TP22 DATA : ElemStatLearn

GUARDIA Quentin, quentin.guardia@etu.u-paris.fr
M1 Cybersécurité FI

Fichier associé : prostate.R
La partie sur les données spam se trouve dans l'autre fichier.

Partie A :

Quelques informations sur la table prostate :

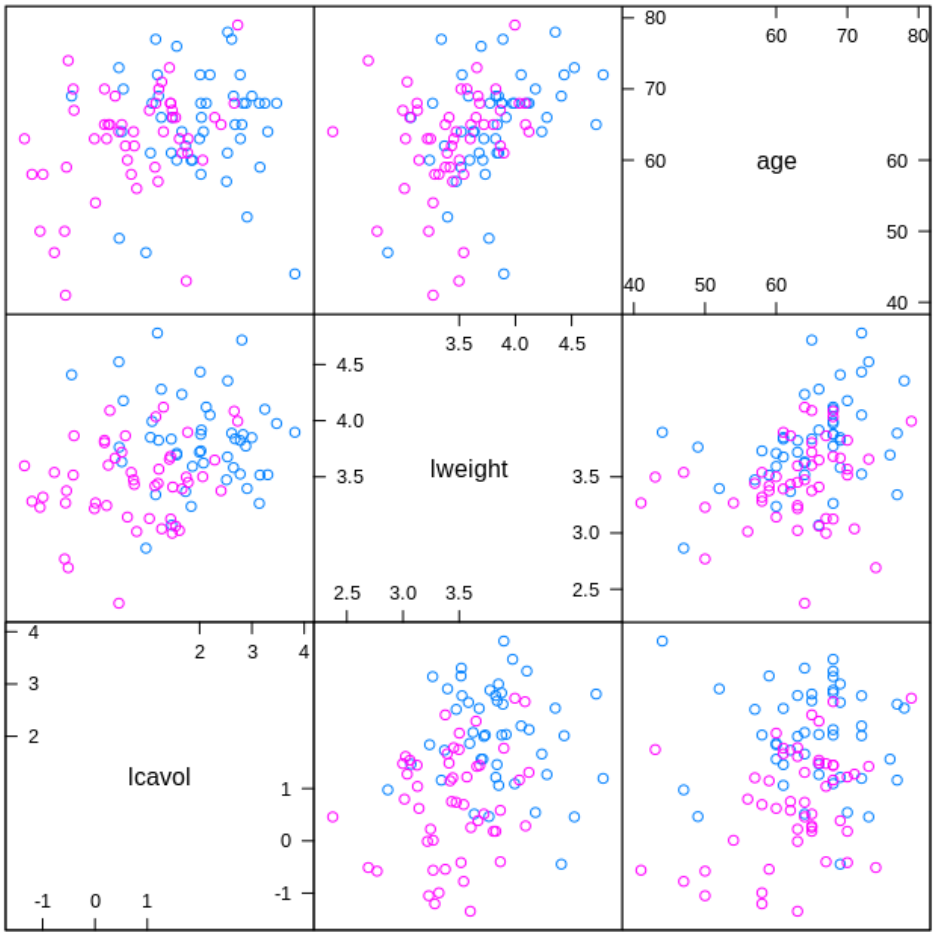
```
A data frame with 97 observations on the following 10 variables.
    lcavol log cancer volume
    lweight log prostate weight
    age in years
    lbph log of the amount of benign prostatic hyperplasia
    svi seminal vesicle invasion
    lcp log of capsular penetration
    gleason a numeric vector
    pgg45 percent of Gleason score 4 or 5
    lpsa response
```

Plus d'informations sur les statistiques avec l'instruction summary(prostate) et sur la nature des données avec ?prostate.

