LERAY Maud

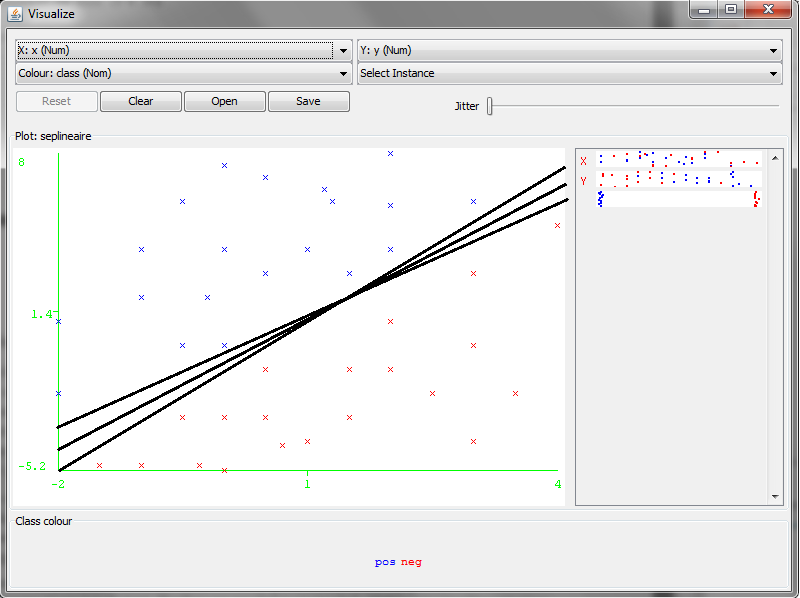
AIRIAU Pierre-Marie

**TP 1 : Acquisition de Connaissances**

**Apprentissage Supervisé**

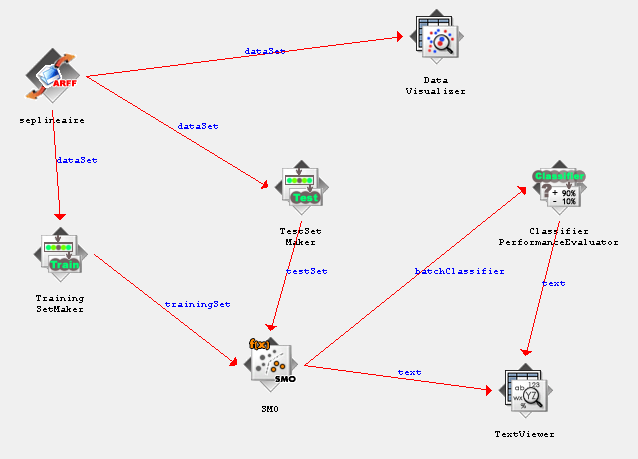
3.1 Données linéairement séparables

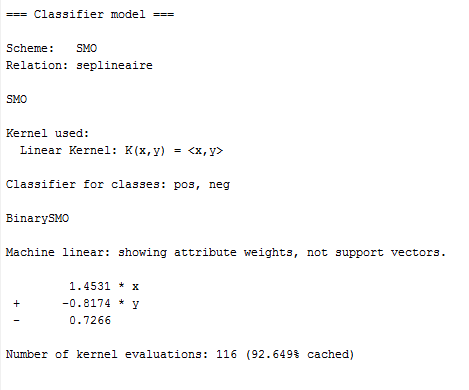
1.



2.

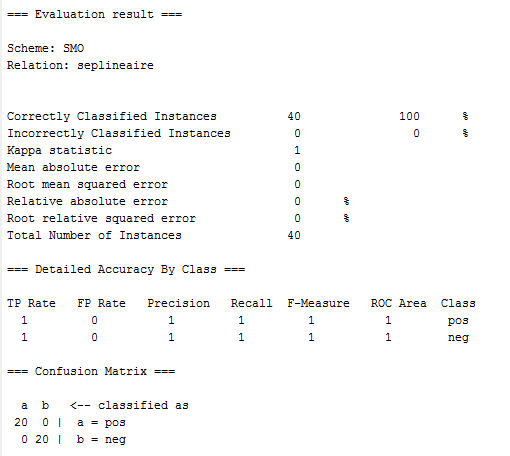
On estime que le risque empirique est de 0% car toutes les prédictions se révèlent correctes.





