omega = \ldots = \omega_p = 0$ contre $H_1 : \exists i$ tel que $\omega_i \neq 0$.

```
> #### Tester (avec rapport de vraisemblance) la validité du modèle complet ####
> # i.e., tester  H0 : ``w1=0, ..., wp+1=0'' contre H1 : ``le contraire de H_0''
> Sn <- modele.RL$null.deviance - modele.RL$deviance #la statistique du rapport de vraisemblance
> print(Sn)
[1] 254.8007
> ddl <- modele.RL$df.null - modele.RL$df.residual #nombre de degrés de liberté de la loi limite de Sn, sous H_0
> print(ddl)
[1] 11
> pvalue <- pchisq(q = Sn, df = ddl, lower.tail = F) #p_value du test : P(Z>Sn) où Z suit une loi du chi^2(ddl)
> print(pvalue) #on obtient 1.253064e-27, on rejette H0, donc le modèle est "très" significatif
[1] 2.758265e-48
```

FIGURE 5 – Test de rapport de vraisemblance pour validation du modèle global

La p-value obtenue est 2.758265×10^{-48}.Comme la p-value est bien inférieure à un seuil de 0,05, on rejette l'hypothèse nulle H_0. Ainsi, le modèle global est considéré comme très significatif, suggérant que l'ensemble des coefficients est significativement différent de zéro.

4.3 Selection du Modèle

4.3.1 Justification de l'approche exhaustive pour la sélection des variables

Pour l'analyse de nos données, la méthode exhaustive a été choisie comme stratégie de sélection des variables explicatives dans notre modèle de régression logistique. Cette technique a été préférée en raison de notre ensemble de données comportant un nombre raisonnable de 8 variables explicatives (une variable catégorielle de 5 niveau et 7 variables numériques) pour un échantillon de 700 observations.

La méthode exhaustive se distingue par sa capacité à tester toutes les combinaisons possibles de variables explicatives, assurant ainsi qu'aucun modèle potentiellement significatif n'est omis de l'analyse. Cela est particulièrement crucial dans notre contexte où le volume de données permet un traitement exhaustif sans contraintes computationnelles excessives. En procédant ainsi, nous maximisons nos chances de détecter la configuration optimale des prédicteurs qui influencent le plus la variable dépendante, ce qui renforce la fiabilité des conclusions tirées de notre modèle de régression logistique.

4.3.2 Comparaison des modèles avec traitement différencié de la variable Ed

Dans notre démarche de sélection des variables pour le modèle de régression logistique, nous allons considérer deux approches différentes en ce qui concerne le traitement de la variable catégorielle **Ed**(niveau d'éducation).

Première approche : Construire un modèle global en traitant la variable Ed comme une variable catégorielle unique avec diffèrent niveau
Cette approche permet d'apprécier l'effet global de cette variable sur la variable cible default.

Deuxième approche : Construire un modèle en traitant chaque niveau d'éducation comme une variable distincte et en remplaçant chaque niveau par une variable indicatrice,

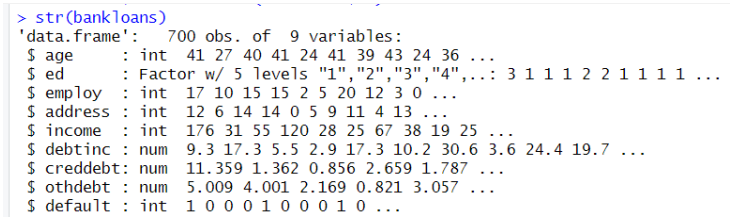


FIGURE 6 – Structure de Modèle avec la Variable 'Ed' en Catégorielle

à l'exception du premier niveau.

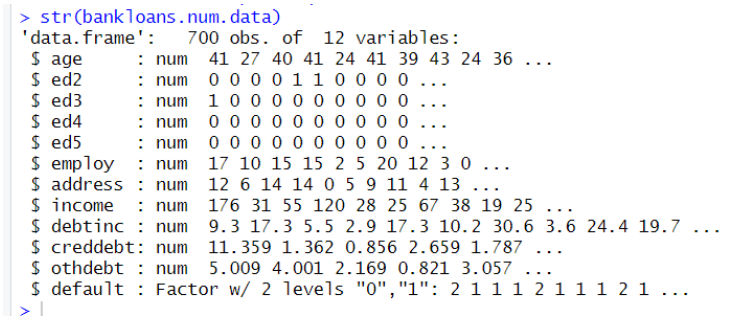


FIGURE 7 – Structure de Modèle en remplaçant chaque niveau par une variable indicatrice, à l'exception du premier niveau

Dans cette approche les variables sont celle de la matrice de design où chaque modalité de la variable Ed est traitée comme sa propre variable binaire . Cela impliquait de créer des variables indicatrices pour chaque niveau d'éducation, à l'exception de la catégorie de référence, afin d'analyser l'effet spécifique de chaque niveau d'éducation individuellement sur la variable cible default.

Remarque : La comparaison de ces deux modèles nous aide à déterminer si le détail supplémentaire fourni par la matrice de design aboutit à un modèle significativement plus performant par rapport à l'approche plus synthétique du modèle global.

4.3.3 Résultats et Interpretations

Résultat de la première approche : Le résultat de la selections des modèles avec les critères AIC AICC et BIC est le suivant.

