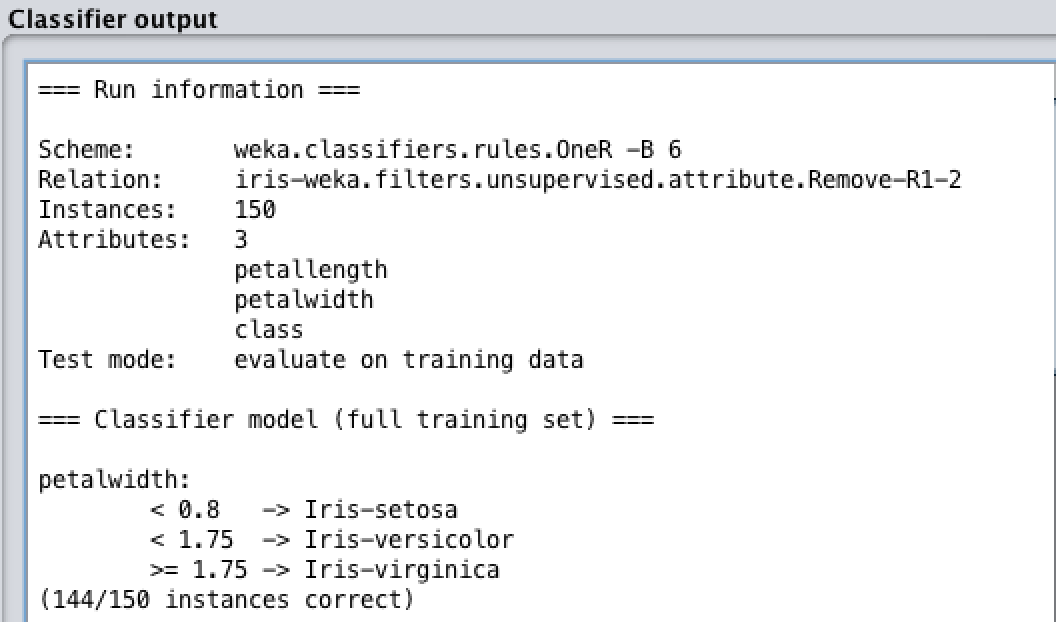
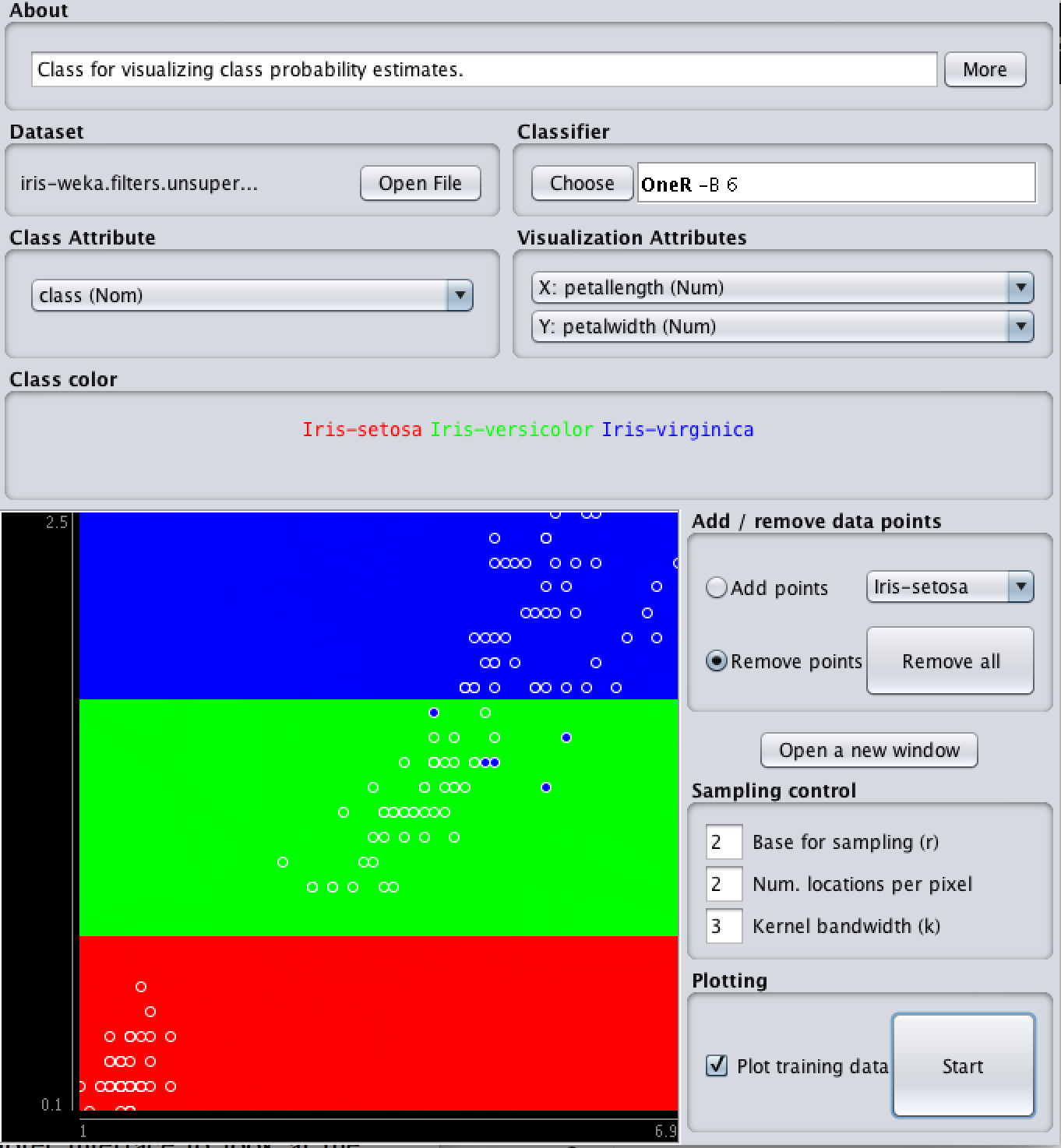
**IS590DT assignment#4**