On peut voir l’équation cartésienne de la droite inférée ci-dessus : -0.8174\*y-0.7266+1.4531\*x=0

Ce qui revient à 0.8174\*y = 1.4531\*x – 0.7266

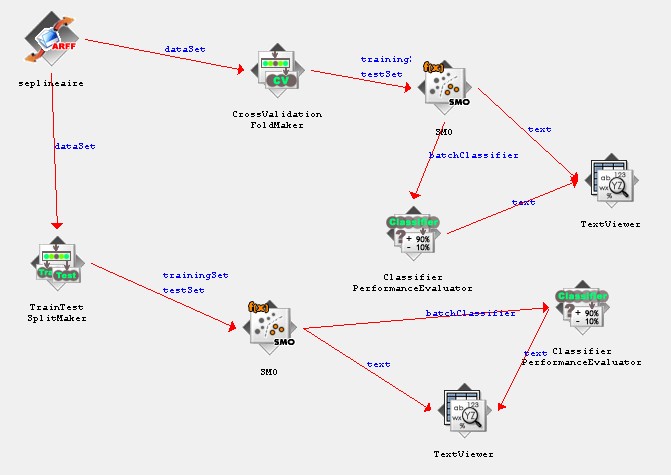


3.

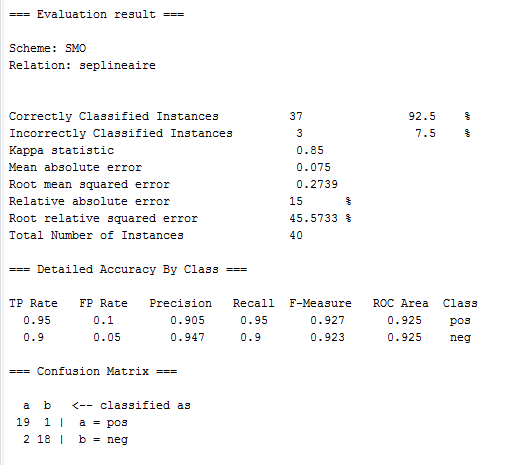
En effet, les résultats du test précédent n’apportent pas d’informations, car les données sont prédites à partir d’elles-mêmes. Il n’est donc pas étonnant d’avoir un taux de prédiction juste de 100%.

Le TrainTestSplitMaker permet d’expliquer une partie du jeu de données (par défaut 33%) appelée « jeu de test » avec le reste des données, appelé « jeu d’apprentissage ».

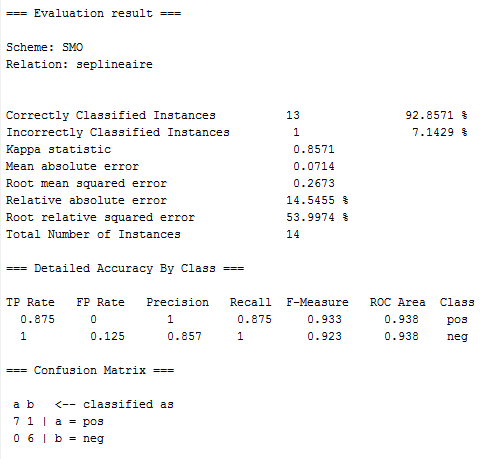
La CrossValidation sépare le jeu de données en plusieurs parties (par défaut 10), et réalise ensuite une succession de TrainTestSplitMaker en les prenant une par une comme jeu de test. Le résultat en sera la moyenne.



CrossValidation :



TrainTestSplitMaker :



