Prédire le décrochage ou la réussite scolaire des élèves

Soutenance de projet

Coërchon Colin & Coutrot Léos

Chargé de Projet : Kylliann De Santiago
Responsable de l'UE : Mathilde Mougeot

ENSIIE

18 décembre 2023

Sommaire

La réussite scolaire : enjeu politique et économique

- **8.2% :** le taux d'abandon scolaire en 2019
- **168,8 milliards d'euros :** dépense intérieure d'éducation en 2021
- **Une année** d'études coûte en moyenne **11 310 €** par élève
- L'abandon scolaire est un **enjeu européen**

La réussite scolaire : enjeu politique et économique

- **8.2% :** le taux d'abandon scolaire en 2019
- **168,8 milliards d'euros :** dépense intérieure d'éducation en 2021
- **Une année** d'études coûte en moyenne **11 310 €** par élève
- L'abandon scolaire est un **enjeu européen**

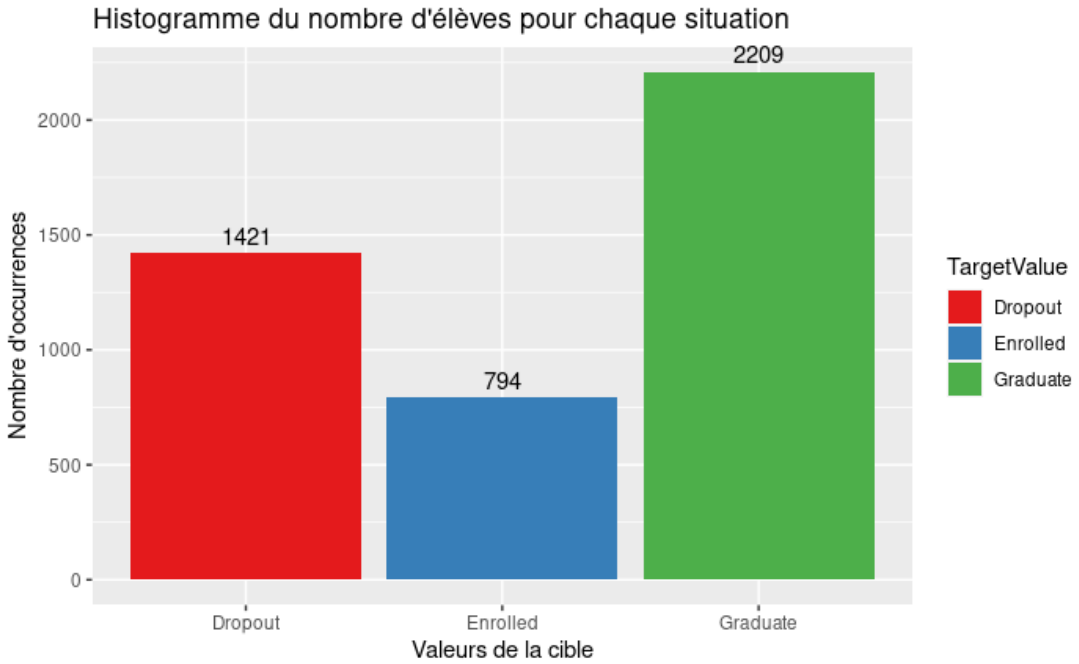
Problématique

Identifier les facteurs clés influençant la réussite ou l'échec académique des étudiants, afin de minimiser l'échec dans l'enseignement supérieur.

Notre jeu de données en quelques chiffres

- Jeu de données obtenues grâce à des élèves de l'Institut polytechnique de Portalegre (Portugal) entre 2009 et 2019
- 37 colonnes et 4424 observations
- Aucune donnée manquante
- De nombreuses variables catégorielles (étudiants étranger, boursier, niveau d'études des parents...)
- Une colonne indiquant si l'élève est resté inscrit, s'il a été diplômé ou s'il a abandonné

