STA03 - APPRENTISSAGE STATISTIQUE

Le raisin

Mai 2025

Josselin De Féligonde - Gabriel Vignon

ENSTA | INSTITUT POLYTECHNIQUE DE PARIS

TABLE DES MATIÈRES

INTRODUCTION

1
ANALYSE NON SUPERVISÉE

1.1 - Analyse descriptive

On commence par mener une analyse descriptive de notre Dataset. La fonction `summary` nous donne déjà un certain nombre d'informations pour le décrypter :

```
 description
      Area           MajorAxisLength MinorAxisLength  Eccentricity      ConvexArea
Extent          Perimeter
 Min.   : 25387   Min.   :225.6   Min.   :143.7   Min.   :0.3487   Min.   : 26139
Min.   :0.3799   Min.   : 619.1
 [...]
 Mean   : 87804   Mean   :430.9   Mean   :254.5   Mean   :0.7815   Mean   : 91186
Mean   :0.6995   Mean   :1165.9
 [...]
 Max.   :235047   Max.   :997.3   Max.   :492.3   Max.   :0.9621   Max.   :278217
Max.   :0.8355   Max.   :2697.8
    Class
 Length:900
 Class :character
 [...]
```

On constate l'existence de 7 variables quantitatives (nos *features*) et d'une variable qualitative (notre *target*), Class. Class prend deux valeurs ; « Kecimen » ou « Besni » (les deux types de raisin étudiés).

On constate par ailleurs que les individus du jeu sont parfaitement répartis entre les deux classes :

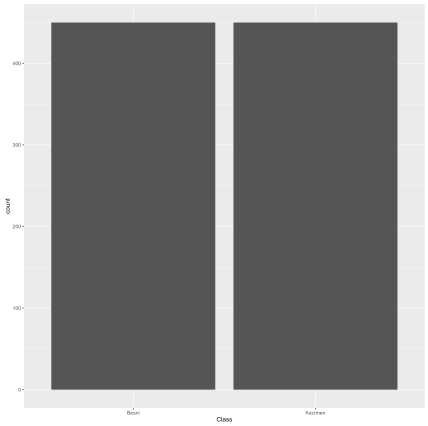

Fig. 1. – Equilibre entre les deux classes

Il y a 450 individus dans chaque classe, on n'a donc pas à se soucier d'éventuels problèmes lors de l'apprentissage dues à un déséquilibre des classes.

Pour étudier la répartition des données et les corrélations entre les variables, on peut enfin s'intéresser aux nuages de points et au graphique des corrélations :

