LERAY Maud

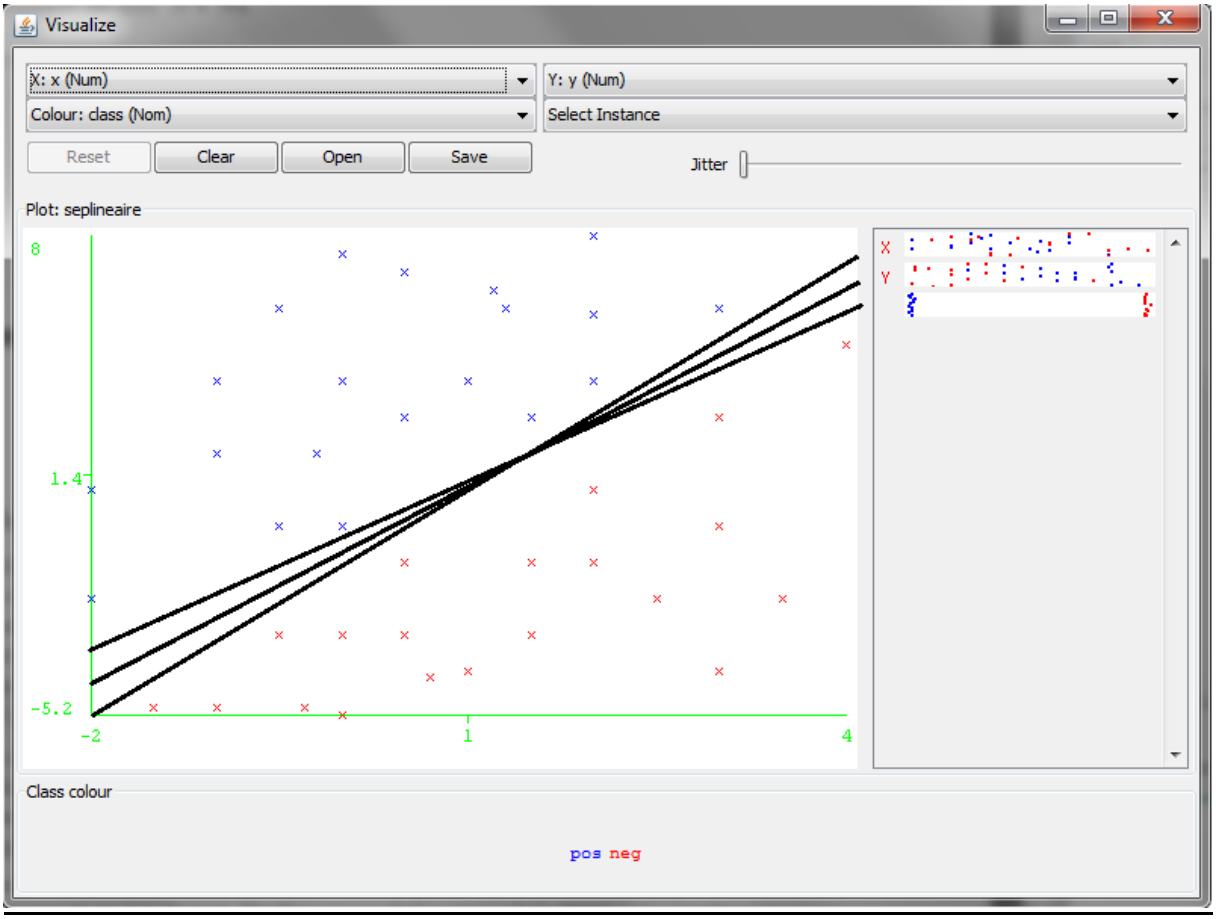
AIRIAU Pierre-Marie

TP 1 : Acquisition de Connaissances

Apprentissage Supervisé

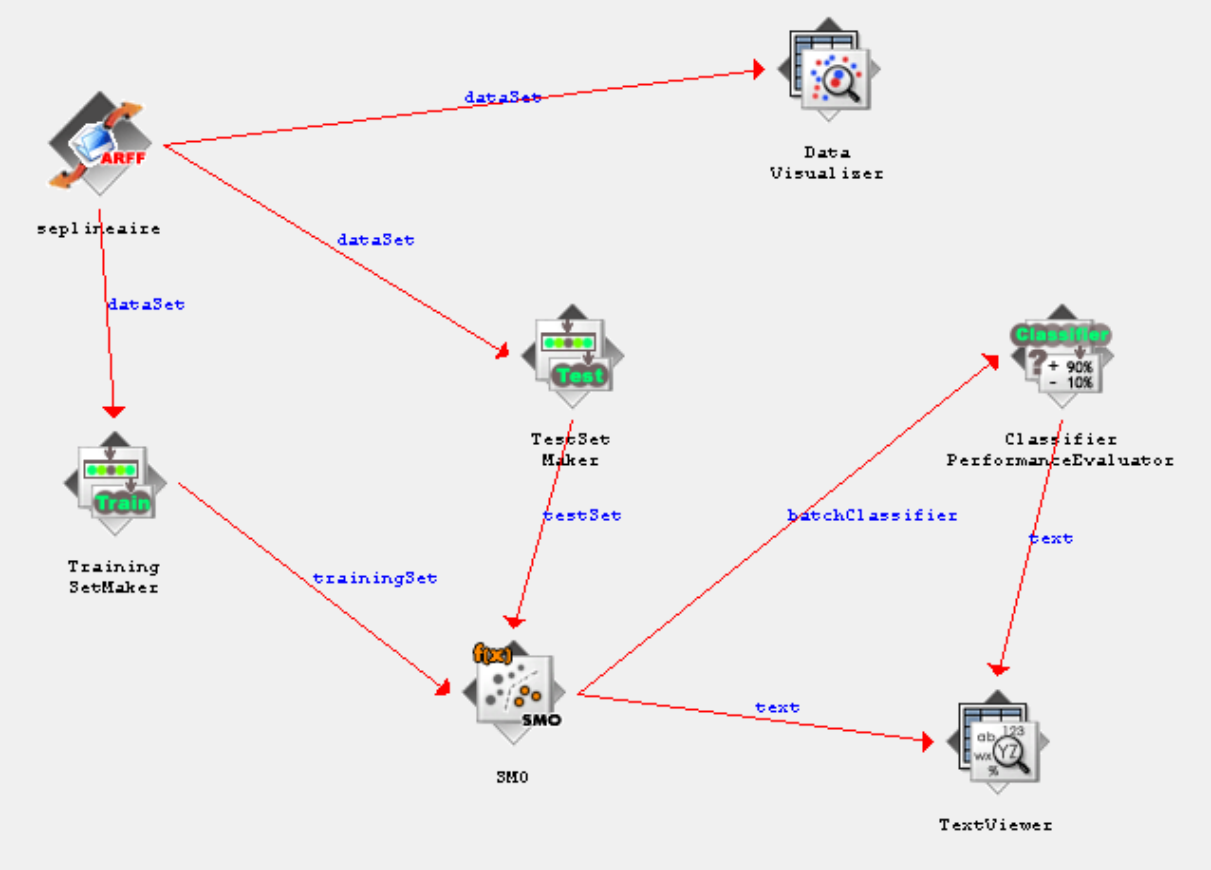
3.1 Données linéairement séparables

1.



2.

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




```
=== Classifier model ===

Scheme:   SMO
Relation: seplineaire

SMO

Kernel used:
  Linear Kernel: K(x,y) = <x,y>

Classifier for classes: pos, neg

BinarySMO

Machine linear: showing attribute weights, not support vectors.

        1.4531 * x
 +     -0.8174 * y
 -      0.7266

Number of kernel evaluations: 116 (92.649% cached)
```