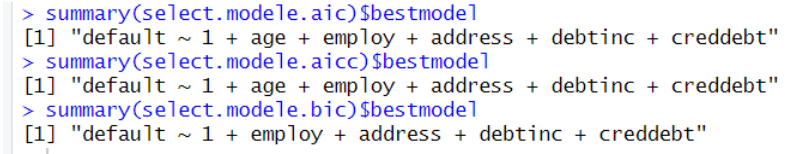


FIGURE 8 – Résultats de Sélection de Modèle avec la Variable 'Ed' en Catégorielle

Résultat de la deuxième approche : Le résultat de la selections des modeles avec les critères AIC AICC et BIC est le suivant.

```
> summary(select.modele.num.aic)$bestmodel
[1] "default ~ 1 + age + employ + address + debtinc + creddebt"
> summary(select.modele.num.aicc)$bestmodel
[1] "default ~ 1 + age + employ + address + debtinc + creddebt"
> summary(select.modele.num.bic)$bestmodel
[1] "default ~ 1 + employ + address + debtinc + creddebt"
```

FIGURE 9 – Résultats de Sélection de Modèle avec des variables indicatrices pour chaque niveau d'éducation

Remarques :
- Après avoir appliqué deux approches distinctes pour traiter la variable 'éducation', nous constatons une convergence remarquable vers le même modèle pour toutes les approches de sélection du modèle (AIC, AICC et BIC).
- Le critère BIC se distingue en excluant la variable 'âge'. Cette sélection est conforme à nos observations antérieures, qui montraient une association moins significative de la variable 'âge' avec la variable de réponse .

Conclusion : Le modèle retenu selon le critère BIC, privilégiant la simplicité et la précision prédictive, est ainsi considéré comme le meilleur. Il se concentre sur les variables ayant les influences les plus substantielles et statistiquement significatives sur la probabilité de défaut, en accord avec nos analyses préliminaires. Cela renforce la pertinence de notre modèle final pour des applications pratiques.

4.4 Évaluation du Modèle

4.4.1 Choix de la méthode de validation croisé pour évaluer l'erreur de classification

Nous avons choisi pour l'évaluation de l'erreur de classification la méthode Leave-One-Out (LOO). Cette approche est particulièrement adaptée à notre ensemble de données qui semble relativement restreint avec 700 observations et 8 variables explicatives. Le LOO implique l'utilisation de chaque observation comme ensemble de test une seule fois, tandis que le reste des données est utilisé pour l'entraînement. Cette méthode est attrayante dans le contexte de petites bases de données, car elle maximise l'utilisation des données disponibles.

4.4.2 Erreur de classification : (Modèle réduit VS Modèle globale)

Dans le contexte de l'erreur de classification, nous examinerons les résultats de notre modèle à travers deux perspectives distinctes : le modèle réduit et le Modèle global. Le modèle réduit implique l'utilisation d'un ensemble de variables explicatives plus restreint, permettant une analyse de la performance dans des conditions simplifiées. En revanche, le Modèle global intègre toutes les variables explicatives disponibles, fournissant une évaluation de la performance dans des conditions plus complexes. Cette approche permettra une compréhension approfondie de la manière dont le modèle se comporte dans différentes configurations, permettant ainsi des insights précieux pour d'éventuelles

itérations et améliorations du modèle.

- Modèle réduit

```
> cv.err.modele.reduit <- cv.glm(data = bankloans.num.data,
glmfit = bankloans_modele_glm, cost = cout, K = K)
> cv.err.modele.reduit$delta[1]
[1] 0.1871429
>
```

FIGURE 10 – Erreur de classification du modèle réduit

- Modèle global

```
> cv.err.modele.complet <- cv.glm(data = bankloans.num.data,
glmfit = modele.glm.complet, cost = cout, K = K)
> cv.err.modele.complet$delta[1]
[1] 0.1928571
>
```

FIGURE 11 – Erreur de classification du Modèle global

On observe que l'erreur de classification pour le modèle réduit est d'environ 0.1871429, tandis que pour le Modèle global, elle est d'environ 0.1971429. Cette constatation suggère que le modèle réduit présente une erreur de classification estimée légèrement inférieure à celle du Modèle global. Ces résultats laissent entrevoir que le modèle réduit, construit sur la base de la sélection de variables à l'aide du critère BIC, pourrait offrir une meilleure performance prédictive que le Modèle global.

4.4.3 Table de confusion : (Modèle réduit VS Modèle global)

Modèle réduit

```
> # Affichez la matrice de confusion
> conf_matrix_opt_bic
         prediction
reference   0   1
        0 478  39
        1  91  92
```

FIGURE 12 – Table de confusion du modèle réduit

Modèle global

```
> # Affichez la matrice de confusion
> conf_matrix_mod_glob
         prediction
reference   0   1
        0 479  38
        1  90  93
```

FIGURE 13 – Table de confusion du Modèle global

les résultats des matrices de confusion pour le modèle réduit et le Modèle global sont pratiquement les mêmes, cela pourrait indiquer une performance similaire entre les deux modèles en termes de classification. Cela suggère que le modèle réduit peut être une option attrayante, notamment pour sa simplicité.

4.4.4 Conclusion

En résumé, l'analyse des résultats indique que le modèle réduit, construit à l'aide du critère BIC pour la sélection des variables, présente des avantages potentiels par rapport au Modèle global. Bien que les performances prédictives des deux modèles soient similaires, le modèle réduit semble offrir une légère amélioration tout en étant plus simple. La décision finale devrait tenir compte de l'équilibre entre la performance prédictive, la simplicité du modèle.

4.5 Test de significativité des variables du modèle réduit

Dans cette section, nous procéderons au test de significativité pour chaque variable du modèle réduit en évaluant l'hypothèse nulle $H_0 : w_i = 0$ contre l'hypothèse alternative $H_1 : w_i \neq 0$, où i varie de 1 à p, représentant les différentes variables du modèle rèduit.

- Pour le test de rapport de vraisemblance, nous avons réalisé ce test pour chaque variable individuellement.

- Pour le test de Wald, nous avons obtenu les résultats en appliquant la fonction `glm` pour la régression logistique du modèle réduit.

```
> sort(vect.des.pvalues.MV)
      employ      debtinc     creddebt      address
7.651686e-26 5.465218e-16 1.400078e-06 1.374216e-05
> sort(vect.des.pvalues.Wald)
      employ     creddebt      debtinc      address
5.313346e-18 5.176853e-11 1.934917e-06 3.394409e-05
>
```

FIGURE 14 – Test de significativité des variables du modèle réduit en ordre croissant

Interprétations :
- Les p-values très faibles pour toutes les variables dans les deux tests indiquent que chaque variable a un impact significatif sur la variable dépendante.