3.2 Données non linéairement séparables

toleranceParameter = 0.0010

Number of Kernel evaluation : 1033

Correctly Classified Instances 113 75.3333 %

toleranceParameter = 0.0020

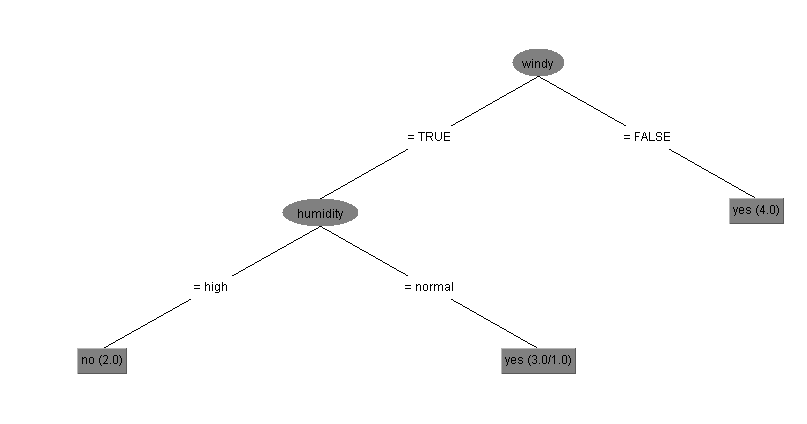
Number of Kernel evaluation : 1055

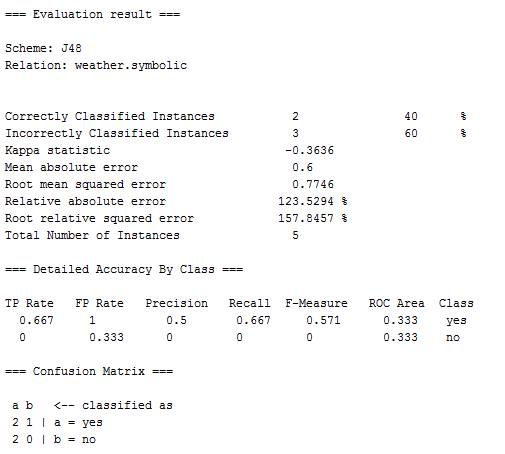
Correctly Classified Instances 116 77.3333 %

4.1 Construction et évaluation d’arbres

1. Il y a 14 instances et 5 attributs : outlook (enum), temperature (enum), humidity (enum), windy (bool), play (bool). La classe à prédire est Play (yes ou no).

2.





Il y a seulement 40% de bien classés.

En diminuant le nombre de données d’apprentissage à 33%, on obtient le même taux de prévisions correctes.

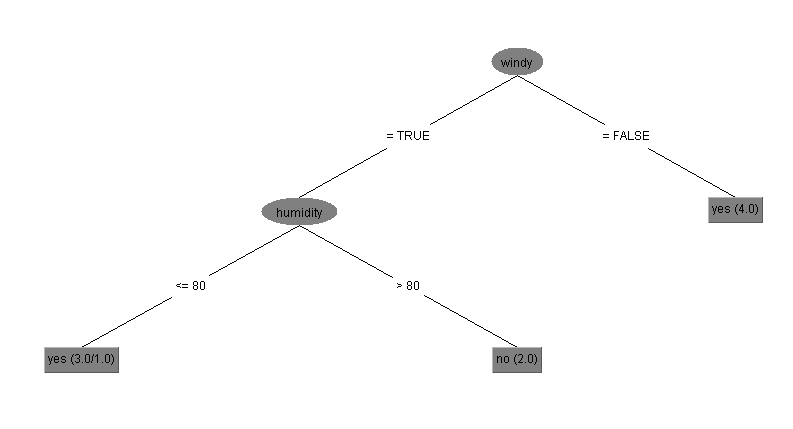
En mettant la graine 2 et le taux de données d’apprentissage à 66%, on passe à 80% de prévisions correctes.

Enfin, en plaçant la moitié des données dans le jeu d’apprentissage (avec une graine de 1), on obtient un taux de prévisions correctes de 57%.

* Il faut donc faire des compromis.

5.

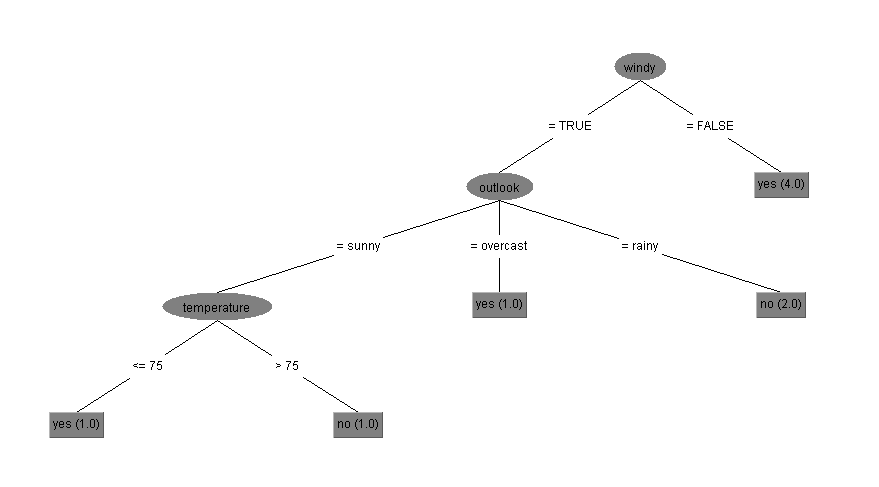
Dans le fichier weather.arff, les attributs humidity et temperature sont chiffrés.



Cela ne change rien à la précision de la prédiction.

4.2 Elagage et simplification

1.



2.