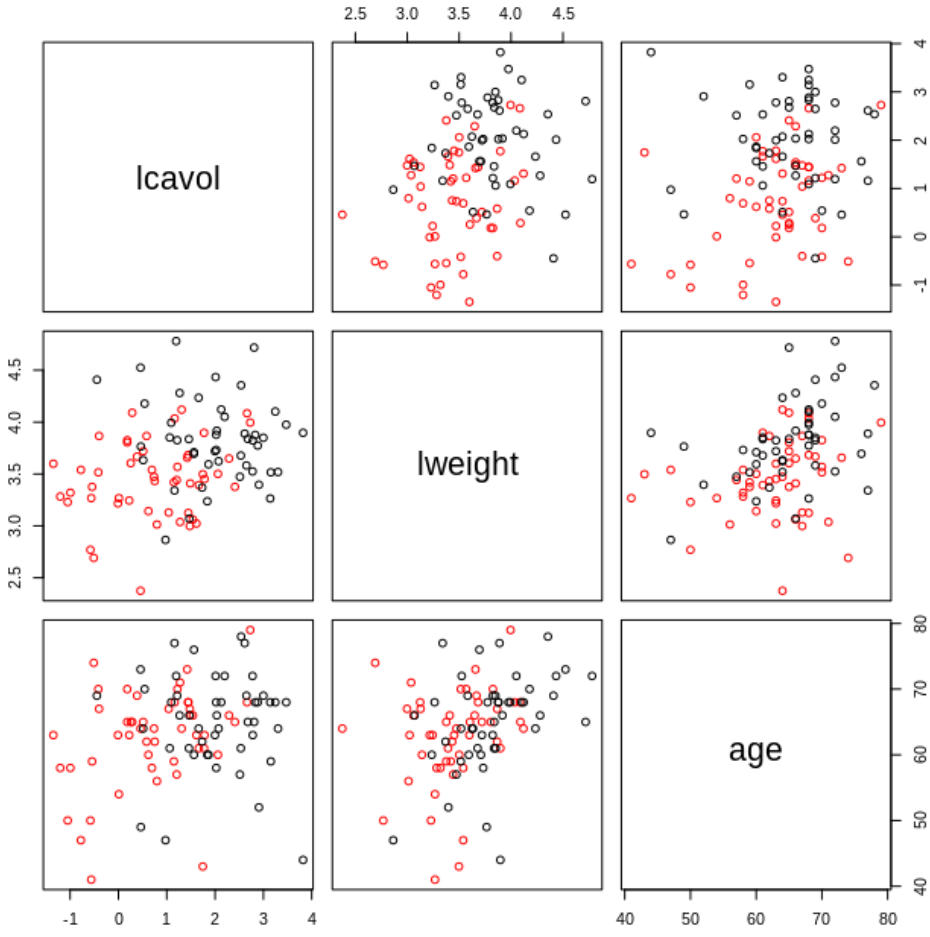
Binarisation de la lpsa en fonction de sa moyenne :

```
[1] low  low  low  low  low  low  low  low  low  low  low  low  low  low  low
[16] low  low  low  low  low  low  low  low  low  low  low  low  low  low  low
[31] low  low  low  low  low  low  low  low  low  low  low  low  low  low  low
[46] low  low  low  low  low  high high high high high high high high high high
[61] high high high high high high high high high high high high high high high
[76] high high high high high high high high high high high high high high high
[91] high high high high high high high
Levels: high low
```

Les nuages de point représentant les valeurs lcavol, lweight et age, les un en fonction des autres, sont ci-dessous. Sur les premiers nuages, les taux hauts sont en bleu et les taux bas en rose. Sur les seconds nuages, les taux hauts sont en noir et les taux bas en rouge.