- La variable "employ" semble être la plus influente, suivie de près par "debtinc", "creddebt", et "address".
- Ces résultats suggèrent que le modèle de régression, avec ces variables, offre une explication significative de la variabilité de la variable dépendante.

En conclusion, toutes les variables incluses dans le modèle semblent contribuer de manière substantielle à la classification de la variable dépendante, selon les tests de Wald et de rapport de vraisemblance

4.6 Scoring : (Modèle global et Modèle réduit)

Dans cette section, nous évaluerons tous les scores des fonctions Logit, LDA, QDA, SVM, etc., pour le modèle global choisi selon le critère du BIC et nous comparerons ces résultats avec le Modèle réduit.

Remarque : Dans cette étude, nous avons opté pour un échantillonnage stratifié en utilisant la même gaine pour tous les résultats afin de les comparer. Cette approche nous permet de garantir une représentation équilibrée des différentes catégories ou strates présentes dans nos données, renforçant ainsi la fiabilité de notre analyse comparative.

4.6.1 Modèle global

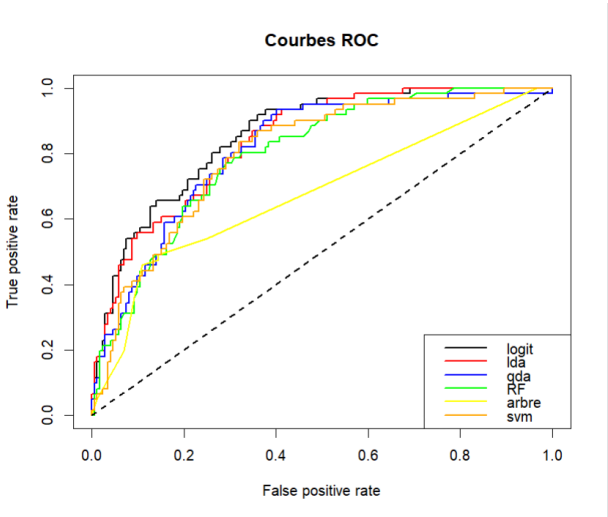


FIGURE 15 – Courbe ROC des scores pour le Modèle global

Remarque : Selon l'analyse de la courbe ROC, il est observé que le score obtenu avec le modèle logistique (logit) se rapproche le plus de 1. Cette observation suggère que le modèle logistique présente une performance supérieure en termes de scoring. Pour confirmer cette constatation, nous vérifions nos résultats en calculant l'aire sous la courbe ROC (AUC).

Remarque : Suite au calcul de l'AUC, il est notable que le modèle logistique (logit) affiche la plus grande aire sous la courbe, obtenant un résultat de 0.855. En comparaison avec les autres modèles, cette valeur plus élevée suggère que le modèle logit excelle en

```
[1] "La valeur de l'AUC de chacun des scores : "
[2] ""
[3] "logit =  0.855223027068243"
[4] "lda =  0.838353030880671"
[5] "qda =  0.813095691955777"
[6] "RF =  0.797464735036218"
[7] "arbre =  0.677277926038887"
[8] "svm =  0.800895920701488"
> |
```

FIGURE 16 – Valeur de l'AUC pour chacun des scores pour le Modèle global

termes de capacité de scoring. Ainsi, ces résultats renforcent l'idée que le modèle logistique est le choix optimal pour la scoring.

4.6.2 Modèle réduit

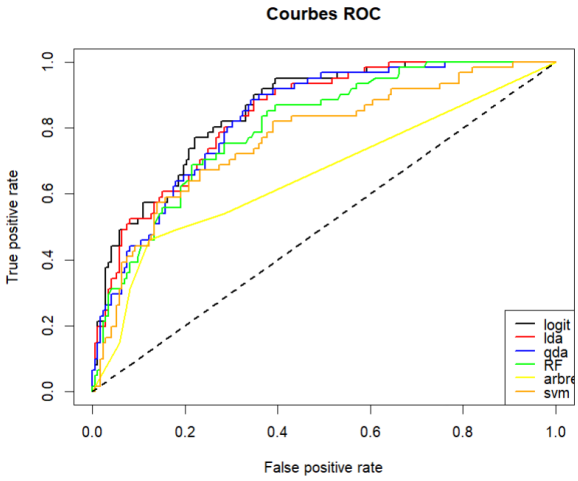


FIGURE 17 – Courbe ROC des scores pour le modèle réduit

Remarque : En appliquant la même analyse à la courbe ROC du modèle réduit, on constate que le modèle logistique (logit) maintient son avantage en affichant un score qui se rapproche le plus de 1. Cette tendance suggère que, même dans le contexte du modèle réduit, le modèle logistique conserve une performance notable en termes de scoring. Pour étayer cette observation, nous confirmons ces résultats en calculant l'aire sous la courbe ROC (AUC).

```
[1] "La valeur de l'AUC de chacun des scores : "
[2] ""
[3] "logit =  0.852459016393443"
[4] "lda =  0.839496759435761"
[5] "qda =  0.829203202439955"
[6] "RF =  0.801849027830728"
[7] "arbre =  0.65797750667175"
[8] "svm =  0.767632481890965"
```

FIGURE 18 – Valeur de l'AUC pour chacun des scores pour le modèle réduit

Remarque : Après avoir étendu notre analyse à la courbe ROC du modèle réduit, les résultats confirment également que le modèle logistique (logit) maintient sa supériorité. L'AUC obtenue, évaluée à 0.852, demeure la plus élevée parmi les autres modèles. Cette constatation souligne que, même dans le contexte du modèle réduit, le modèle logistique excelle en termes de capacité de scoring. Ainsi, ces résultats renforcent l'idée que le modèle logistique demeure le choix optimal pour le scoring même avec le modèle réduit.

4.6.3 Conclusion

En comparant les performances des modèles global et réduit, on observe une convergence remarquable dans les résultats obtenus pour les deux cas étudiés. Que ce soit à travers l'analyse de la courbe ROC ou du calcul de l'AUC, les deux modèles, qu'ils soient global ou réduit, affichent des performances presque équivalentes.