**yuweic3**

**2017/9/18**

**# 17.3.1**

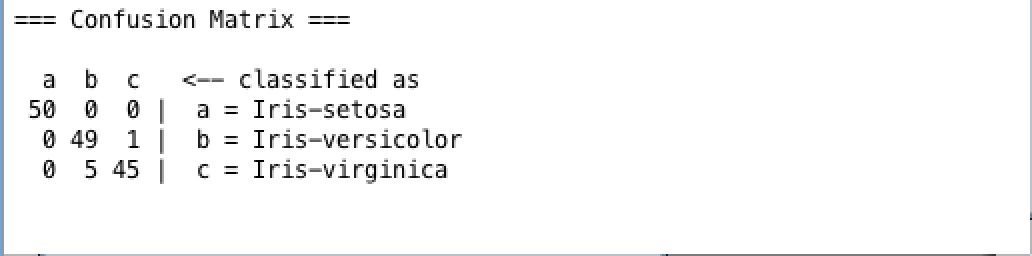


Explanation:

From the Visualization panel, we find that 1R-6 classifier classified the iris training set into 3 type based on the petalwidth:

1. Iris-setosa(red) with petal width < 0.8
2. Iris-versicolor(green) with petalwidth < 1.75
3. Iris-virginica(blue) with petalwidth >= 1.75