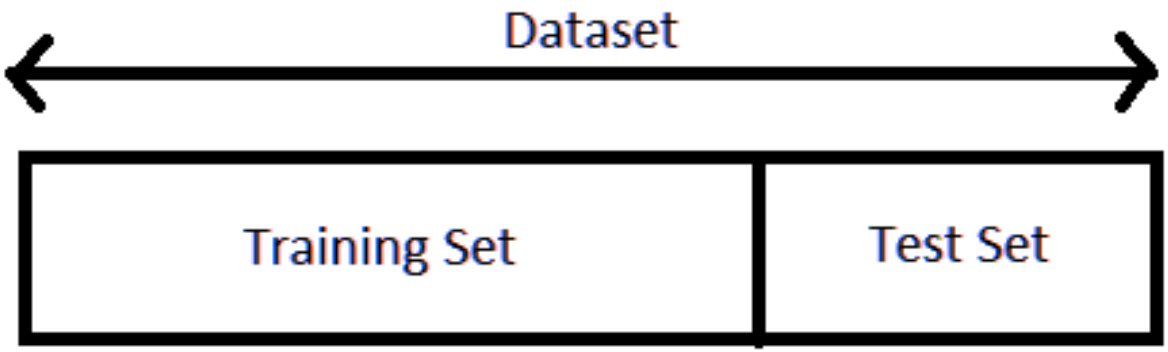
Variable cible



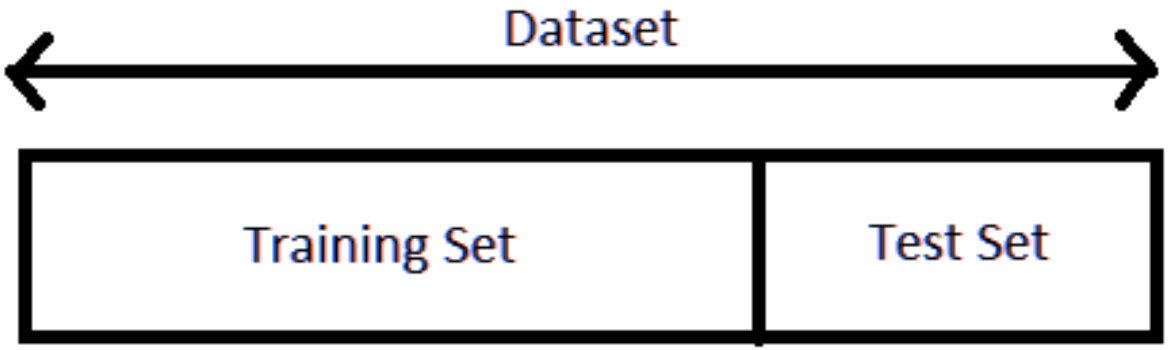
Transformation de la variable cible

- 1 s'il l'élève a été diplômé
- 0 sinon

Découpe du jeu de données

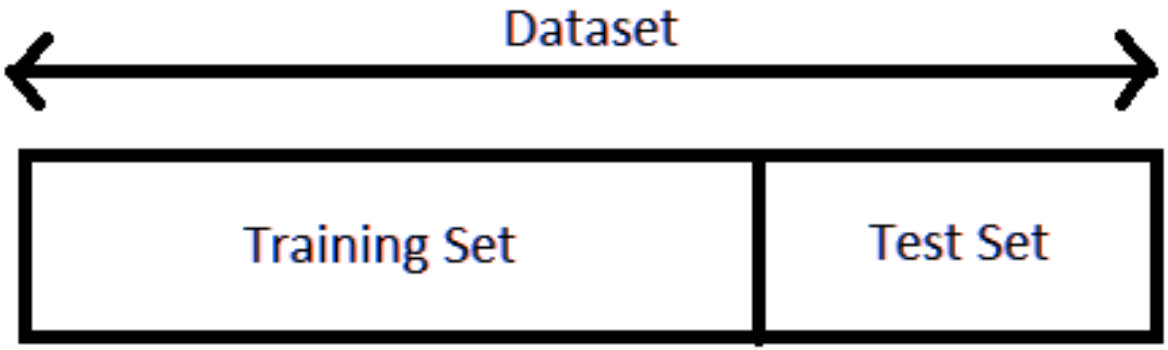


Découpe du jeu de données



Problème : Comment faire pour les catégories qui n'apparaissent qu'une seule fois ?

Découpe du jeu de données



Problème : Comment faire pour les catégories qui n'apparaissent qu'une seule fois ?

- Supprimer les observations appartenant à une catégorie "unique"
- Forcer toutes les autres catégories à apparaître au moins une fois dans chaque partie

Pour arriver à prédire du mieux possible notre variable cible Y, il est possible d'effectuer une **régression linéaire** dépendant des différentes variables de notre problème.

Modèle de régression linéaire

$$Y = \beta_1 + \sum_{j=2}^{p+1} \beta_j X_j + \varepsilon$$

Avec :

- Y la valeur cible.
- $X = (X_1, \ldots, X^{p+1})$ les valeurs des différentes variables (avec $p = 37$ ici).
- Le "Data Set" : $D_n = \{(x_i, y_i) \mid i \in [\![1, n]\!],\ y_i \in \mathbb{R}, x_i \in \mathbb{R}^p\}$ (avec $n = 4424$ ici).
- ε correspond à l'erreur résiduelle.

Dans notre étude, la variable cible Y est une variable **catégorielle**, transformée en variable **binaire**. Il est donc nécessaire de passer par une **régression logistique**

Modèle de régression logistique

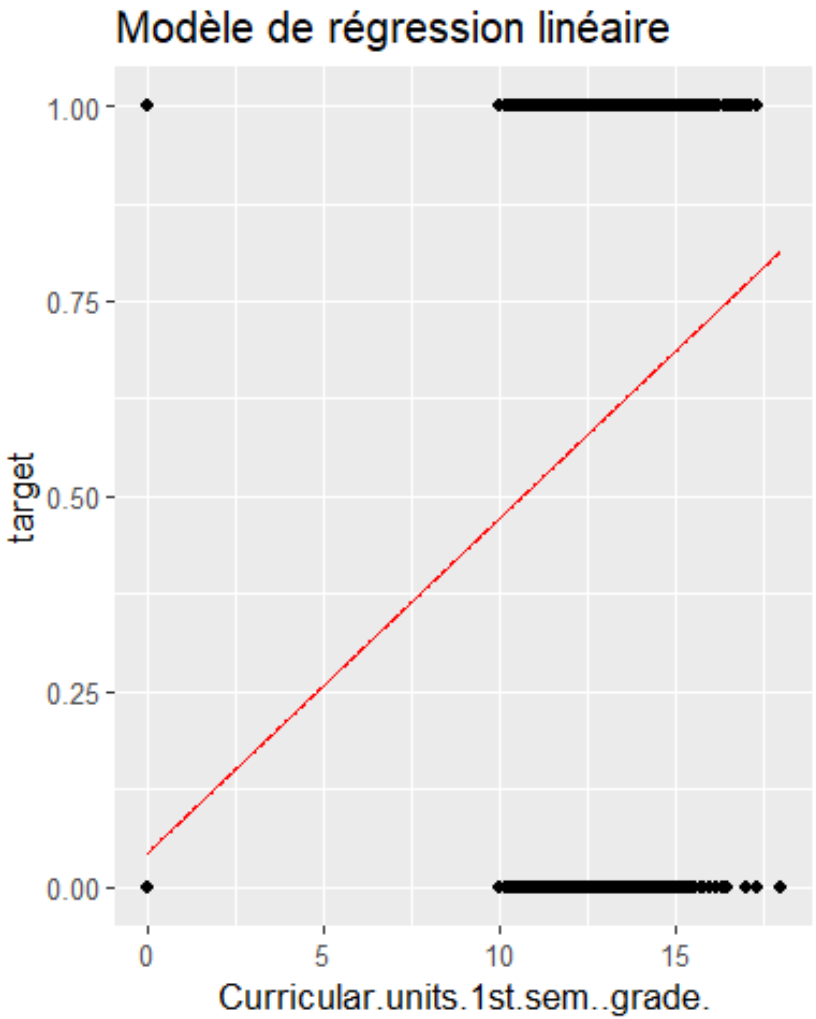
Dans ce modèle, on part du principe que les observations y_i sont des réalisations de variables aléatoires Y_i indépendantes, de loi de Bernoulli de paramètre $p_\beta(x_i)$ tel que :