Matrice de nuages de points

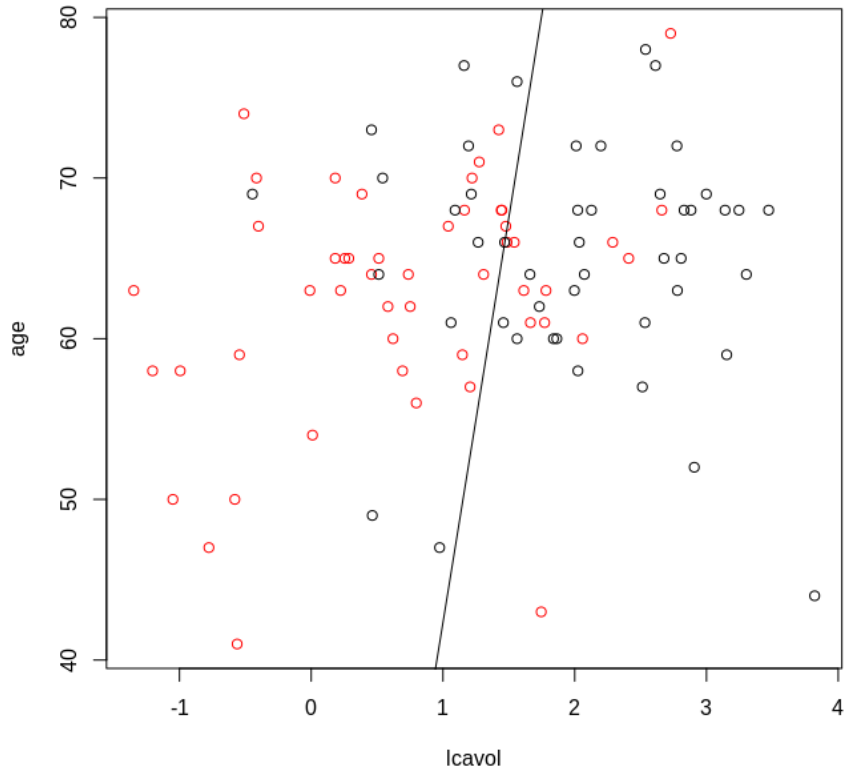


Partie B :

On transforme en indicateur binaire (high → 1 et low → 0)

```
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[39] 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[77] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Voici le modèle de régression linéaire avec la droite de prédiction



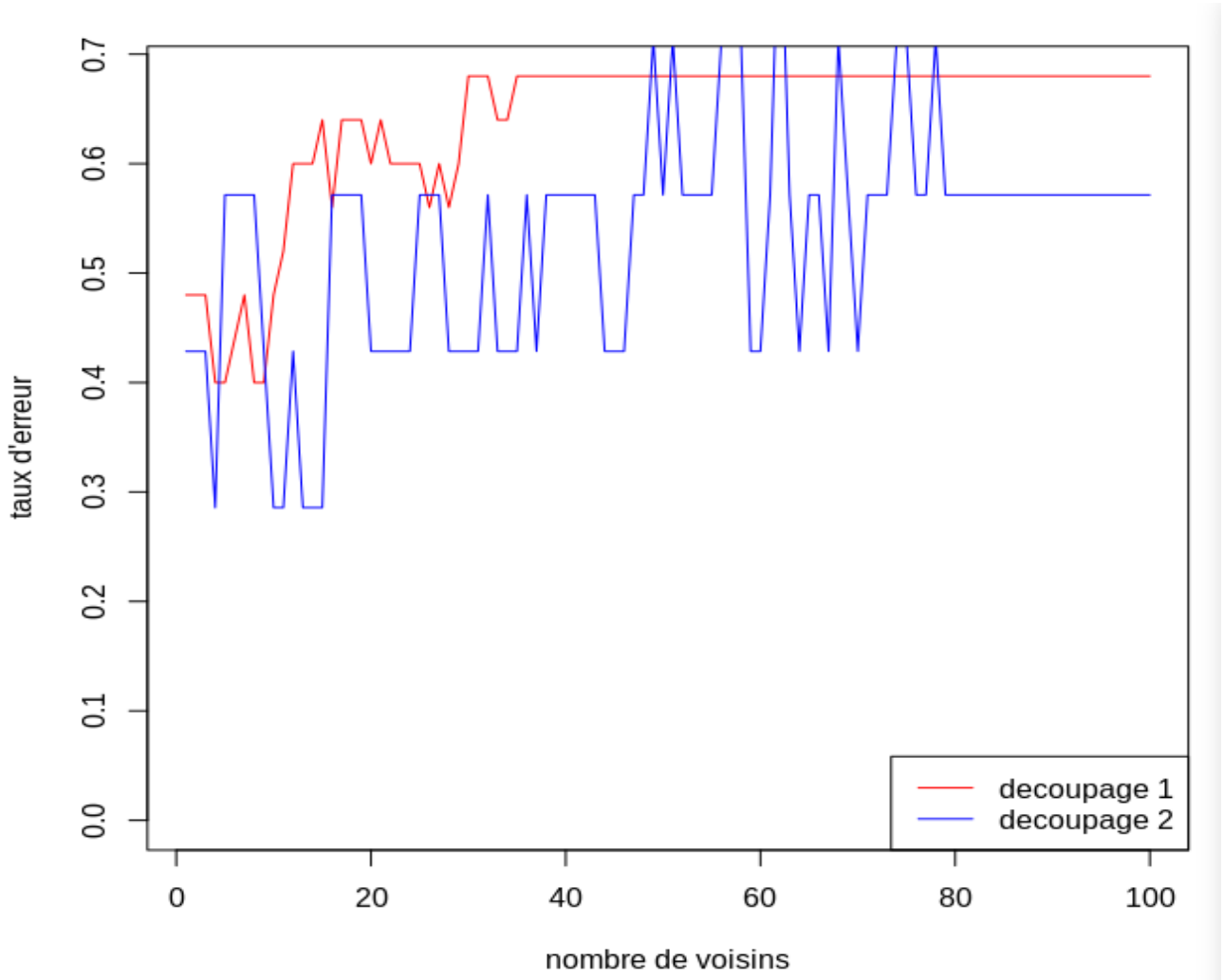
Il y a 19 exemples mal classés ce qui provoque un taux d'erreur de 0.1958763.

Voici la matrice de confusion associée, où l'on peut voir les erreurs :