Cette cohérence dans les résultats entre les deux modèles suggère que, malgré la simplification du modèle réduit, les variables sélectionnées maintiennent leur pouvoir discriminatoire, et le modèle logistique demeure optimal pour la tâche de scoring. Cette convergence renforce ainsi la robustesse et la fiabilité des conclusions, indiquant que le modèle réduit, bien que plus simple, offre des performances comparables au modèle global dans notre contexte d'analyse.

5 Conclusion

Dans le cadre de notre étude sur l'analyse de risque de crédit bancaire, nous avons entrepris une approche rigoureuse en utilisant la régression logistique pour examiner la variable binaire "default". En premier lieu, nous avons développé un modèle global exhaustif, explorant les relations complexes entre les variables et la probabilité de défaut.

Par la suite, dans une démarche de simplification, nous avons réduit notre modèle, éliminant les variables moins significatives pour former un modèle réduit. La comparaison entre le modèle global et le modèle réduit a révélé que ce dernier, malgré sa simplicité, offre des performances tout aussi valables, soulignant son potentiel en termes de praticité et de compréhension.

Une partie significative de notre projet a également porté sur l'application de différentes techniques de scoring, telles que la régression logistique, SVM, LDA, QDA, etc., aussi bien pour le modèle global que pour le modèle réduit. Nos résultats ont convergé vers une conclusion claire : le modèle logistique, avec son équivalent pour le modèle réduit, s'est démarqué comme le choix optimal en termes de scoring. Sa capacité à assigner des scores de manière précise et fiable en fait un outil essentiel dans notre évaluation du risque de crédit.

En somme, notre étude démontre que le modèle réduit, bien qu'offrant une simplification appréciable, ne compromet pas la qualité des classifications. En outre, la préférence pour le modèle logistique dans le contexte du scoring renforce la pertinence de ce choix méthodologique. Ces conclusions combinées soulignent l'efficacité du modèle réduit et du modèle logistique dans notre analyse de risque de crédit bancaire, fournissant ainsi des insights précieux pour la prise de décision dans ce domaine crucial.

Références

[1] Amor Keziou. *INFO0901 Apprentissage statistique et Data mining* . Laboratoire des mathématiques de Reims, 2023/2024.