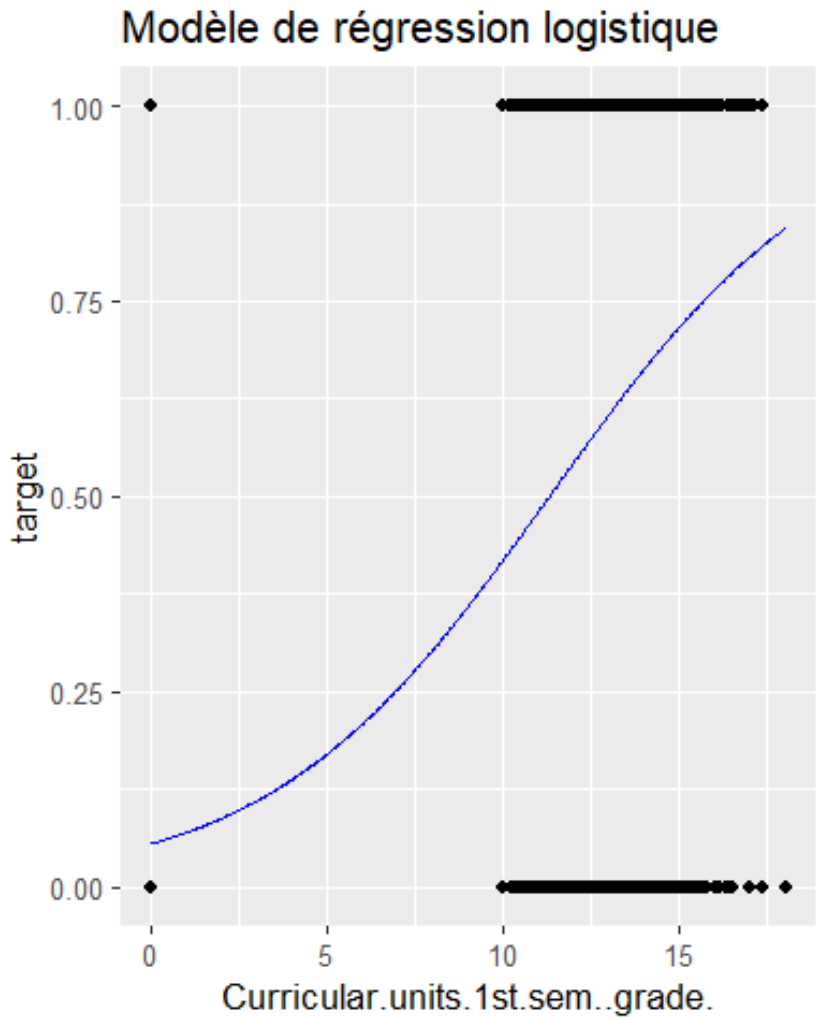
$$\text{logit}\, p_\beta(x_i) = \ln\left(\frac{p_\beta(x_i)}{1-p_\beta(x_i)}\right) = \beta_1 x_{i,1} + \cdots + \beta_p x_{i,p} = \beta^T x_i$$

Où la fonction de transfert est donnée par **la fonction sigmoïde** :

$$p_\beta(x_i) = \frac{e^{\beta^T x_i}}{1 + e^{\beta^T x_i}}$$

Un exemple pertinent





Explication brève des modèles simples

L'idée maîtresse : **mettre de côté les variables non-significatives** pour notre régression logistique.

Réduction du nombre de variables p

On définit pour cela l'ensemble $\mathcal{M} \subset [\![1, p]\!]$ tel que :

$$\forall i \in [\![1, n]\!], \quad \text{logit}\left(p_\beta(x_i)\right) = (X_i)_\mathcal{M}\beta$$
$$\implies \quad \text{logit}(P) = X\beta$$

Avec :

- $X_\mathcal{M} = [x_{i,j_k}]_{i \in [\![1,n]\!], j_k \in \mathcal{M}}$
- $\beta = [\beta_1, \beta_2, \ldots, \beta_k]^T$ (on prend la transposée !)

3 algorithmes "gloutons" vus en cours

- "**Forward selection**" : On commence avec aucune variable et ajoute la variable la plus significative à chaque étape.
- "**Backward elimination**" : On commence avec toutes les variables et on élimine la moins significative à chaque étape.
- "**Stepwise regression**" : On commence avec aucune variable, et on effectue une sorte d'entre d'eux des deux méthodes précédentes.

Ces méthodes de sélection de variables sont dites **incrémentales** car elles ajoutent ou retirent une variable à la fois et réévaluent le modèle à chaque étape.

3 algorithmes "gloutons" vus en cours

- "**Forward selection**" : On commence avec aucune variable et ajoute la variable la plus significative à chaque étape.
- "**Backward elimination**" : On commence avec toutes les variables et on élimine la moins significative à chaque étape.
- "**Stepwise regression**" : On commence avec aucune variable, et on effectue une sorte d'entre d'eux des deux méthodes précédentes.

Ces méthodes de sélection de variables sont dites **incrémentales** car elles ajoutent ou retirent une variable à la fois et réévaluent le modèle à chaque étape.

Attention au temps de calcul !

Lorsque p est grand, elles peuvent être vraiment gourmandes en temps de calcul.

Premiers résultats

Forward	**Backward**	**Stepwise**
$\begin{pmatrix} 440 & 90 \\ 60 & 442 \end{pmatrix}$	$\begin{pmatrix} 437 & 93 \\ 60 & 442 \end{pmatrix}$	$\begin{pmatrix} 441 & 89 \\ 57 & 445 \end{pmatrix}$
0.8546512	0.8517442	0.8585271
35.14 secondes	265.34 secondes	28.89 secondes

Table 1 – Matrices de confusion des 3 modèles simples

Explication brève des modèles de pénalisation

La régression Ridge (ℓ_2) et la régression Lasso (ℓ_1) sont des **méthodes de régression pénalisée** qui visent à résoudre ces problèmes d'une manière différente.