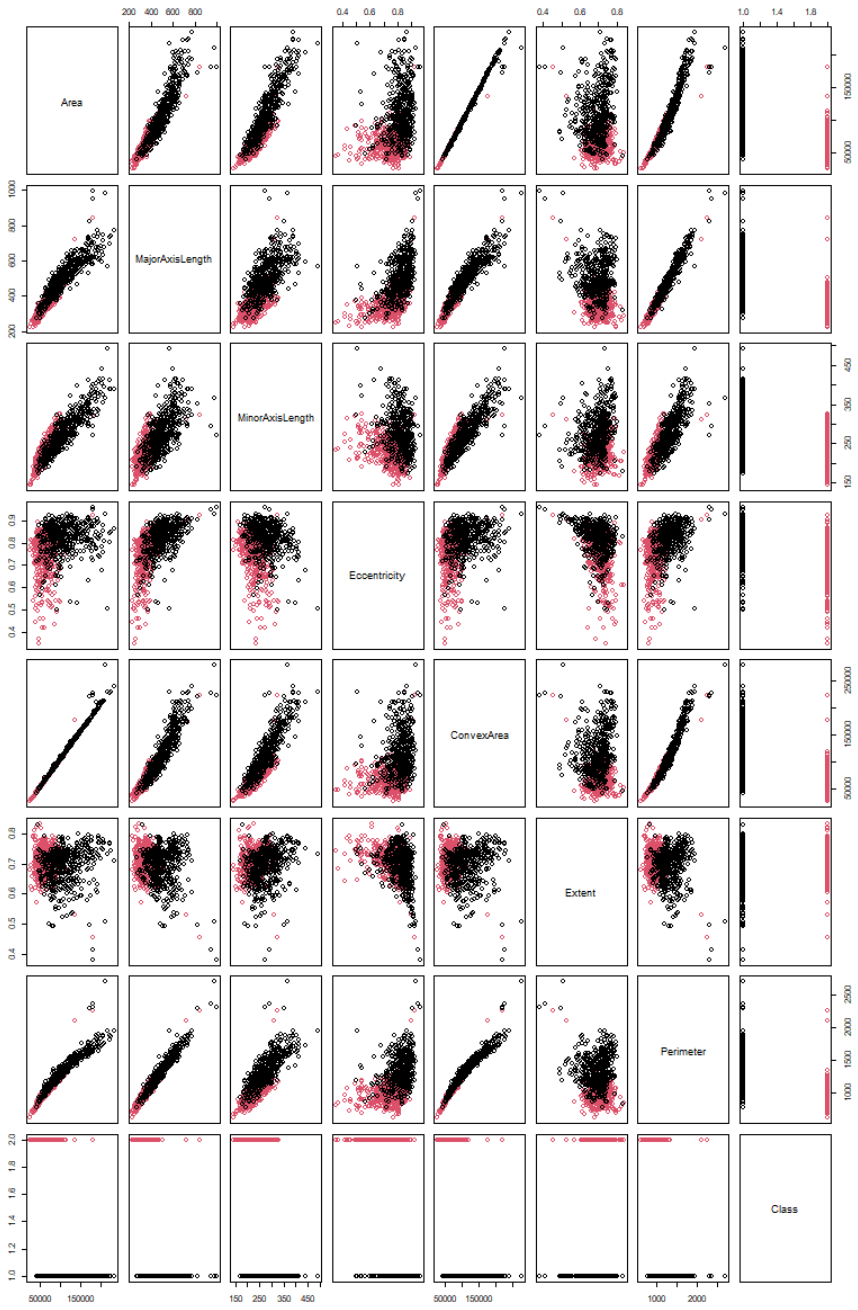

Fig. 2. – Nuages des individus

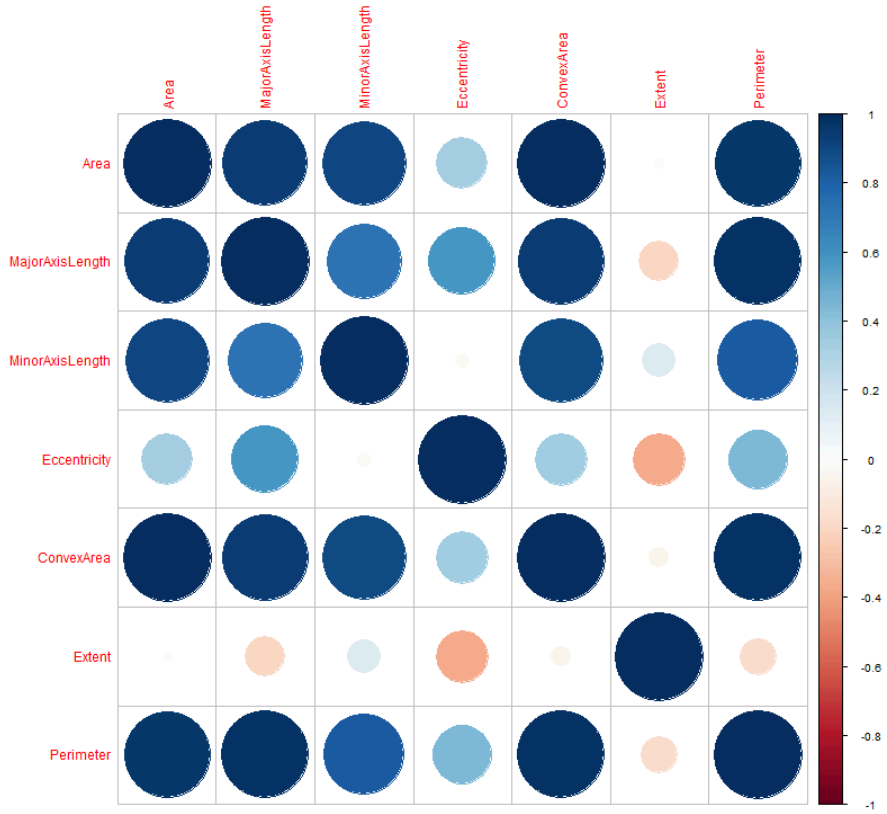

Fig. 3. – Graphique des corrélations

On observe des corrélations positives particulièrement importantes entre :
- Area et ConvexArea,
- Area et Perimeter,
- MajorAxisLength et Perimeter,
- MajorAxisLength et ConvexArea,
- ConvexArea et Perimeter.

Ces corrélations semblent cohérentes si on redonne leurs significations à nos variables (Un raisin au périmètre important aura problablement une aire importante, etc.).

1.2 - Réduction de dimension par ACP

On va effectuer une analyse en composantes principales afin d'opérer une réduction de dimension. En voici les grandes étapes :