L’arbre obtenu est plus volumineux car non élagué. On observe également que le taux de prédictions correctes passe de 40% à 60%.

3.

En autorisant 2 exemples, on retombe sur l’arbre à deux niveaux de la question 4.5, avec le même taux de prédiction : 40%.

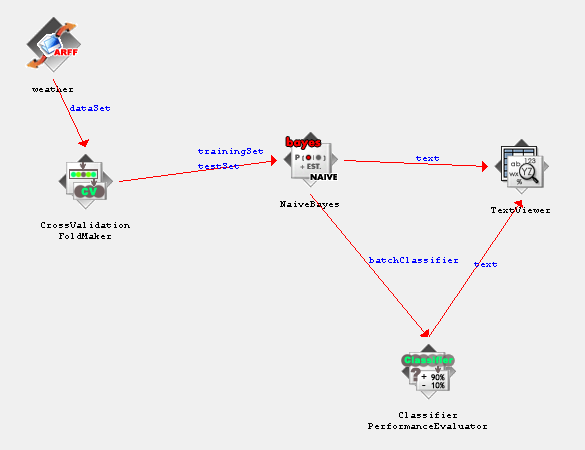
En en autorisant 3, on obtient un arbre binaire, et toujours le même taux de 40%.

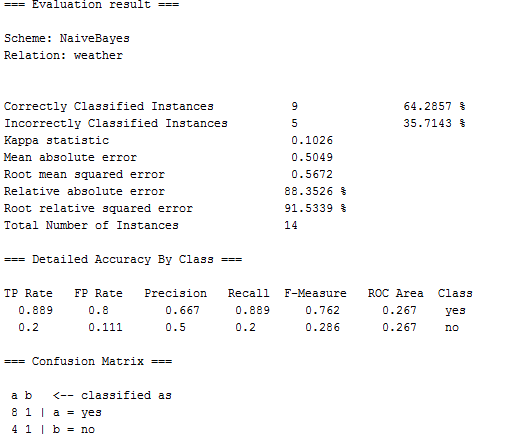
5.1 Bayes naïf

1.

L’hypothèse de Bayes Naïve est que tous les attributs sont indépendants.

2.



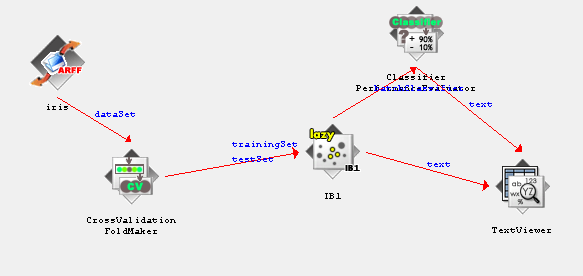


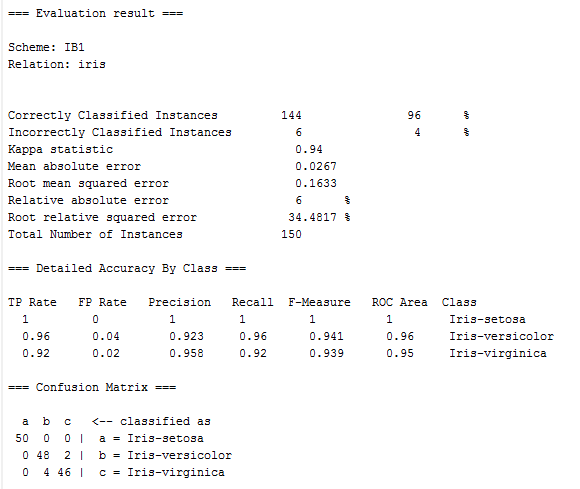
5.2 Approche non paramétrique

1.

Cet algorithme fait implicitement une estimation comparative de toutes les densités de probabilités des classes apparaissant dans le voisinage de l’instance à prédire, et choisit la plus probable.

2.





On observe que la prédiction de la classe d’iris est très bonne car son taux de prédictions correctes est de 96%. De plus la ROC area est très proche de 1 pour les trois classes, ce qui confirme la qualité prédictive.

3.

En passant à un algorithme 2-NN, on retrouve le même taux de prédictions correctes ainsi que la même matrice de confusion. On remarque une très faible augmentation de la ROC area.

4.

Suite à un essai avec l’algorithme 10-NN, on observe une ROC area d’au moins 0.99 pour chacune des espèces. Cela signifie que la ROC area et donc la qualité prédictive augmentent avec le nombre de plus proches voisins choisis. En revanche, le taux de prédictions correctes reste stable (du moins dans ce cas).