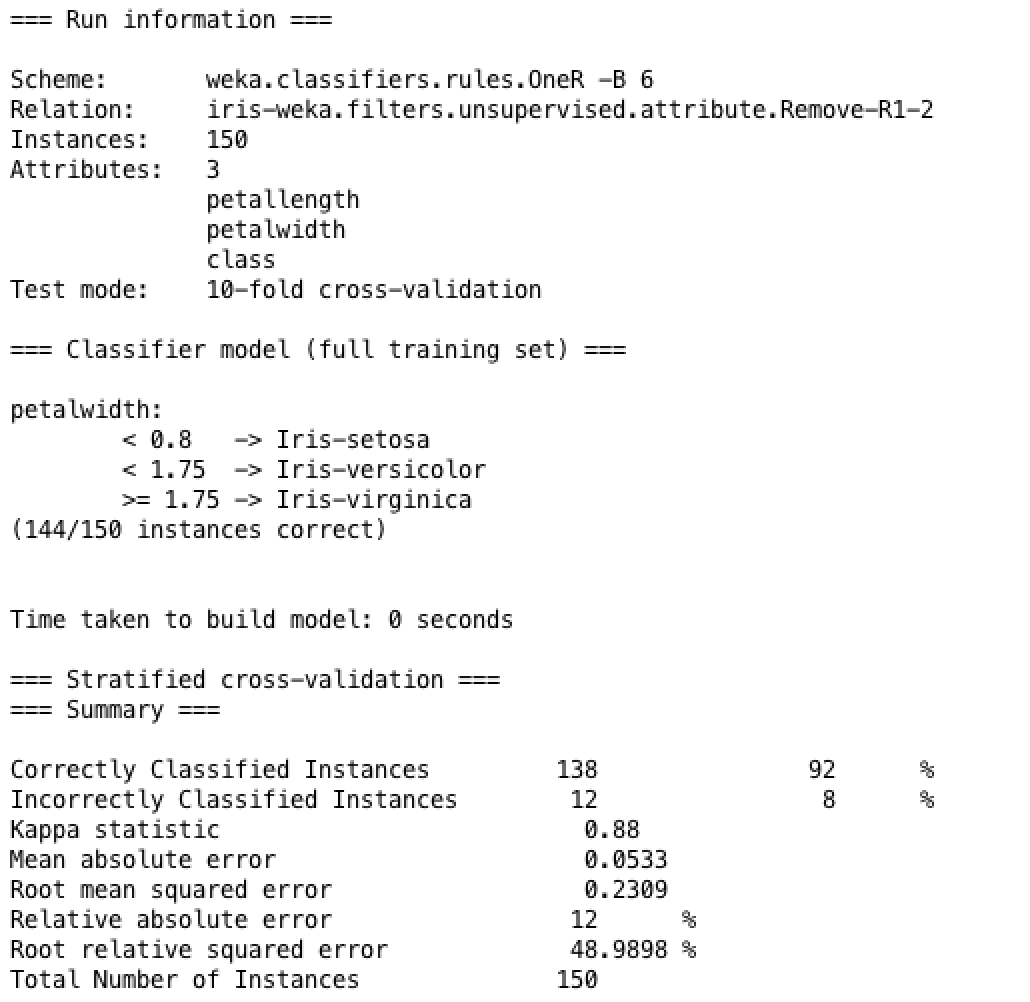
Besides, from the Classifier output, we also notice that there are 5 Iris-virginica (type c) sample are mistakenly assigned into Iris-versicolot (type b)