- **Centrer et réduire le nuage des individus.**
- **Diagonaliser la matrice des corrélations et ne garder que les plus grandes valeurs propres. Les vecteurs propres associés correspondent aux axes principaux que l'on souhaite retenir.**

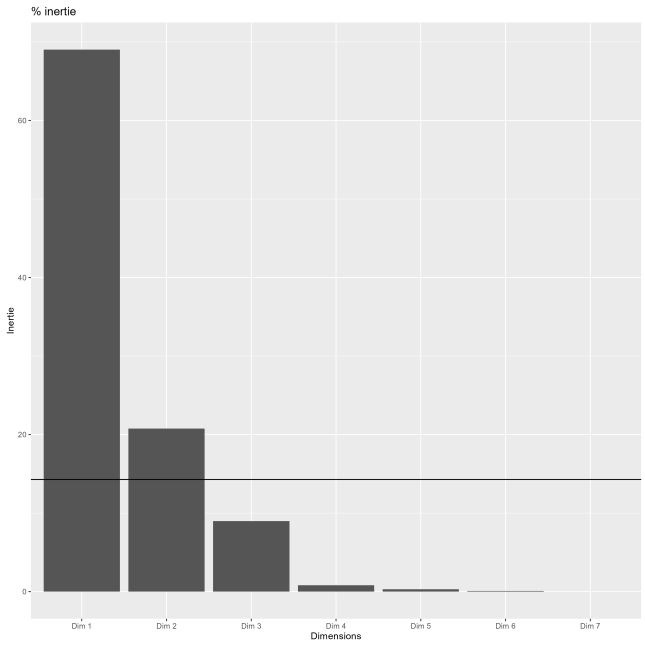

Fig. 4. – Valeurs propres de la matrice des corrélations

Il semble naturel à ce stade de retenir deux composantes principales. On peut justifier ce choix en notant que ces deux composantes suffisent à expliquer environ 90% de l'inertie totale (ou autrement par la méthode du coude).

- **Projeter les individus et les variables dans l'espace décorrélé formé des axes principaux que l'on souhaite retenir.**