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

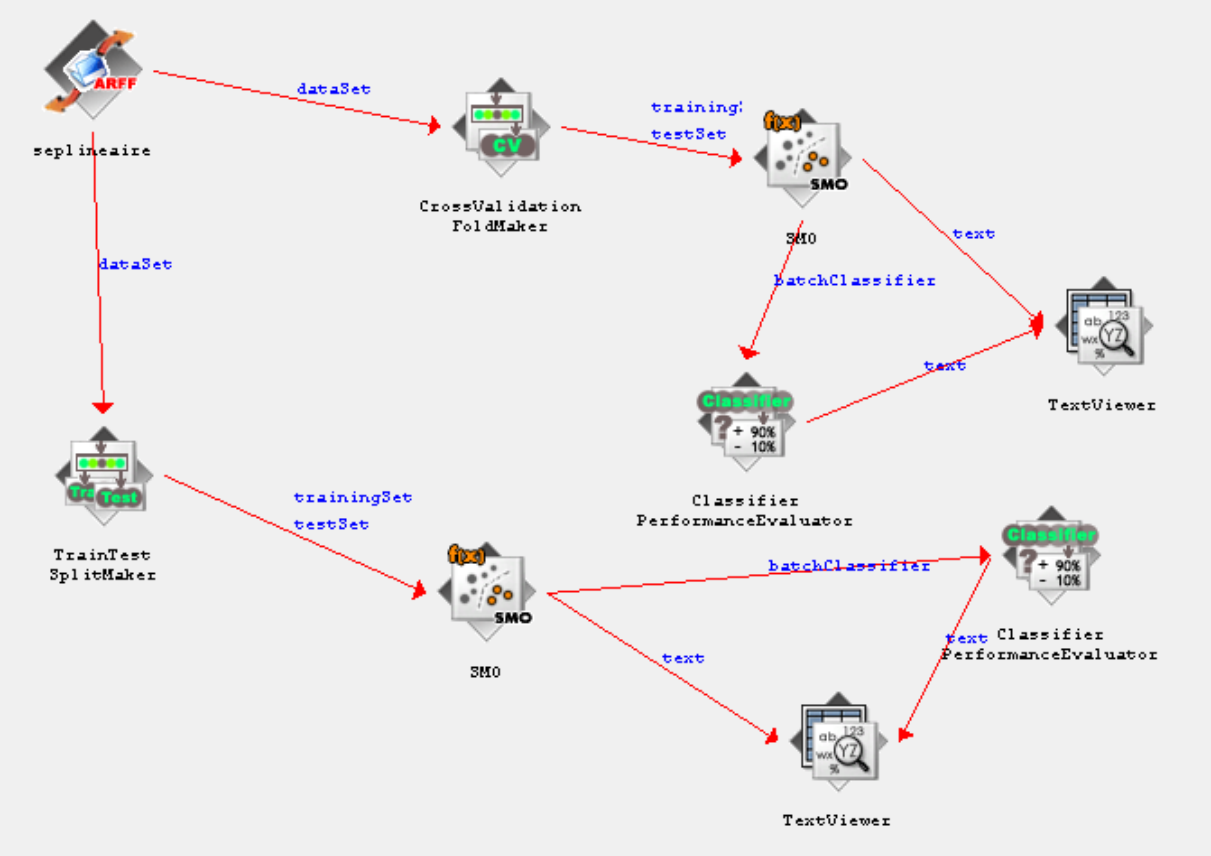
```
=== Evaluation result ===

Scheme: SMO
Relation: seplineaire


Correctly Classified Instances          40               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances               40

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  1         0          1         1         1          1        pos
  1         0          1         1         1          1        neg

=== Confusion Matrix ===

  a  b   <-- classified as
 20  0 |  a = pos
  0 20 |  b = neg
```

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 :

```
=== Evaluation result ===

Scheme: SMO
Relation: seplineaire


Correctly Classified Instances          37               92.5    %
Incorrectly Classified Instances         3                7.5    %
Kappa statistic                          0.85
Mean absolute error                      0.075
Root mean squared error                  0.2739
Relative absolute error                 15      %
Root relative squared error             45.5733 %
Total Number of Instances               40

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.95      0.1        0.905     0.95      0.927      0.925    pos
  0.9       0.05       0.947     0.9       0.923      0.925    neg

=== Confusion Matrix ===

  a  b   <-- classified as
 19  1 |  a = pos
  2 18 |  b = neg
```

TrainTestSplitMaker :

```
=== Evaluation result ===

Scheme: SMO
Relation: seplineaire


Correctly Classified Instances          13               92.8571 %
Incorrectly Classified Instances         1                7.1429 %
Kappa statistic                          0.8571
Mean absolute error                      0.0714
Root mean squared error                  0.2673
Relative absolute error                 14.5455 %
Root relative squared error             53.9974 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.875     0          1         0.875     0.933      0.938    pos
  1         0.125      0.857     1         0.923      0.938    neg

=== Confusion Matrix ===

 a b   <-- classified as
 7 1 | a = pos
 0 6 | b = neg
```