A ANNEXE

```
#vider la mémoire
rm(list = ls())

library(corrplot)
library(gtsummary)
library(GGally)
library(forestmodel)
library(effects)
library(report)
library(kernlab)
library(MASS)
library(randomForest)
library(rpart)
library (e1071)
library(class)
library(ROCR)

#### Modèle de régression logistique binomial pour la classification binaire ####

bankloans <- read.csv("C:/Users/MSI/Desktop/M2 Intelligence Artificielle/
Apprentissage stat et data mining/projet/archive (1)/bankloans.csv", stringsAsFactors=TRUE)
View(bankloans)
str(bankloans)

##############################################################################
### prétraitement des données###
##############################################################################

#supprimer les ligne ou il'ya des valeurs manquantes
bankloans <- na.omit(bankloans)
#str(bankloans)

#la variable ed est une variable categoriale mais elle est importé comme une variable numerique
#On la change en variable categorielle en utilisant la commande as.factors
bankloans$ed<-as.factor(bankloans$ed)
str(bankloans)

##############################################################################
### Modèle global###
##############################################################################

# matrcie de correlation des variables
X <- model.matrix(default ~., data = bankloans)[,-1]
XX <- cbind(as.data.frame(X), default = bankloans[,"default"])
# Calcul de la matrice de corrélation
correlation_matrix <- cor(XX)
corrplot(correlation_matrix, method = "circle")
# Affichage de la matrice de corrélation
print(correlation_matrix)


#faire une régression logistique de la variable binaire default en fonction des variables (explicatives)
#de la bd bankloans :
modele.RL <- glm(formula = default~ .,  data = bankloans, family = binomial)


#Affichage
print(modele.RL)
summary(modele.RL)
attributes(modele.RL)
tbl_regression(modele.RL, exponentiate = FALSE)
ggcoef_model(modele.RL, exponentiate = FALSE)
forest_model(modele.RL, exponentiate = FALSE)
plot(allEffects(modele.RL))
report(modele.RL)

#### Tester (avec rapport de vraisemblance) la validité du Modèle global ####
```

```
# i.e., tester  H0 : ‘‘w1=0, ..., wp+1=0’’ contre H1 : ‘‘le contraire de H_0’’
Sn <- modele.RL$null.deviance - modele.RL$deviance #la statistique du rapport de vraisemblance
print(Sn)
ddl <- modele.RL$df.null - modele.RL$df.residual #nombre de degrés de liberté de la loi limite de Sn, sous H_0
print(ddl)
pvalue <- pchisq(q = Sn, df = ddl, lower.tail = F) #p_value du test : P(Z>Sn) où Z suit une loi du chi^2(ddl)
print(pvalue) #on obtient 1.253064e-27, on rejette H0, donc le modèle est "très" significatif




##############################################################################
#### Sélection de modèles (de variables) selon les critères AIC, AICC et BIC ###
##############################################################################

#### Recherche exhaustive ####
library(glmulti)
#AIC
select.modele.aic <- glmulti(default~.,  data = bankloans, family = binomial, level = 1,
                             fitfunction = glm, crit = "aic",
                             plotty = FALSE, method = "h")
#BIC
select.modele.bic <- glmulti(default ~.,  data = bankloans, family = binomial, level = 1,
                             fitfunction = glm, crit = "bic",
                             plotty = FALSE, method = "h")
#AICC
select.modele.aicc <- glmulti(default ~.,  data = bankloans, family = binomial, level = 1,
                              fitfunction = glm, crit = "aicc",
                              plotty = FALSE, method = "h")

summary(select.modele.aic)$bestmodel
summary(select.modele.aicc)$bestmodel
summary(select.modele.bic)$bestmodel

##Si on veut choisir parmi les variables de la matrice de design, on fait comme suit :

#supprimer les ligne ou il’ya des valeurs manquantes
bankloans <- na.omit(bankloans)
XX <- model.matrix(default ~., data = bankloans)[,-1] #cette fonction construit la matrice de design en remplaçant
#chacune des variables qualitatives pour les indicatrices
#de ses modalités (la première modalité est supprimée)
#on supprime la première colonne correspondant à l’intercept
bankloans.num.data <- cbind(as.data.frame(XX), default = as.factor(bankloans[,"default"])) #bd constituée que de variables e
#et une variable réponse qualitative (binaire)
str(bankloans.num.data)
View(bankloans.num.data)

#AIC
select.modele.num.aic <- glmulti(default ~.,  data = bankloans.num.data, family = binomial, level = 1,
                                 fitfunction = glm, crit = "aic",
                                 plotty = FALSE, method = "h")
#BIC
select.modele.num.bic <- glmulti(default ~.,  data = bankloans.num.data, family = binomial, level = 1,
                                 fitfunction = glm, crit = "bic",
                                 plotty = FALSE, method = "h")
#AICC
select.modele.num.aicc <- glmulti(default ~.,  data = bankloans.num.data, family = binomial, level = 1,
                                  fitfunction = glm, crit = "aicc",
                                  plotty = FALSE, method = "h")

summary(select.modele.num.aic)$bestmodel
summary(select.modele.num.aicc)$bestmodel
summary(select.modele.num.bic)$bestmodel




##############################################################################
#### Estimation de l’erreur de classification par les méthodes de validation croisée ####
##############################################################################
```

```
# Nous allons évaluer l'erreur de classification issue du modèle optimal selon le critère BIC
str(bankloans.num.data)

#rappelons le modèle optimal selon BIC
modele.opt.bic.formula <- summary(select.modele.num.bic)$bestmodel
modele.opt.bic.formula

n <- nrow(bankloans.num.data)

library(boot)
bankloans_modele_glm <- glm(formula = modele.opt.bic.formula, data = bankloans.num.data, family = binomial)
cout <- function(r, pi) mean(abs(r-pi) > 0.5)
#le cas K = n, n étant le nombre d'observations, correspond à la méthode leave-one-out :
n <- nrow(bankloans.num.data)
K <- n
cv.err <- cv.glm(data = bankloans.num.data, glmfit = bankloans_modele_glm, cost = cout, K = K)