En quelque sorte, elles construisent un modèle en optimisant tous les coefficients simultanément avec la contrainte de la pénalité λ.

Explication brève des modèles de pénalisation

La régression Ridge (ℓ_2) et la régression Lasso (ℓ_1) sont des **méthodes de régression pénalisée** qui visent à résoudre ces problèmes d'une manière différente.

En quelque sorte, elles construisent un modèle en optimisant tous les coefficients simultanément avec la contrainte de la pénalité λ.

Deux méthodes de pénalisation

- **La régression Ridge** : Pénalisation en norme ℓ_2.
- **La régression Lasso** : Pénalisation en norme ℓ_1.

Régression Ridge

On souhaite ici minimiser la fonction Φ donnée par :

$$\Phi(\beta) = \|Y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

$$\Longrightarrow \quad \Phi(\beta) = (Y - X\beta)^T (Y - X\beta) + \lambda \sum_{j=1}^{p} \beta_j^2 \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

Il est donc important d'optimiser notre hyper-paramètre λ.

Régression Ridge

On souhaite ici minimiser la fonction Φ donnée par :

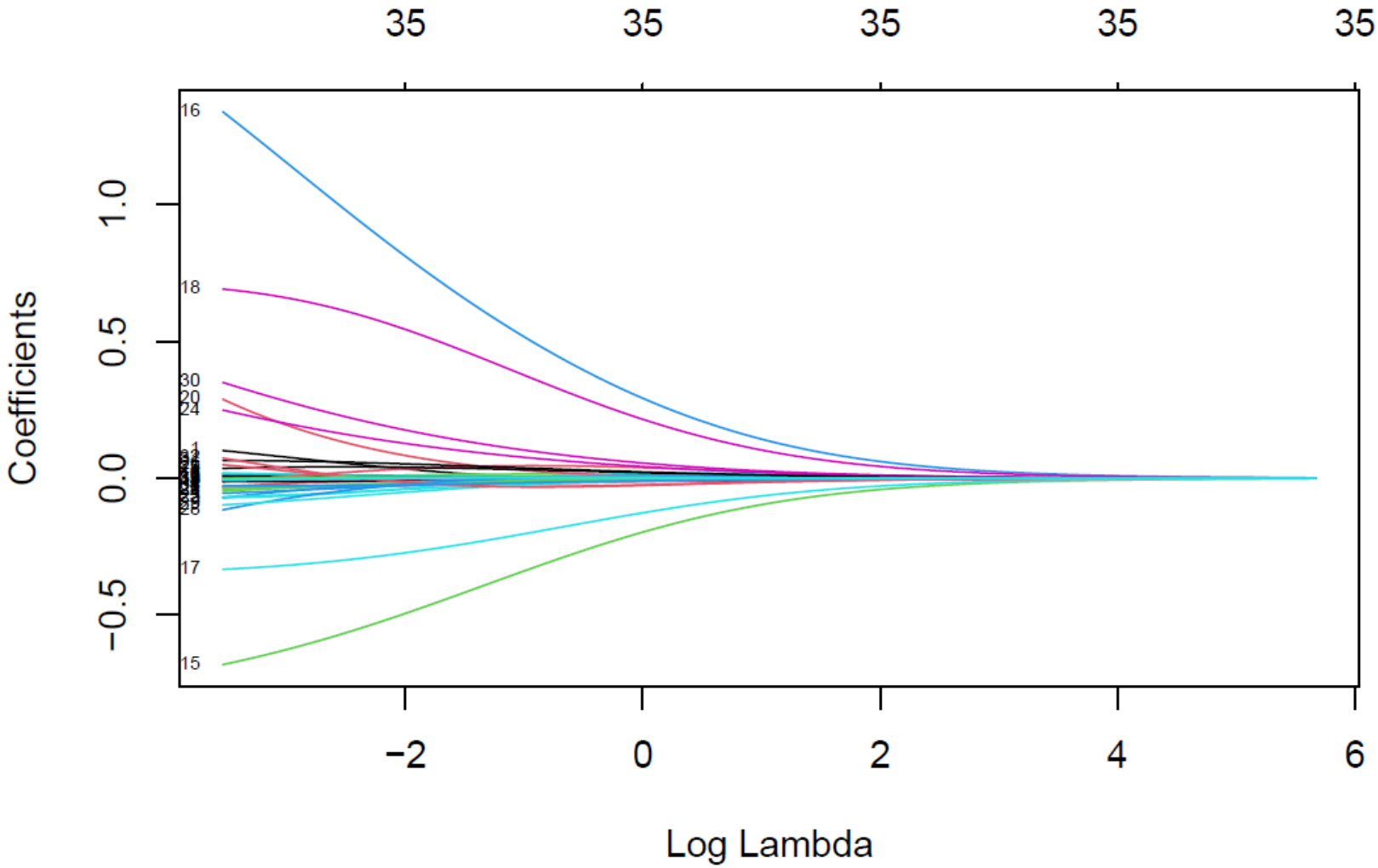
$$\Phi(\beta) = \|Y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

$$\Longrightarrow \quad \Phi(\beta) = (Y - X\beta)^T (Y - X\beta) + \lambda \sum_{j=1}^{p} \beta_j^2 \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

Il est donc important d'optimiser notre hyper-paramètre λ.

- Validation croisée pour optimiser λ.
- Choisir le paramètre parmi les différentes possibilités :
  - λ **minimum** (la valeur de λ qui donne le minimum d'erreur de validation croisée).
  - λ **1se** (la valeur de régularisation lambda qui est à un d'écart-type du modèle, avec l'erreur de validation croisée la plus faible).

Résultat graphique



Régression Lasso

On souhaite ici minimiser la fonction Φ donnée par :

$$\Phi(\beta) = \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1 \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

$$\Longrightarrow \quad \Phi(\beta) = (Y - X\beta)^T (Y - X\beta) + \lambda \sum_{j=1}^{p} |\beta_j| \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

Là aussi, il est important d'optimiser notre hyper-paramètre λ.