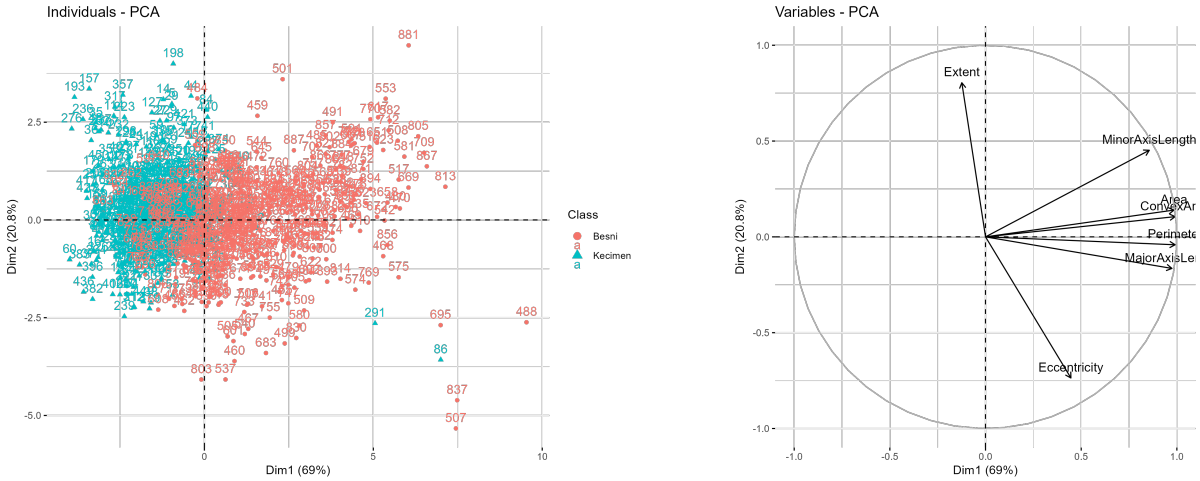

Fig. 5. – Premier plan de l'ACP : individus et variables

Les variables liées à la taille du raisin (`Area`, `ConvexArea`, `Perimeter`, `MajorAxisLength`) sont groupées et fortement corrélées entre elles sur l'axe 1. Les variables `Extent` et `Eccentricity` sont opposées sur l'axe 2. Ce second axe semble être lié à la forme du raisin. `MinorAxisLength` est entre les deux groupes, elle apporte probablement des informations à la fois sur la taille et la forme. Il semblerait que la taille du raisin soit la principale différence entre les deux types.

- **Analyser les résultats obtenus.**
  - *cos2 (i.e. la qualité de représentation) :*

```
cos2_variables_pca
                     Dim.1       Dim.2        Dim.3        Dim.4        Dim.5
Area            0.97109063 0.019587803 1.889866e-05 0.0007018902 8.142754e-03
MajorAxisLength 0.94935877 0.027110793 6.353558e-03 0.0139349408 1.672380e-04
MinorAxisLength 0.73266049 0.204270120 3.501504e-02 0.0244444415 3.225444e-03
Eccentricity    0.19907684 0.542191863 2.482627e-01 0.0103267491 1.230124e-04
ConvexArea      0.98262357 0.011155576 8.451798e-04 0.0001770053 3.358172e-03
Extent          0.01535408 0.647175333 3.367208e-01 0.0006757833 7.050454e-05
Perimeter       0.98212422 0.001696952 1.233368e-03 0.0065637022 6.721143e-03
```

Les variables `Area`, `ConvexArea`, `Perimeter` et `MajorAxisLength` sont extrêmement bien représentées sur la première dimension. Les variables `Extent` et `Eccentricity` sont bien représentées sur la seconde dimension, alors qu'elles ne l'étaient pas bien sur la première. La majorité des variables ont une très bonne qualité de représentation dans le premier plan, donc le choix que l'on a fait semble pertinent

- *contributions des variables aux axes :*

```
contributions_variables_pca
                     Dim.1      Dim.2        Dim.3      Dim.4      Dim.5
Area            20.0958741  1.3479190  0.003007188  1.2351892 37.3379225
MajorAxisLength 19.6461521  1.8656075  1.010989518 24.5227635  0.7668561
MinorAxisLength 15.1617701 14.0566849  5.571655160 43.0174242 14.7900045
Eccentricity     4.1197217 37.3104993 39.503997598 18.1730537  0.5640629
ConvexArea      20.3345382  0.7676621  0.134486513  0.3114945 15.3986199
Extent           0.3177392 44.5348529 53.579608236  1.1892461  0.3232927
Perimeter       20.3242045  0.1167744  0.196255786 11.5508289 30.8192413
```

Pour le premier axe, les contributions sont très équilibrées entre `Area` , `MajorAxisLength`, `ConvexArea` et `Perimeter`. Cet axe représente une combinaison équilibrée des variables de taille. Pour le second axe, les variables `Extent` et `Eccentricity` sont les principales contributrices.