```
        g
lm.ghat high low
  high   39  11
  low     8  39
```

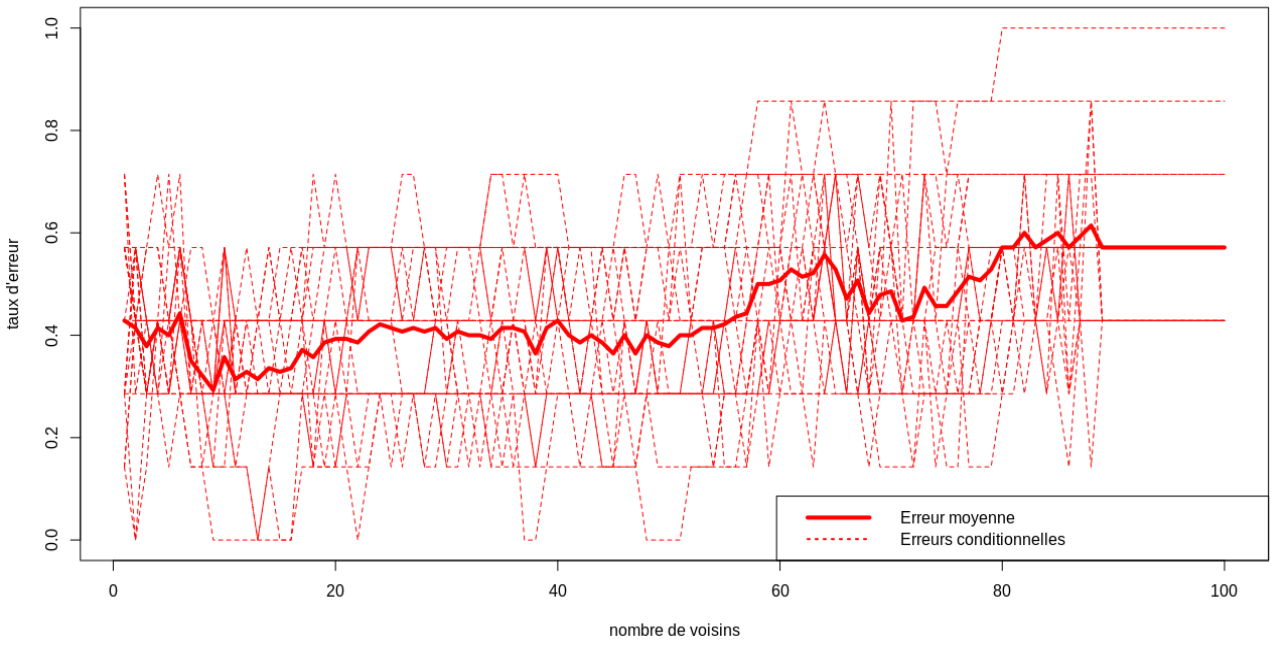
Partie C :

Ci-dessous, en rouge le taux d'erreur de la méthode knn sur les données test, avec les données d'apprentissage représentant 75 % des données.
En bleu un taux d'erreur où les données d'apprentissage représentent environ 93 % des données.
Les données sont exprimées en fonction du nombre de voisins k