Régression Lasso

On souhaite ici minimiser la fonction Φ donnée par :

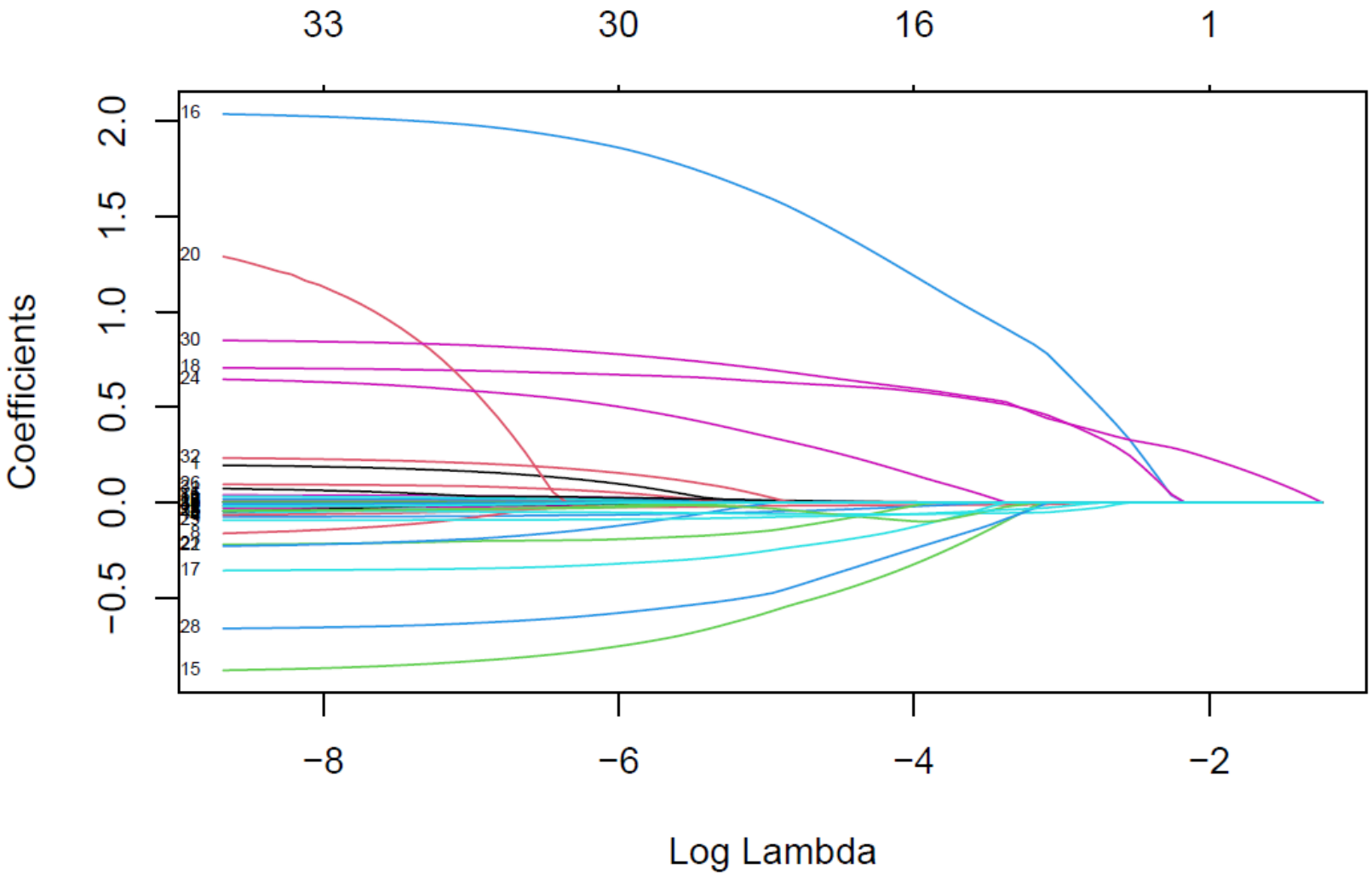
$$\Phi(\beta) = \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1 \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

$$\Longrightarrow \quad \Phi(\beta) = (Y - X\beta)^T (Y - X\beta) + \lambda \sum_{j=1}^{p} |\beta_j| \qquad (\text{avec } \lambda \in \mathbb{R}_+^*)$$

Là aussi, il est important d'optimiser notre hyper-paramètre λ.

- Validation croisée pour optimiser λ.
- Choisir le paramètre parmi les différentes possibilités :
  - λ **minimum**.
  - λ **1se**.