1.3 - Classification ascendante hiérarchique

On effectue une classification ascendante hiérarchique. On utilise la méthode complète et la distance euclidienne. Il est nécesseraire de centrer et réduire les variables pour ne pas fausser les distances qui entrent en jeu. Sans cette étape, les variables à forte variance auraient une importance démesurée dans l'analyse. On obtient ainsi un dendrogramme (figure 6) :

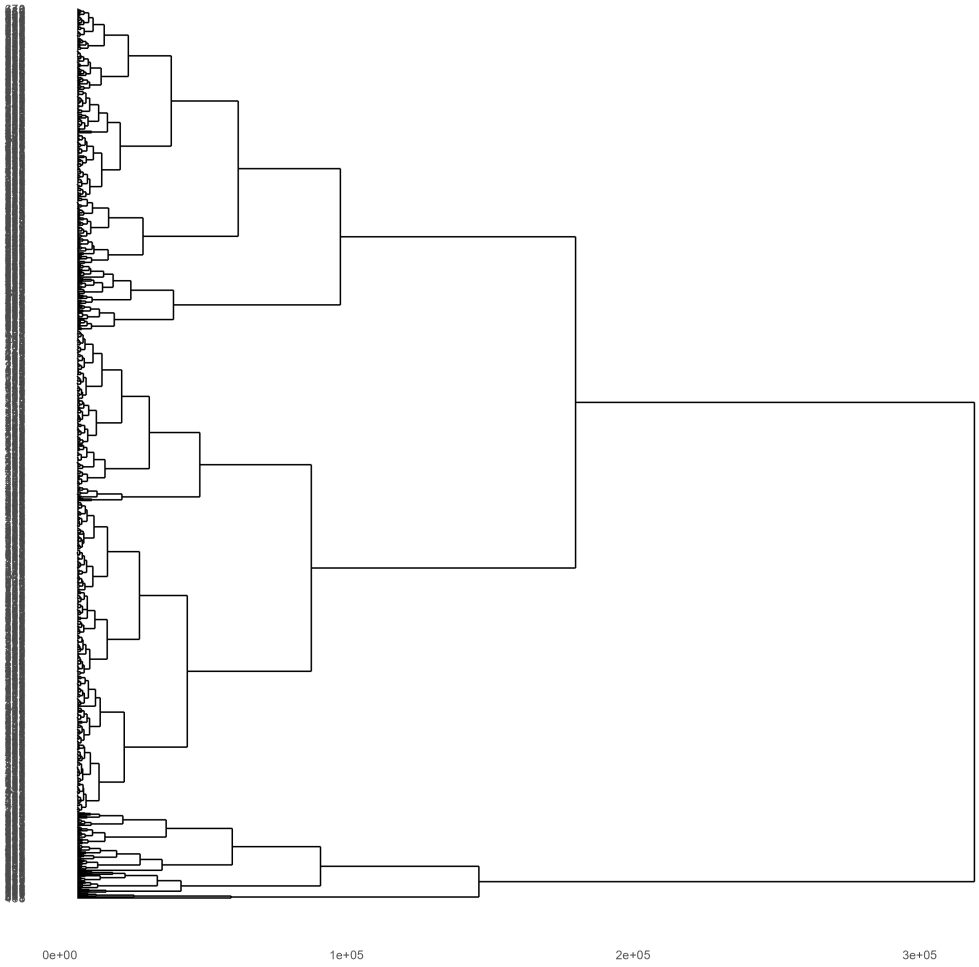

Fig. 6. – Dendrogramme

Au vu des hauteurs des sauts, on est tenté de garder 2 ou 3 classes. Ayant connaissance du problème, on n'en garde que deux. La classification en deux groupes mène à une erreur de **37%**, assez importante.

Pour réduire, cette erreur, il serait intéressante de faire la classification à partir des composantes principales de la section précédente. Avec une classification en deux classes, on obtient maintenant une erreur de **34%**, un peu plus faible. On teste différents nombre de classes (figure 7). La classification en 3 groupes mène à l'erreur la plus faible, de **25%**.