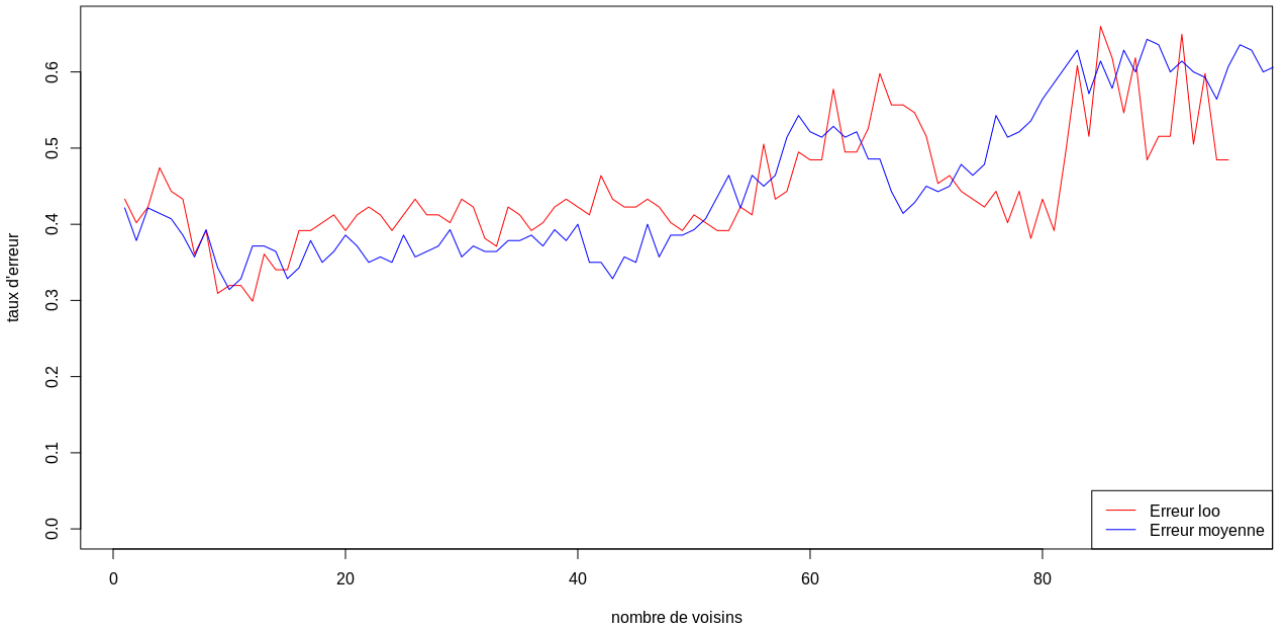


Il y a la plus petite erreur lorsqu'il y a 4 voisins.

Voici les erreurs moyennes et conditionnelles pour k=100. On voit qu'il y a le moins d'erreur lorsque k=4

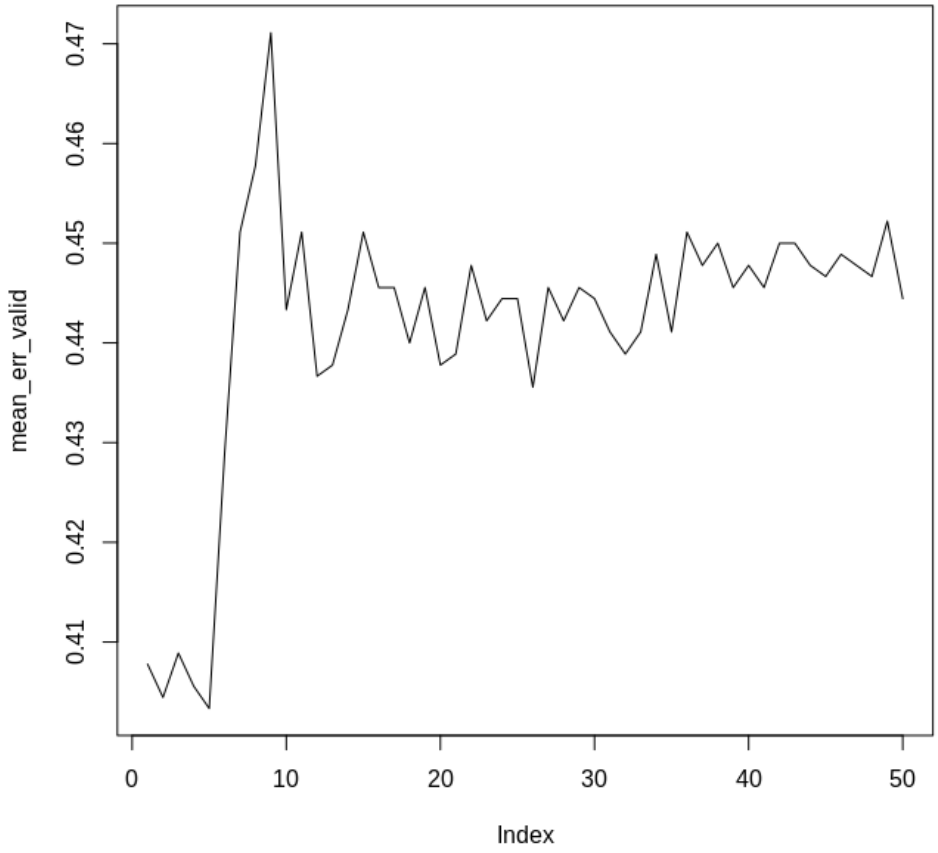


Avec k=100 (voisins), voici le taux d'erreur obtenu par cross-validation LOO avec la fonction knn.cv



Partie D :

L'ensemble apprentissage fait 75 % des données et l'ensemble test 25 %