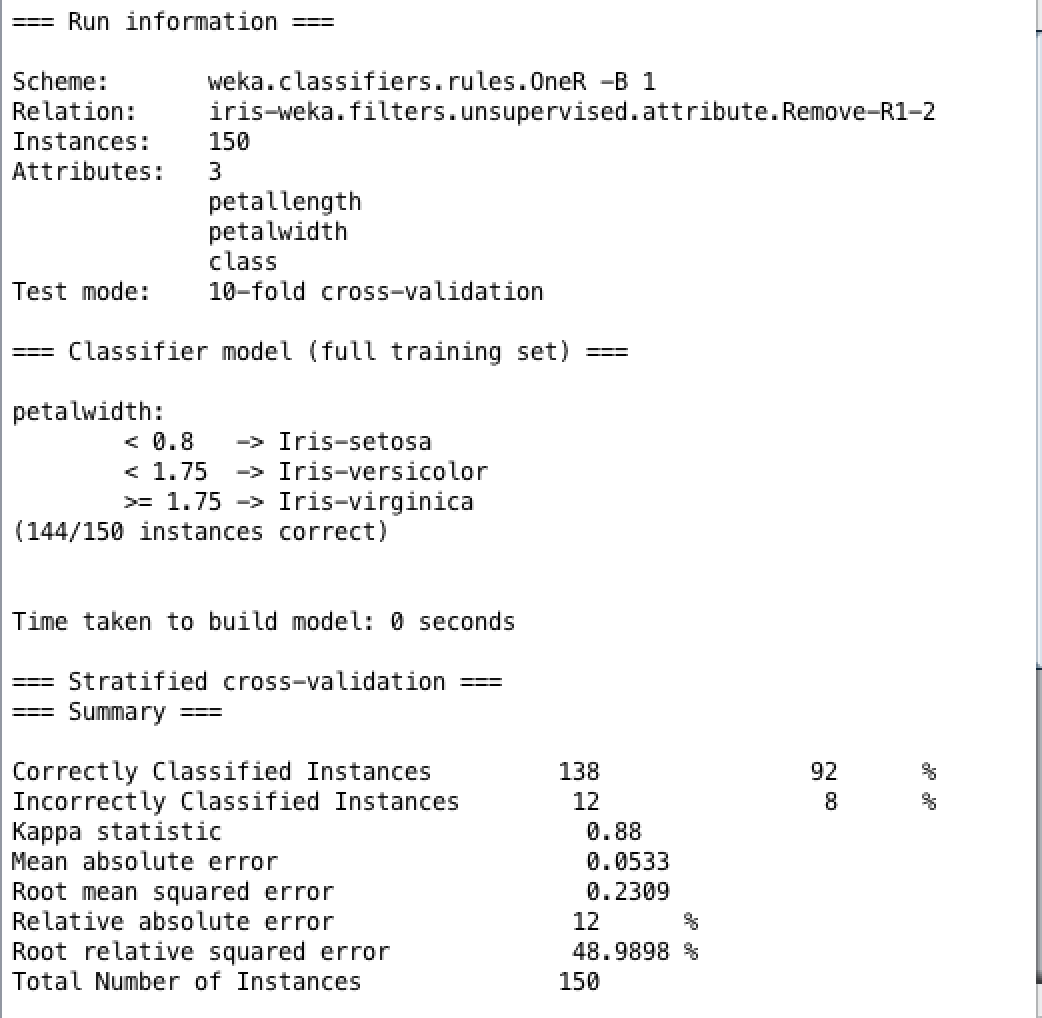
**# 17.3.2**

The is used for avoid overfitting. The default value is 6. Theoretically, when it is changed to 1, the number of correctly classified instance should be decrease. Similarly, by changing the parameter to something larger, it will prevent classifier to be overfitting.

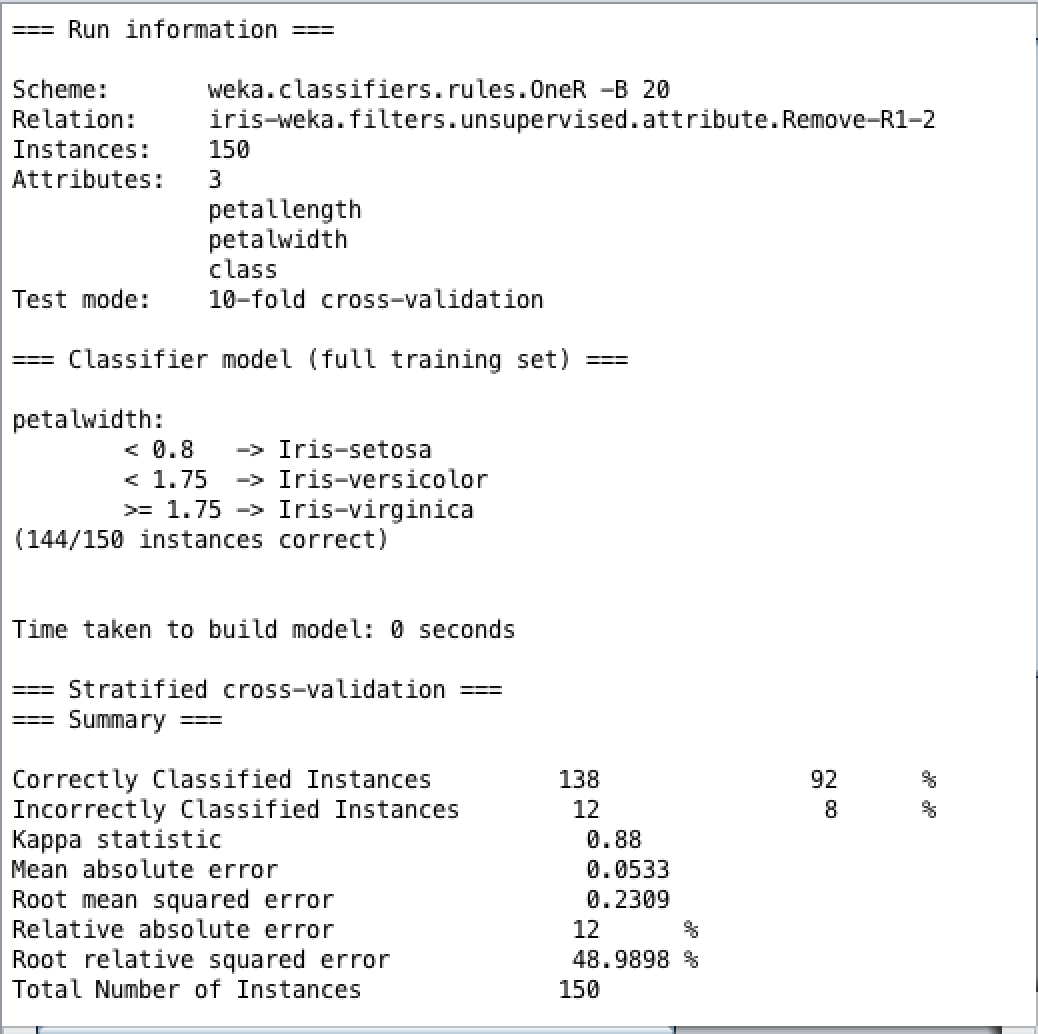
However, when I change the from 6 to 1 and 20, nothing is changed. The results of the classification are always the same and the rules do not change in Explorer