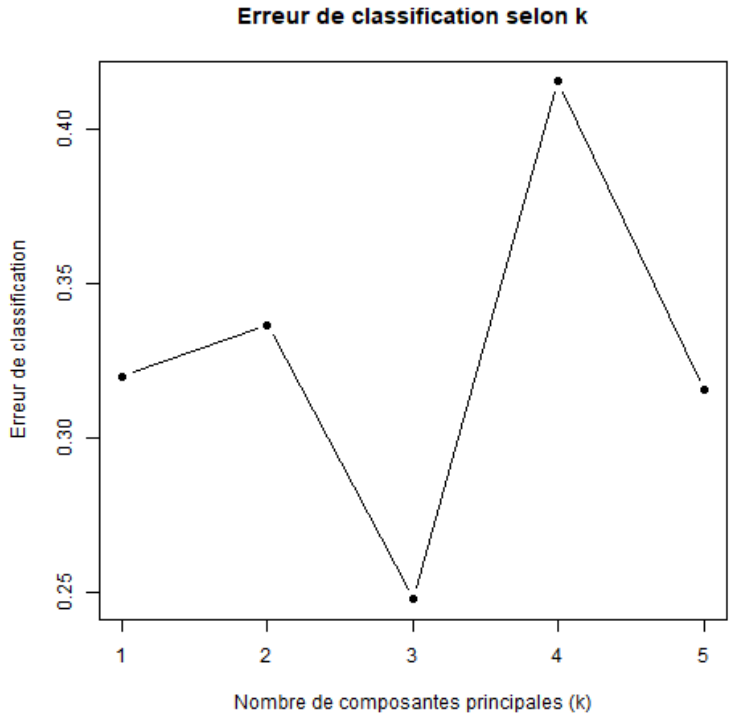

Fig. 7. – Erreurs en fonction du nombre de classes retenues

Ce résultat n'est pas sans biais. La performance avec k = 3 pourrait être propre à ce dataset et ne pas se généraliser à d'autres échantillons, on teste ici notre choix sur le jeu d'entraînement. Pour obtenir une estimation sans biais, il faudrait faire une validation croisée.

2
CHOISIR UN MODÈLE

2.1 - Régression logistique

La régression logistique est un modèle linéaire généralisé adapté à des réponses binaires (Bernoulli ou binomiales). Dans notre cas (où la fonction de lien est par défaut logit), si on prend $p(X) = \mathbb{P}(y = \text{TRUE} \mid \text{X})$:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = X\theta$$

ou, sous la forme inversée :

$$p(X) = \frac{1}{1+e^{-X\theta}}$$

Centrer et réduire le jeu de données peut être utile pour préparer les étapes suivantes, par exemple une régularisation.

2.2 - ACP

- On réalise l'ACP sur l'échantillon d'apprentissage uniquement.
- On projette ensuite les nouvelles observations (ici le jeu de test) dans cet espace factoriel. Voici le résultat :

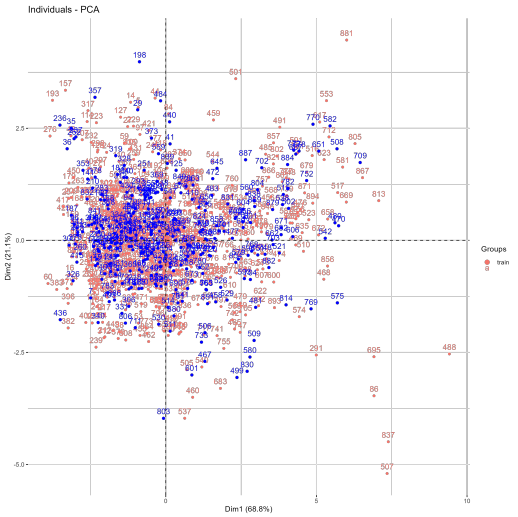

Fig. 8. – Projection des individus des testeurs (en bleu) dans le premier plan de l'ACP

2.3 - Estimation de différents modèles

On estime les modèles logistiques suivant :

- **le modèle complet**
- **le modèle ne prenant en compte que les de deux premières composantes principales**
- **le modèle obtenu par sélection de variables avec critères AIC**

On choisit la pénalité λ par validation croisée.

- **le modèle obtenu par régression lasso**

Ainsi que :

- **un SVM linéaire**
- **un SVM avec noyau polynomial**

Il y a un hyperparamètre ici, le degré du polynôme. Plus il sera grand, plus le modèle sera flexible mais plus il y aura un risque de sur-apprentissage. 2 et 3 semble être un bon compromis mais une recherche du meilleur degré par grid search serait sûrement préférable.