3.2 Données non linéairement séparables

toleranceParameter = 0.0010

Number of Kernel evaluation : 1033

Correctly Classified Instances 113 75.3333 %

toleranceParameter = 0.0020

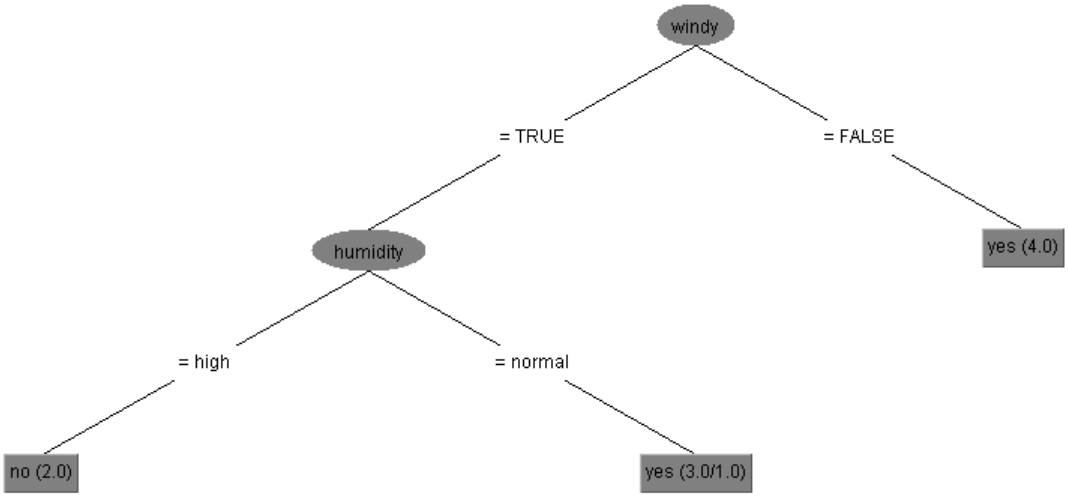
Number of Kernel evaluation : 1055

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.




```
=== Evaluation result ===

Scheme: J48
Relation: weather.symbolic


Correctly Classified Instances           2               40      %
Incorrectly Classified Instances         3               60      %
Kappa statistic                         -0.3636
Mean absolute error                      0.6
Root mean squared error                  0.7746
Relative absolute error                123.5294 %
Root relative squared error            157.8457 %
Total Number of Instances                5

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.667     1          0.5       0.667     0.571      0.333    yes
  0         0.333      0         0         0          0.333    no

=== Confusion Matrix ===

 a b   <-- classified as
 2 1 | a = yes
 2 0 | b = no
```

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.

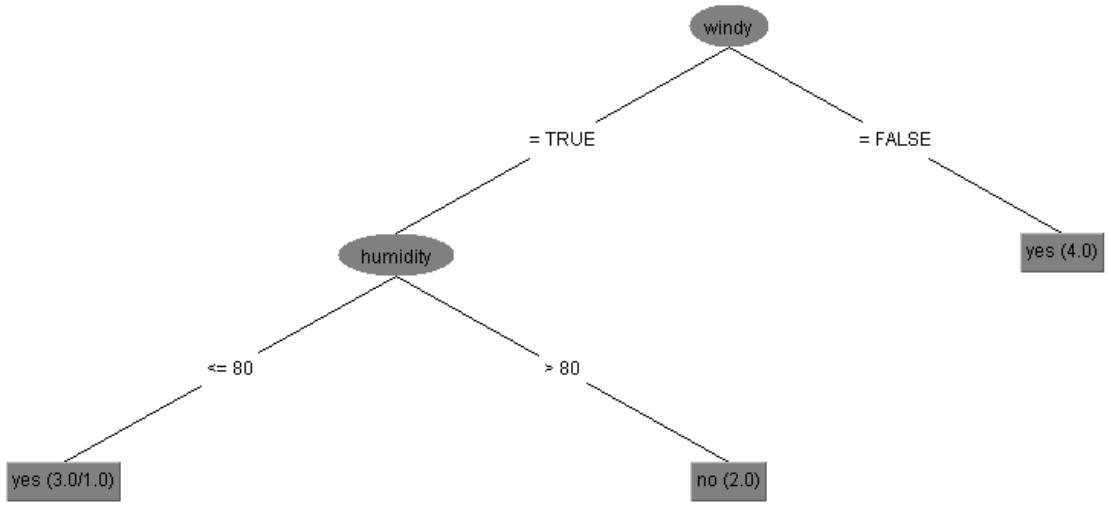
En mettant la graine 2 et le taux de données d'apprentissage à 66%, on passe à 80% de prévisions correctes.