cv.err$delta[1]

#comparaison avec le Modèle global
modele.glm.complet <- glm(formula = default ~., data = bankloans.num.data, family = binomial)
cv.err.modele.complet <- cv.glm(data = bankloans.num.data,
                                glmfit = modele.glm.complet, cost = cout, K = K)
cv.err.modele.complet$delta[1]




##############################################################################
###matrice de confusion###
##############################################################################

## modèle global
# Ajuster le modèle global
bankloans_modele_glm <- glm(formula = default ~., data = bankloans.num.data, family = binomial)

# Obtenez les classificationss du modèle global
pi_mod_glob <- predict(bankloans_modele_glm, type = "response")

# Définissez la matrice de confusion pour le modèle global
conf_matrix_mod_glob <- table(reference = bankloans.num.data$default, prediction = ifelse(pi_mod_glob  > 0.5, 1, 0))

# Affichez la matrice de confusion
conf_matrix_mod_glob

# Calcul des résultats (faux positif, vrai positif, faux négatif, vrai négatif)
fp_mod_glob <- conf_matrix_mod_glob[1, 2]
fn_mod_glob <- conf_matrix_mod_glob[2, 1]
vp_mod_glob <- conf_matrix_mod_glob[2, 2]
vn_mod_glob <- conf_matrix_mod_glob[1, 1]

# Affichez les résultats
cat("Faux Positifs (FP) :", fp_mod_glob, "\n")
cat("Faux Négatifs (FN) :", fn_mod_glob, "\n")
cat("Vrais Positifs (VP) :", vp_mod_glob, "\n")
cat("Vrais Négatifs (VN) :", vn_mod_glob, "\n")



## modèle réduit

# Ajuster le modèle optimal selon BIC

bankloans_modele_glm <- glm(formula = modele.opt.bic.formula, data = bankloans.num.data, family = binomial)

# Obtenez les classifications du modèle optimal selon BIC
pi_opt_bic <- predict(bankloans_modele_glm, type = "response")

# Définissez la matrice de confusion pour le modèle optimal selon BIC
conf_matrix_opt_bic <- table(reference = bankloans.num.data$default, prediction = ifelse(pi_opt_bic > 0.5, 1, 0))
```

```
# Affichez la matrice de confusion
conf_matrix_opt_bic

# Calcul des résultats (faux positif, vrai positif, faux négatif, vrai négatif)
fp_opt_bic <- conf_matrix_opt_bic[1, 2]
fn_opt_bic <- conf_matrix_opt_bic[2, 1]
vp_opt_bic <- conf_matrix_opt_bic[2, 2]
vn_opt_bic <- conf_matrix_opt_bic[1, 1]

# Affichez les résultats
cat("Faux Positifs (FP) :", fp_opt_bic, "\n")
cat("Faux Négatifs (FN) :", fn_opt_bic, "\n")
cat("Vrais Positifs (VP) :", vp_opt_bic, "\n")
cat("Vrais Négatifs (VN) :", vn_opt_bic, "\n")

##############################################################################
#### classer les variables d'un modèle selon leur niveau de significativité ####
##############################################################################


# on utilise une approche test : (utiliser les p_values des tests correspondants,
# test de Wald et/ou test du rapport de vraisemblance).

# Considérons par exemple le modèle optimal selon BIC
# et classons les variables par ordre décroissant des valeurs des p_values de chacun des tests de Wald

#rappelons le modèle optimal selon BIC
modele.opt.bic.formula <- summary(select.modele.num.bic)$bestmodel
modele.opt.bic.formula #ce modèle utilise les v.a. explicatives :"employ + address + debtinc + creddebt"
#la bd correspondante est la suivante
bankloans.opt.data <- bankloans.num.data[,c("employ", "address", "debtinc", "creddebt",
                                            "default")]
str(bankloans.opt.data)
View(bankloans.opt.data)

modele <- glm(formula = default~., data = bankloans.opt.data, family = binomial)
#Affichage
tbl_regression(modele, exponentiate = FALSE)
ggcoef_model(modele, exponentiate = FALSE)
forest_model(modele, exponentiate = FALSE)
plot(allEffects(modele))

print(modele)
summary(modele)
tab.modele <- summary(modele)$coefficients
tab.modele <- as.data.frame(tab.modele)
str(tab.modele)
View(tab.modele)
vect.des.pvalues.Wald <- tab.modele[,"Pr(>|z|)"]
names(vect.des.pvalues.Wald) <- row.names(tab.modele)

#on supprime la pvalue de l'intercept
vect.des.pvalues.Wald <- vect.des.pvalues.Wald[!(names(vect.des.pvalues.Wald) == "(Intercept)")]
vect.des.pvalues.Wald

#ranger les variables par ordre croissant des valeurs des p_values
sort(vect.des.pvalues.Wald) # la variable la plus significative est depression, ensuite sexe, ensuite typedouleurD, ensuite

#On classe maintenant les variables par ordre décroissant des valeurs des pvalues de chacun des tests par maximum de vraisem
#### classement des variables par pvalues du test du rapport de vraisemblance, préférable à celui de Wald ####
modele <- glm(formula = default ~ ., data = bankloans.opt.data, family = binomial)
print(modele)

##on procède maintenant au test d'hypothèses par la statistique du rapport de vraisemblance, et au calcul des p_values corre

##Tester l'hypothèse : la variable employ n'est pas significative
#i.e., tester H_0 : w1 = 0 contre H_1 : w1 != 0
modele.reduit <- glm(default ~ ., data = bankloans.opt.data[,!(colnames(bankloans.opt.data) == "employ")], family = binomial
#Statistique du rapport de vraisemblance
Sn <- modele.reduit$deviance - modele$deviance
print(Sn)
```

```
pvalue.employ <- pchisq(q = Sn, df = 1, lower.tail = F) #donne P(Z>Sn) où Z est une variable suivant une chi2(1).
print(pvalue.employ)

##Tester H_0 : la variable address n'est pas significative
modele.reduit <- glm(default ~ ., data = bankloans.opt.data[,!(colnames(bankloans.opt.data)
                                                               == "address")], family = binomial)
#Statistique du rapport de vraisemblance
Sn <- modele.reduit$deviance - modele$deviance
print(Sn)
pvalue.address = pchisq(q = Sn, df = 1, lower.tail = F)
print(pvalue.address)

##Tester H_0 : la variable debtinc n'est pas significative
modele.reduit <- glm(default ~ ., data = bankloans.opt.data[,!(colnames(bankloans.opt.data)
                                                               == "debtinc")], family = binomial)
#Statistique du rapport de vraisemblance
Sn <- modele.reduit$deviance - modele$deviance
print(Sn)
pvalue.debtinc = pchisq(q = Sn, df= 1, lower.tail = F)
print(pvalue.debtinc)

##Tester H_0 : la variable creddebt n'est pas significative