Résultat graphique

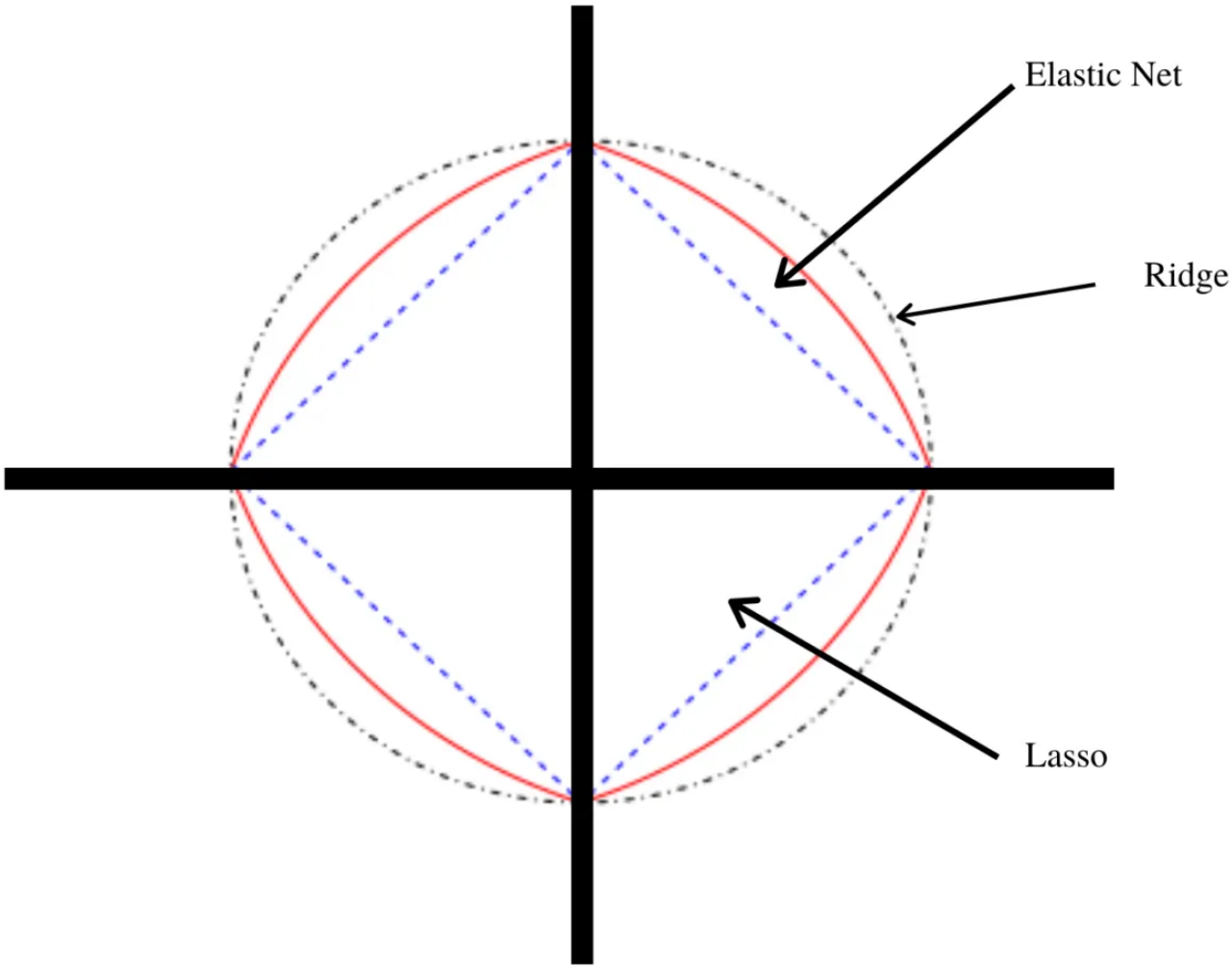


Premiers résultats

Ridge $\lambda_{\mathbf{min}}$	**Ridge** $\lambda_{\mathbf{1se}}$	**Lasso** $\lambda_{\mathbf{min}}$	**Lasso** $\lambda_{\mathbf{1se}}$
$\begin{pmatrix} 404 & 114 \\ 46 & 468 \end{pmatrix}$	$\begin{pmatrix} 403 & 115 \\ 46 & 468 \end{pmatrix}$	$\begin{pmatrix} 418 & 100 \\ 54 & 460 \end{pmatrix}$	$\begin{pmatrix} 415 & 103 \\ 46 & 468 \end{pmatrix}$
0.8449612	0.8439922	0.85077519	0.8556202
0.20 secondes		0.17 secondes	

Table 2 – Comparaison des modèles de régression Ridge et Lasso

Explication visuelle derrière Elastic Net



Régression "Elastic Net"

Mathématiquement, en reprenant l'idée d'une régression Ridge et Lasso, on souhaite ici minimiser la fonction Φ donnée par :

$$\Phi(\beta) = \|Y - X\beta\|_2^2 + \lambda \left(\frac{1-\alpha}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 \right)$$

(avec $\lambda \in \mathbb{R}_+^*$, $\alpha \in [0,1]$)

$$\Longrightarrow \quad \Phi(\beta) = (Y - X\beta)^T (Y - X\beta) + \lambda \left(\frac{1-\alpha}{2} \sum_{j=1}^{p} \beta_j^2 + \alpha \sum_{j=1}^{p} |\beta_j| \right)$$

Régression "Elastic Net"

Mathématiquement, en reprenant l'idée d'une régression Ridge et Lasso, on souhaite ici minimiser la fonction Φ donnée par :

$$\Phi(\beta) = \|Y - X\beta\|_2^2 + \lambda \left(\frac{1-\alpha}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 \right)$$

(avec $\lambda \in \mathbb{R}_+^*$, $\alpha \in [0,1]$)

$$\Longrightarrow \quad \Phi(\beta) = (Y - X\beta)^T (Y - X\beta) + \lambda \left(\frac{1-\alpha}{2} \sum_{j=1}^{p} \beta_j^2 + \alpha \sum_{j=1}^{p} |\beta_j| \right)$$