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.

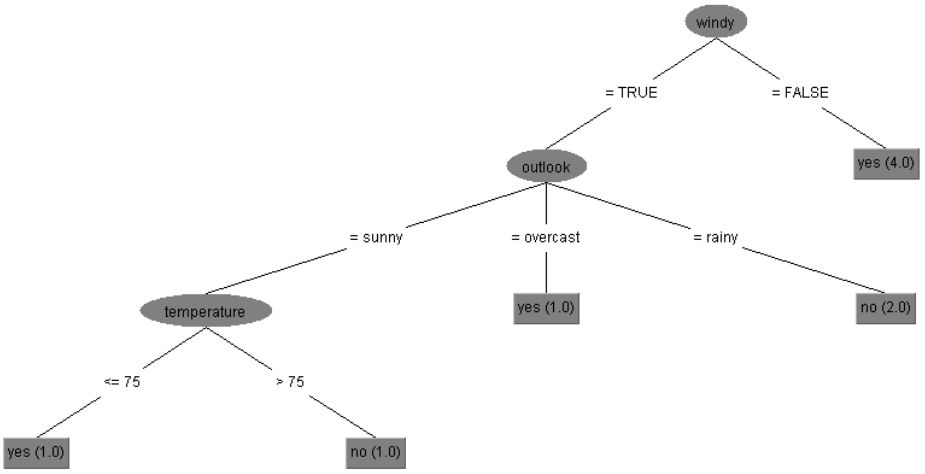
Dans le fichier weather.arff, les attributs humidity et temperature sont chiffrés.



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%.

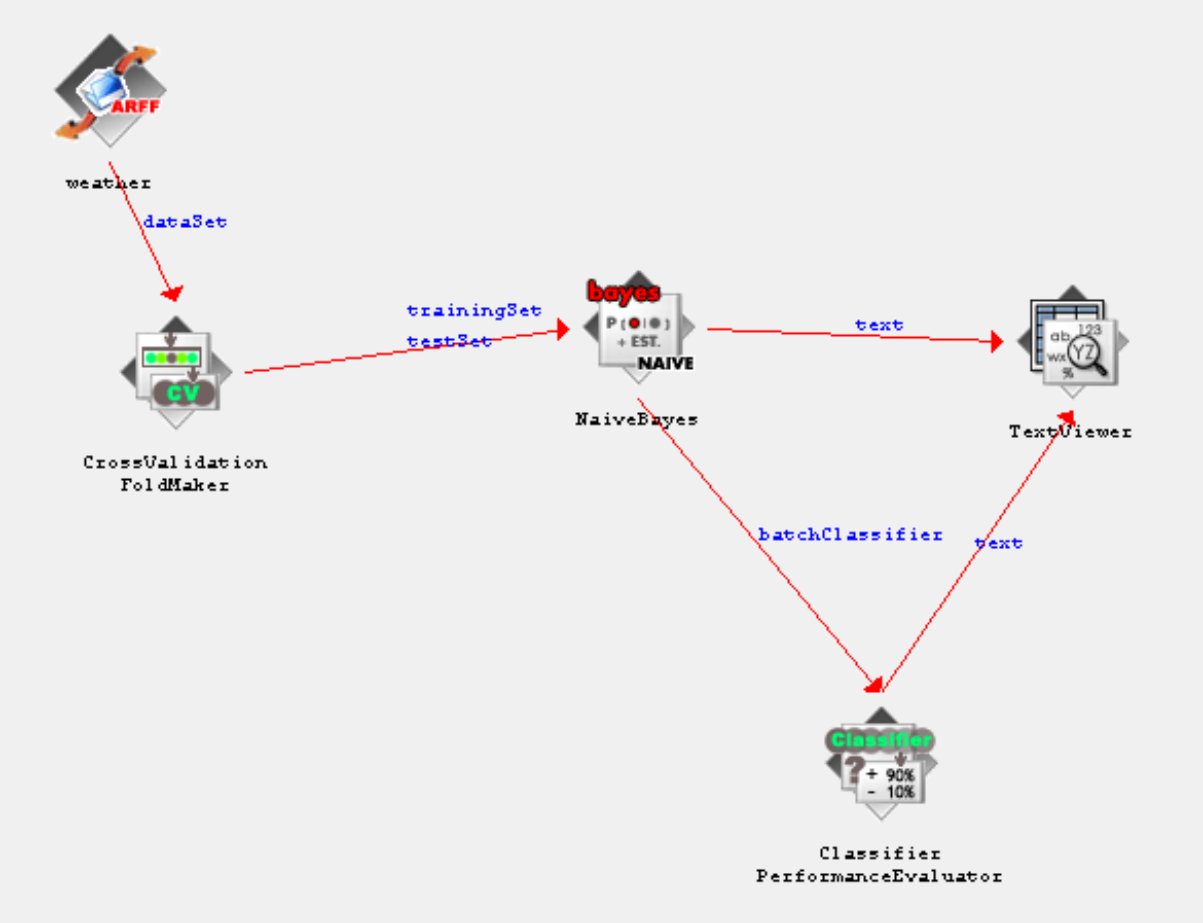
En en autorisant 3, on obtient un arbre binaire, et toujours le même taux de 40%.