Voici les courbes ROCs (avec valeurs des AUCs - les aires sous les courbes - en légende) pour les différents modèles :

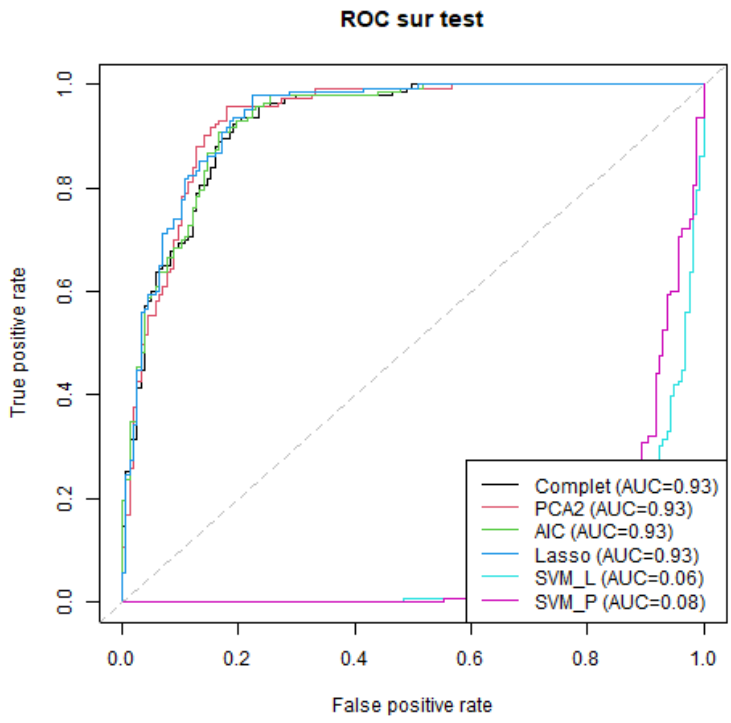

Fig. 9. – ROCs

On affiche les erreurs des différents modèles :

	train	test
Complet	0.1416667	0.1400000
PCA2	0.1350000	0.1233333
AIC	0.1433333	0.1466667
Lasso	0.1366667	0.1333333
SVM_L	0.8683333	0.8600000
SVM_P	0.8266667	0.8400000

On peut également visualiser les erreurs de test avec un bar plot :

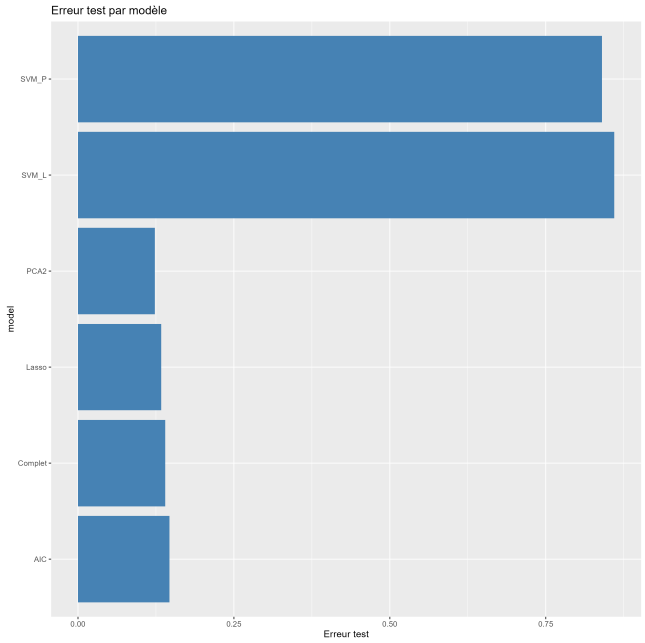

Fig. 10. – Erreurs test

Les modèles de régression logistique semblent bien meilleurs que les SVM, le commentaire des auteurs semble, sauf erreur de notre part, douteuse

3
FOCUS SUR L'ANALYSE DISCRIMINANTE

CONCLUSION