$\longrightarrow$ Nouvel hyper-paramètre : α

Optimisation du Paramètre α via la Validation Croisée

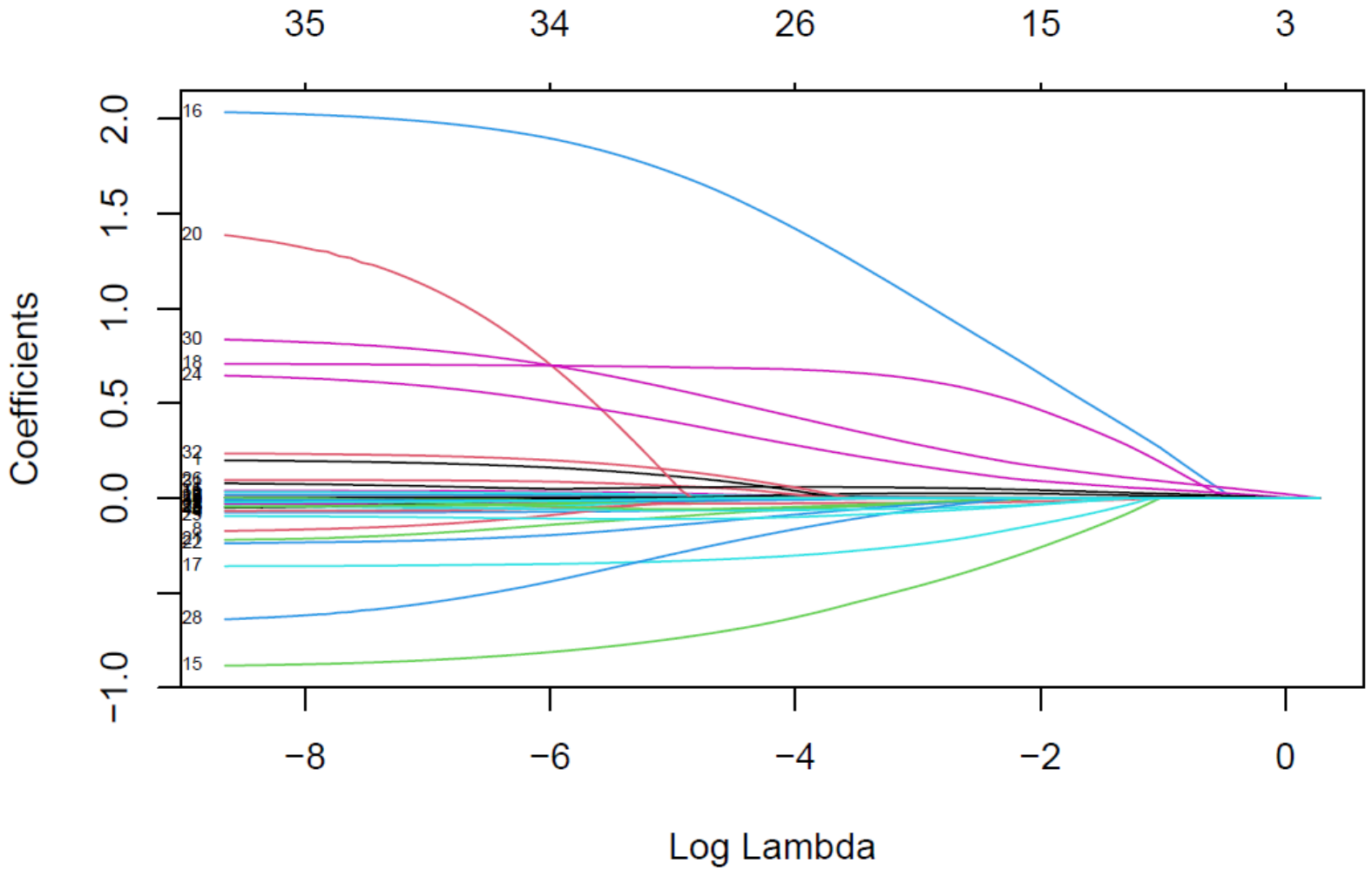
Objectif : Sélectionner la valeur d'alpha qui maximise la précision du modèle.

Pour cela, on a effectué **une validation croisée à K blocs** (*K-fold validation*). [cf.Annexe 1]

Avantages de la Validation Croisée

- Sélectionne la valeur d'alpha qui a donné l'erreur de validation croisée minimale.
- Aide à équilibrer le biais et la variance, évitant ainsi le surajustement.

Résultat graphique



Régression "Elastic Net" avec $\alpha = 0.22$

Premiers résultats

Elastic Net λ_{min}	**Elastic Net λ_{1se}**
$\begin{pmatrix} 419 & 91 \\ 52 & 470 \end{pmatrix}$	$\begin{pmatrix} 415 & 95 \\ 49 & 473 \end{pmatrix}$
0.8614341	0.8604651
0.35 secondes	

Table 3 – Résultats du modèle Elastic Net

Résultats principaux de nos modèles : modèles simples

Forward	**Backward**	**Stepwise**
0.8546512	0.8517442	0.8585271
35.14 secondes	265.34 secondes	28.89 secondes
2nd.sem..approved	2nd.sem..approved	2nd.sem..approved
1st.sem..approved	1st.sem..approved	1st.sem..approved
Tuition.fees.up.to.date	Tuition.fees.up.to.date	Tuition.fees.up.to.date
Scholarship.holder	Scholarship.holder	Scholarship.holder
2nd.sem..enrolled	2nd.sem..enrolled	2nd.sem..enrolled
Debtor	2nd.sem..evaluations	2nd.sem..evaluations
2nd.sem..evaluations	2nd.sem..without.evaluations	2nd.sem..without.evaluations
2nd.sem..grade	Debtor	Debtor
2nd.sem..without.evaluations	1st.sem..enrolled	1st.sem..enrolled

Table 4 – Comparaison des modèles simples de régression

Résultats principaux de nos modèles : modèles intermédiaires

Ridge λ min	**Ridge λ 1se**	**Lasso λ min**	**Lasso λ 1se**
0.8449612	0.8439922	0.85077519	0.8556202
0.20 secondes		0.17 secondes	
Tuition.fees.up.to.date	Tuition.fees.up.to.date	Tuition.fees.up.to.date	Tuition.fees.up.to.date
Scholarship.holder	Scholarship.holder	International	2nd.sem..approved.
Debtor	Debtor	Debtor	Scholarship.holder
International	International	2nd.sem..approved.	Debtor
2nd.sem..approved.	Gender	Scholarship.holder	2nd.sem..enrolled.
Gender	2nd.sem..approved.	2nd.sem..enrolled.	1st.sem..approved.
1st.sem..approved.	1st.sem..approved.	1st.sem..approved.	Gender
2nd.sem..evaluations.	2nd.sem..evaluations.	Gender	International
2nd.sem..enrolled.	2nd.sem..grade.	1st.sem..enrolled.	1st.sem..credited.

Table 5 – Comparaison des modèles de régression Ridge et Lasso

Résultats principaux de nos modèles : modèles simples

Elastic Net λ_{min}	**Elastic Net λ_{1se}**
0.8614341	0.8604651
0.35 secondes	
Tuition.fees.up.to.date	Tuition.fees.up.to.date
International	International
Debtor	Debtor
Scholarship.holder	Scholarship.holder
2nd.sem..approved.	2nd.sem..approved.
1st.sem..approved.	1st.sem..approved.
Gender	Gender
2nd.sem..enrolled.	2nd.sem..enrolled.
2nd.sem..without.evaluations.	2nd.sem..without.evaluations.

Table 6 – Analyse du modèle Elastic Net

Les principaux facteurs

Les variables "évidentes"

Certains variables ressortent de **tous** les modèles.