modele.reduit <- glm(default ~ ., data = bankloans.opt.data[,!(colnames(bankloans.opt.data)
                                                               == "creddebt")], family = binomial)
#Statistique du rapport de vraisemblance
Sn <- modele.reduit$deviance - modele$deviance
print(Sn)
pvalue.creddebt <- pchisq(q = Sn, df = 1, lower.tail = F)
print(pvalue.creddebt)

#vecteur des p_values
vect.des.pvalues.MV <- c(pvalue.employ, pvalue.address, pvalue.creddebt, pvalue.debtinc)
names(vect.des.pvalues.MV) <- colnames(bankloans.opt.data[,!(colnames(bankloans.opt.data) == "default")])
vect.des.pvalues.MV

sort(vect.des.pvalues.MV) #on ordonne les variables de la plus significative à la moins significative
sort(vect.des.pvalues.Wald)
#on obtient le même classement ici! pas toujours le cas ...
#il est recommandé de retenir le classement selon le test du rapport de vraisemblances.


##############################################################################################
##Scoring##
##############################################################################################




#On partage la base en deux parties : train.set et test.set (avec échantillonnage stratifié)

##Modèle global
library(sampling)
set.seed(12)
mod.default <- as.vector(unique(bankloans.num.data$default))#les modalités de la variable réponse "default"
table(bankloans.num.data$default)[mod.default]
size <- as.vector(table(bankloans.num.data$default)[mod.default]/3) #on veut 1/3 d'observations pour le test
s <- strata(bankloans.num.data, stratanames="default", size=size, method="srswor")
test.set.index <- s$ID_unit
bankloans.test.set <- bankloans.num.data[test.set.index, ] #ensemble de test
bankloans.train.set <- bankloans.num.data[- test.set.index, ] #ensemble d'apprentissage

#On compare les scores construits par RegLog, lda, qda, forêts aléatoires,
#arbres de décision et svm
modele.logit <- glm(formula = default ~ ., data = bankloans.train.set, family = binomial)
modele.lda <- lda(formula = default ~., data = bankloans.train.set)
modele.qda <- qda(formula = default ~., data = bankloans.train.set)
modele.RF <- randomForest(formula = default ~., data = bankloans.train.set)
modele.arbre <- rpart(formula = default ~., data = bankloans.train.set)

set.seed(12)
tune.out <- tune(svm, default ~ ., data=bankloans.train.set, kernel="radial", scale=TRUE,
```

```
                ranges=list(cost = c(2^(-2) ,2^(-1), 1, 2^2, 2^3, 2^4),
                            gamma=c(2^(-3),2^(-2),2^(-1),1)))
tune.out
modele.svm <- tune.out$best.model

#On calcule ensuite pour chaque modèle le score des individus de l'échantillon de test
Score.logit <- predict(modele.logit, newdata = bankloans.test.set, type = "response")
Score.lda <- predict(modele.lda, newdata = bankloans.test.set, type = "prob")$posterior[, 2]
Score.qda <- predict(modele.qda, newdata = bankloans.test.set, type = "prob")$posterior[, 2]
Score.RF <- predict(modele.RF, newdata = bankloans.test.set, type = "prob")[, 2]
Score.arbre <- predict(modele.arbre, newdata = bankloans.test.set, type = "prob")[, 2]
Score.svm <- attributes(predict(modele.svm, newdata = bankloans.test.set, scale = TRUE,
                                decision.values = TRUE))$decision.values

#On trace maintenant les 6 courbes ROC
require(ROCR)
S1.pred <- prediction(Score.logit, bankloans.test.set$default)
S2.pred <- prediction(Score.lda, bankloans.test.set$default)
S3.pred <- prediction(Score.qda, bankloans.test.set$default)
S4.pred <- prediction(Score.RF, bankloans.test.set$default)
S5.pred <- prediction(Score.arbre, bankloans.test.set$default)
S6.pred <- prediction(Score.svm, bankloans.test.set$default)

roc1 <- performance(S1.pred, measure = "tpr", x.measure = "fpr")
roc2 <- performance(S2.pred, measure = "tpr", x.measure = "fpr")
roc3 <- performance(S3.pred, measure = "tpr", x.measure = "fpr")
roc4 <- performance(S4.pred, measure = "tpr", x.measure = "fpr")
roc5 <- performance(S5.pred, measure = "tpr", x.measure = "fpr")
roc6 <- performance(S6.pred, measure = "tpr", x.measure = "fpr")

par(mfrow = c(1,1))

#Tracer les courbes ROC des scores
plot(roc1, col = "black", lwd = 2, main = "Courbes ROC")
plot(roc2, add = TRUE, col = "red", lwd = 2)
plot(roc3, add = TRUE, col = "blue", lwd = 2)
plot(roc4, add = TRUE, col = "green", lwd = 2)
plot(roc5, add = TRUE, col = "yellow", lwd = 2)
plot(roc6, add = TRUE, col = "orange", lwd = 2)
bissect <- function(x) x
curve(bissect(x),  col = "black", lty = 2, lwd = 2, add = TRUE)
legend("bottomright", legend = c("logit", "lda", "qda", "RF", "arbre", "svm"),
       col = c("black", "red", "blue", "green", "yellow", "orange"), lty = 1, lwd = 2)

#Calcul de l'AUC
AUC1 <- performance(S1.pred, "auc")@y.values[[1]]
AUC2 <- performance(S2.pred, "auc")@y.values[[1]]
AUC3 <- performance(S3.pred, "auc")@y.values[[1]]
AUC4 <- performance(S4.pred, "auc")@y.values[[1]]
AUC5 <- performance(S5.pred, "auc")@y.values[[1]]
AUC6 <- performance(S6.pred, "auc")@y.values[[1]]

print(c("La valeur de l'AUC de chacun des scores : ", "",
        paste("logit = ", as.character(AUC1)),
        paste("lda = ", as.character(AUC2)),
        paste("qda = ", as.character(AUC3)),
        paste("RF = ", as.character(AUC4)),
        paste("arbre = ", as.character(AUC5)),
        paste("svm = ", as.character(AUC6))
))

#La courbe Lift de chacun des scores
lift1 <- performance(S1.pred, measure =  "tpr", x.measure =  "rpp")
lift2 <- performance(S2.pred, measure = "tpr", x.measure = "rpp")
lift3 <- performance(S3.pred, measure = "tpr", x.measure = "rpp")
lift4 <- performance(S4.pred, measure = "tpr", x.measure = "rpp")
lift5 <- performance(S5.pred, measure = "tpr", x.measure = "rpp")
lift6 <- performance(S6.pred, measure = "tpr", x.measure = "rpp")

plot(lift1, col = "black", lwd = 2, main = "Courbes Lift")
plot(lift2, add = TRUE, col = "red", lwd = 2)
```

```
plot(lift3, add = TRUE, col = "blue", lwd = 2)
plot(lift4, add = TRUE, col = "green", lwd = 2)
plot(lift5, add = TRUE, col = "yellow", lwd = 2)
plot(lift6, add = TRUE, col = "orange", lwd = 2)
bissect <- function(x) x
curve(bissect(x),  col = "black", lty = 2, lwd = 2, add = TRUE)