Graphique de l'erreur de prédiction du classifieur :



L'erreur est minimale à 4 et a pour moyenne 0.4416222. On se rend compte en augmentant k que la moyenne de l'erreur reste comprise plus ou moins dans le même intervalle qu'entre l'abscisse 25 et 50.

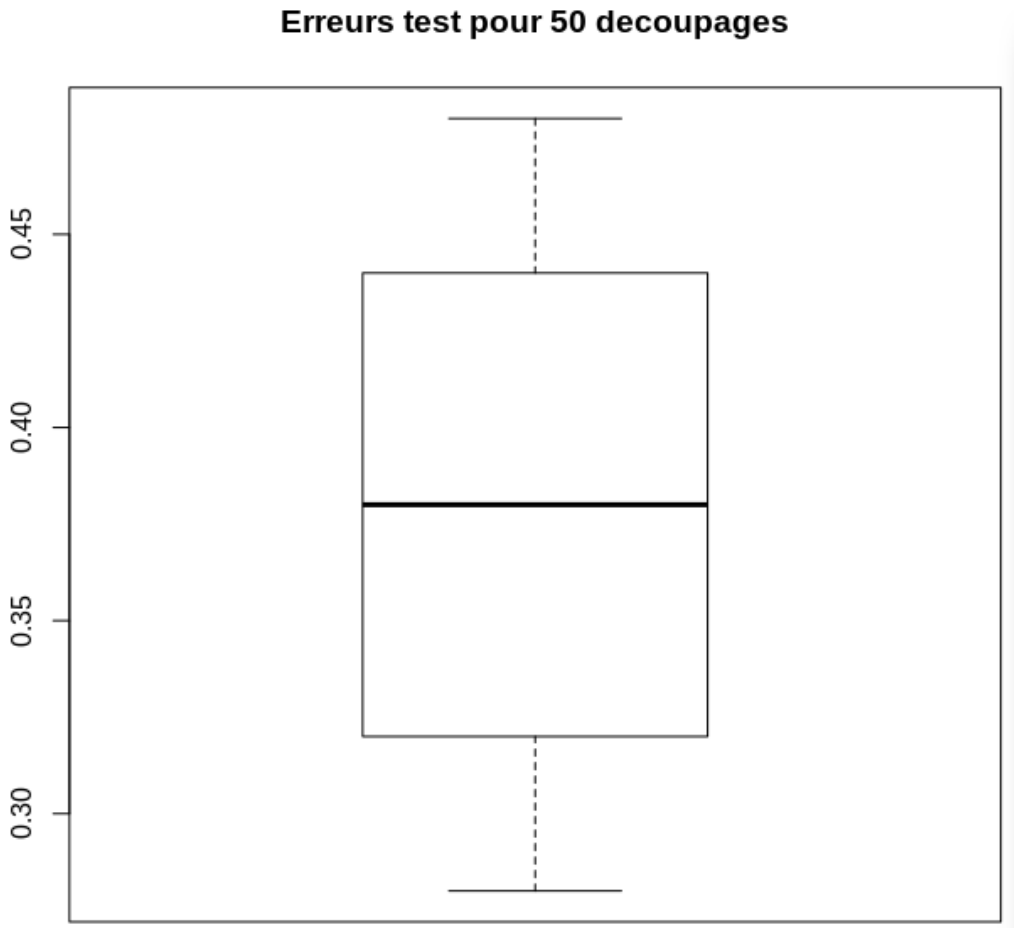
En construisant le classifieur sur l'ensemble "apprentissage-validation" , on trouve que le taux d'erreur sur les données tests est