- `Tuition.fees.up.to.date`
- `Debtor`
- `2nd.sem..approved`
- `1st.sem..approved`

Les principaux facteurs

Les variables "évidentes"

Certains variables ressortent de **tous** les modèles.

- `Tuition.fees.up.to.date`
- `Debtor`
- `2nd.sem..approved`
- `1st.sem..approved`

Des variables plus surprenantes

Certains variables sont significatives, mais cela dépend clairement des modèles.

- `Scholarship.holder`
- `International`
- `Gender`

Autres résultats

Les cours importants

Certains cours semblent également sortir un peu du lot dans notre étude. La corrélation entre la réussite dans ces cours et la réussite globale de l'année pour les élèves est forte pour ces cours par exemple :

- Le cours de "Service Social" (et également le cours de soutien le soir).
- Le cours d'éducation de base.
- Le cours d'agronomie.
- Le cours de conception d'animation et multimédia.

Les points forts et les points faibles de nos modèles

Les points forts de nos modèles

- Une grande variété de modèles
- Des résultats cohérents, avec des variables significatives plutôt logiques.
- Des précisions très satisfaisantes.

Quelles perspectives ?

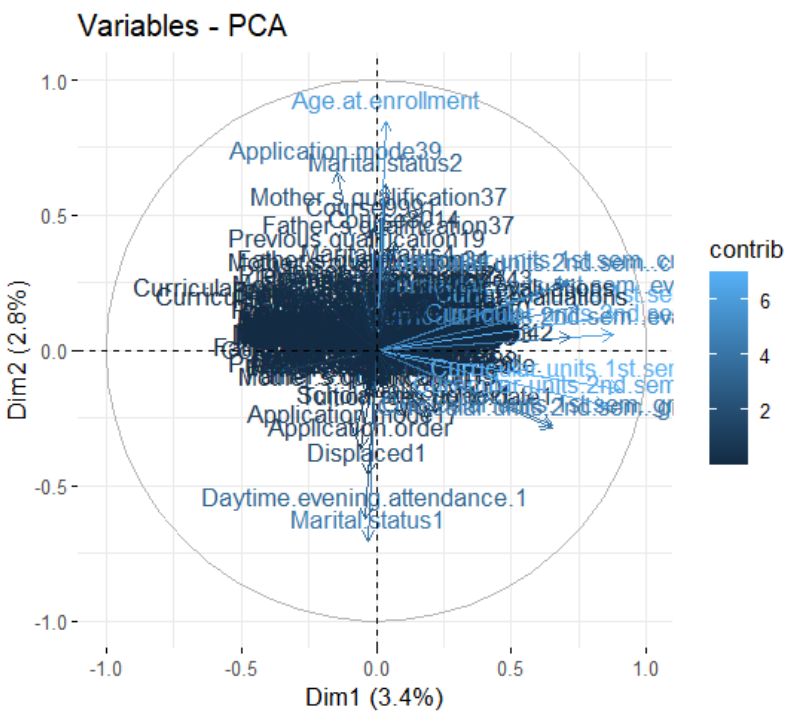
- Supprimer des variables trop "évidentes" pour continuer de chercher des nouveaux modèles intéressants.
- Une analyse en composantes principales (ACP) pourrait nous donner de nouvelles informations pertinentes.

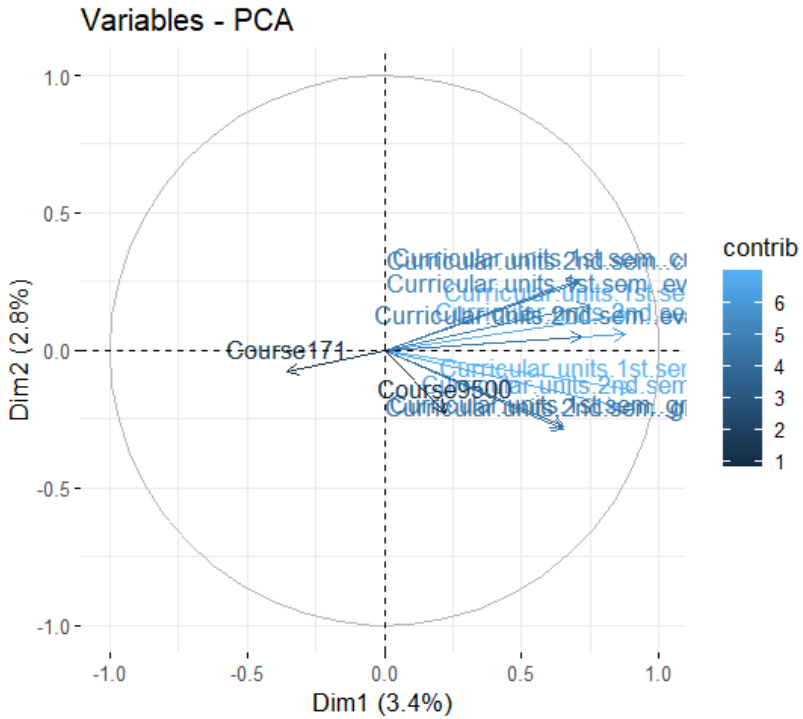
Annexe 1

Processus d'Optimisation K-fold

1. Diviser le jeu de données en plusieurs sous-ensembles (par exemple, en k sous-ensembles).
2. Pour chaque valeur d'alpha :
   - Former le modèle sur k-1 sous-ensembles.
   - Évaluer sa performance sur le sous-ensemble restant.
   - Calculer l'erreur moyenne sur les k itérations.
3. Sélectionner la valeur d'alpha qui minimise l'erreur de validation croisée.

Annexe 2


Variables - PCA
Age.at.enrollment
Application.mode39
Marital.status2
Mother.s.qualification37
Course9003
Father.s.qualification37
Previous.qualification19
Application.mode
Application.order
Displaced1
Daytime.evening.attendance.1
Marital.status1
contrib
6
4
2
Dim1 (3.4%)
Dim2 (2.8%)
1.0
0.5
0.0
-0.5
-1.0


Variables - PCA
Course171
Course9500
contrib
6
5
4
3
2
1
Dim1 (3.4%)
Dim2 (2.8%)
1.0
0.5
0.0
-0.5
-1.0