legend("bottomright", legend = c("logit", "lda", "qda", "RF", "arbre", "svm"),
       col = c("black", "red", "blue", "green", "yellow", "orange"),
       lty = 1, lwd = 2)



#### le score logit est le meilleur d'après les résulats précédents ####
modele.logit <- glm(formula = default ~ ., data = bankloans.train.set, family = binomial)
Score.logit <- predict(modele.logit, newdata = bankloans.test.set, type = "response")

#voici comment ordonner les individus de test (dans l'ordre décroissant des valeurs du score)
sort(Score.logit, decreasing=TRUE)

#le top dix :
sort(Score.logit, decreasing=TRUE)[1:10]

#Modèle reduit :modèle selectionné avec BIC

set.seed(12)
mod.default <- as.vector(unique(bankloans.opt.data$default))#les modalités de la variable réponse "default"
table(bankloans.opt.data$default)[mod.default]
size <- as.vector(table(bankloans.opt.data$default)[mod.default]/3) #on veut 1/3 d'observations pour le test
s <- strata(bankloans.opt.data, stratanames="default", size=size, method="srswor")
test.set.index <- s$ID_unit
bankloans.test.set <- bankloans.opt.data[test.set.index, ] #ensemble de test
bankloans.train.set <- bankloans.opt.data[- test.set.index, ] #ensemble d'apprentissage

#On compare les scores construits par RegLog, lda, qda, forêts aléatoires,
#arbres de décision et svm
modele.logit <- glm(formula = default ~ ., data = bankloans.train.set, family = binomial)
modele.lda <- lda(formula = default ~., data = bankloans.train.set)
modele.qda <- qda(formula = default ~., data = bankloans.train.set)
modele.RF <- randomForest(formula = default ~., data = bankloans.train.set)
modele.arbre <- rpart(formula = default ~., data = bankloans.train.set)

set.seed(12)
tune.out <- tune(svm, default ~ ., data=bankloans.train.set, kernel="radial", scale=TRUE,
                 ranges=list(cost = c(2^(-2) ,2^(-1), 1, 2^2, 2^3, 2^4),
                             gamma=c(2^(-3),2^(-2),2^(-1),1)))
tune.out
modele.svm <- tune.out$best.model

#On calcule ensuite pour chaque modèle le score des individus de l'échantillon de test
Score.logit <- predict(modele.logit, newdata = bankloans.test.set, type = "response")
Score.lda <- predict(modele.lda, newdata = bankloans.test.set, type = "prob")$posterior[, 2]
Score.qda <- predict(modele.qda, newdata = bankloans.test.set, type = "prob")$posterior[, 2]
Score.RF <- predict(modele.RF, newdata = bankloans.test.set, type = "prob")[, 2]
Score.arbre <- predict(modele.arbre, newdata = bankloans.test.set, type = "prob")[, 2]
Score.svm <- attributes(predict(modele.svm, newdata = bankloans.test.set, scale = TRUE,
                                decision.values = TRUE))$decision.values

#On trace maintenant les 6 courbes ROC
require(ROCR)
S1.pred <- prediction(Score.logit, bankloans.test.set$default)
S2.pred <- prediction(Score.lda, bankloans.test.set$default)
S3.pred <- prediction(Score.qda, bankloans.test.set$default)
S4.pred <- prediction(Score.RF, bankloans.test.set$default)
S5.pred <- prediction(Score.arbre, bankloans.test.set$default)
S6.pred <- prediction(Score.svm, bankloans.test.set$default)

roc1 <- performance(S1.pred, measure = "tpr", x.measure = "fpr")
roc2 <- performance(S2.pred, measure = "tpr", x.measure = "fpr")
roc3 <- performance(S3.pred, measure = "tpr", x.measure = "fpr")
roc4 <- performance(S4.pred, measure = "tpr", x.measure = "fpr")
roc5 <- performance(S5.pred, measure = "tpr", x.measure = "fpr")
```

```
roc6 <- performance(S6.pred, measure = "tpr", x.measure = "fpr")

par(mfrow = c(1,1))

#Tracer les courbes ROC des scores
plot(roc1, col = "black", lwd = 2, main = "Courbes ROC")
plot(roc2, add = TRUE, col = "red", lwd = 2)
plot(roc3, add = TRUE, col = "blue", lwd = 2)
plot(roc4, add = TRUE, col = "green", lwd = 2)
plot(roc5, add = TRUE, col = "yellow", lwd = 2)
plot(roc6, add = TRUE, col = "orange", lwd = 2)
bissect <- function(x) x
curve(bissect(x),  col = "black", lty = 2, lwd = 2, add = TRUE)
legend("bottomright", legend = c("logit", "lda", "qda", "RF", "arbre", "svm"),
       col = c("black", "red", "blue", "green", "yellow", "orange"), lty = 1, lwd = 2)

#Calcul de l'AUC
AUC1 <- performance(S1.pred, "auc")@y.values[[1]]
AUC2 <- performance(S2.pred, "auc")@y.values[[1]]
AUC3 <- performance(S3.pred, "auc")@y.values[[1]]
AUC4 <- performance(S4.pred, "auc")@y.values[[1]]
AUC5 <- performance(S5.pred, "auc")@y.values[[1]]
AUC6 <- performance(S6.pred, "auc")@y.values[[1]]

print(c("La valeur de l'AUC de chacun des scores : ", "",
        paste("logit = ", as.character(AUC1)),
        paste("lda = ", as.character(AUC2)),
        paste("qda = ", as.character(AUC3)),
        paste("RF = ", as.character(AUC4)),
        paste("arbre = ", as.character(AUC5)),
        paste("svm = ", as.character(AUC6))
))

#La courbe Lift de chacun des scores
lift1 <- performance(S1.pred, measure =  "tpr", x.measure =  "rpp")
lift2 <- performance(S2.pred, measure = "tpr", x.measure = "rpp")
lift3 <- performance(S3.pred, measure = "tpr", x.measure = "rpp")
lift4 <- performance(S4.pred, measure = "tpr", x.measure = "rpp")
lift5 <- performance(S5.pred, measure = "tpr", x.measure = "rpp")
lift6 <- performance(S6.pred, measure = "tpr", x.measure = "rpp")

plot(lift1, col = "black", lwd = 2, main = "Courbes Lift")
plot(lift2, add = TRUE, col = "red", lwd = 2)
plot(lift3, add = TRUE, col = "blue", lwd = 2)
plot(lift4, add = TRUE, col = "green", lwd = 2)
plot(lift5, add = TRUE, col = "yellow", lwd = 2)
plot(lift6, add = TRUE, col = "orange", lwd = 2)
bissect <- function(x) x
curve(bissect(x),  col = "black", lty = 2, lwd = 2, add = TRUE)
legend("bottomright", legend = c("logit", "lda", "qda", "RF", "arbre", "svm"),
       col = c("black", "red", "blue", "green", "yellow", "orange"),
       lty = 1, lwd = 2)



#### le score logit est le meilleur d'après les résulats précédents ####
modele.logit <- glm(formula = default ~ ., data = bankloans.train.set, family = binomial)
Score.logit <- predict(modele.logit, newdata = bankloans.test.set, type = "response")

#voici comment ordonner les individus de test (dans l'ordre décroissant des valeurs du score)
sort(Score.logit, decreasing=TRUE)

#le top dix :
sort(Score.logit, decreasing=TRUE)[1:10]
```