```
> mean(err_valid)
[1] 0.4414889
```

Avec la méthode de validation croisée LOO vue précédemment pour trouver k, on trouve une moyenne d'erreur de 0.4 pour 5 erreurs

Partie E :

Ci-dessous la boîte à moustache des erreurs de test et ce pour 50 découpages différents



Bayésien naïf :

Réultats obtenus par le classifieur bayésien naïf de la table X :

```
A-priori probabilities:
Y
    high       low
0.4845361 0.5154639

Conditional probabilities:
      lcavol
Y          [,1]      [,2]
  high 2.0163957 0.9345935
```

```
 low  0.7236067 1.0369628

     lweight
Y         [,1]      [,2]
 high 3.823771 0.4019291
 low  3.445804 0.3705293

     age
Y         [,1]     [,2]
 high 65.10638 7.349162
 low  62.70000 7.418262

     lbph
Y           [,1]     [,2]
 high  0.3220379 1.485922
 low  -0.1080257 1.399836

     svi
Y          [,1]      [,2]
 high 0.4042553 0.4960529
 low  0.0400000 0.1979487

     lcp
Y           [,1]     [,2]
 high  0.4029949 1.486868
 low  -0.7267844 1.060140

     gleason
Y         [,1]      [,2]
 high 6.978723 0.6423246
 low  6.540000 0.7342913

     pgg45
Y         [,1]     [,2]
 high 35.7234 27.82375
 low  13.7200 24.33779

> table(predict(m, prostate.d), g)
     g
      high low
 high   31   6
 low    16  44
```