5.1 Bayes naïf

1.

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

2.



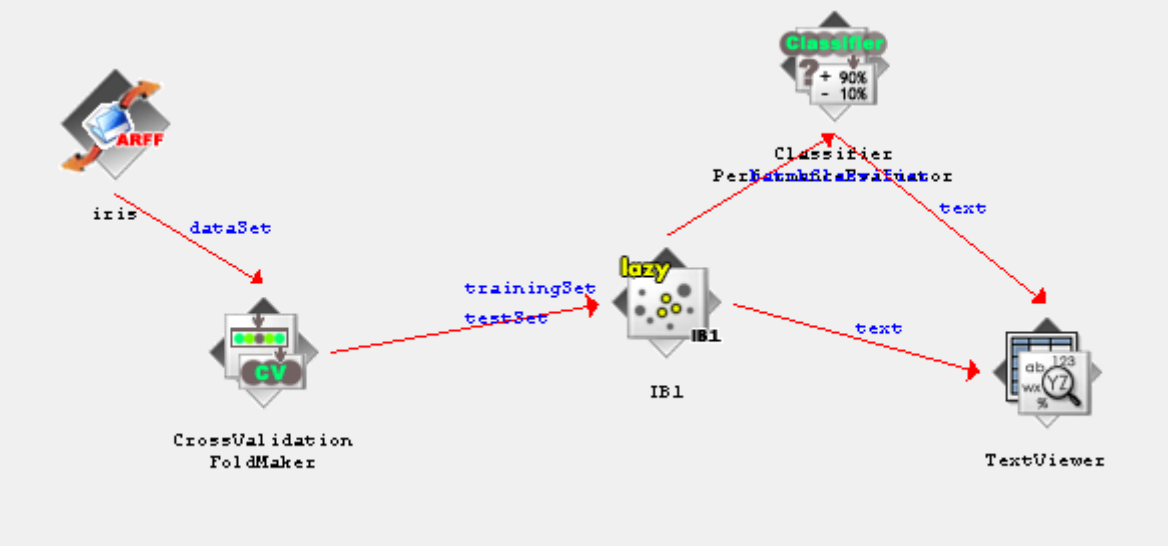
```
=== Evaluation result ===

Scheme: NaiveBayes
Relation: weather


Correctly Classified Instances           9               64.2857 %
Incorrectly Classified Instances         5               35.7143 %
Kappa statistic                          0.1026
Mean absolute error                      0.5049
Root mean squared error                  0.5672
Relative absolute error                 88.3526 %
Root relative squared error             91.5339 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.889     0.8        0.667     0.889     0.762      0.267    yes
  0.2       0.111      0.5       0.2       0.286      0.267    no

=== Confusion Matrix ===

 a b   <-- classified as
 8 1 | a = yes
 4 1 | b = no
```

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.



```
=== Evaluation result ===

Scheme: IB1
Relation: iris


Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.0267
Root mean squared error                  0.1633
Relative absolute error                  6      %
Root relative squared error             34.4817 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  1         0          1         1         1          1        Iris-setosa
  0.96      0.04       0.923     0.96      0.941      0.96     Iris-versicolor
  0.92      0.02       0.958     0.92      0.939      0.95     Iris-virginica

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
  0 48  2 |  b = Iris-versicolor
  0  4 46 |  c = Iris-virginica
```

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).