OneR - 6



OneR - 1



OneR-20

**# 17.3.3**

When , there are 3 large areas of color in the visualization. The classification is done according to the classes and there are only 3 classes for classification.

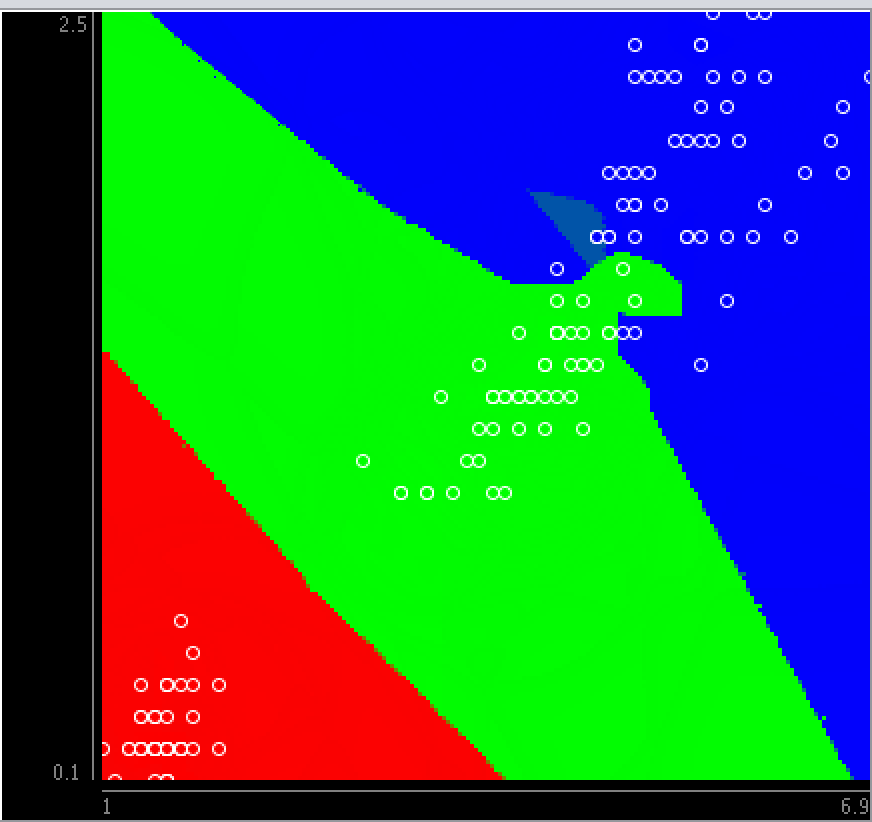
**# 17.3.4**

It is not possible to have less than 3 regions no matter how the value is changed, because there are 3 classes in iris dataset.

In fact, the iris-setona class set the minimum value for . When , we have 3 distinct regions.

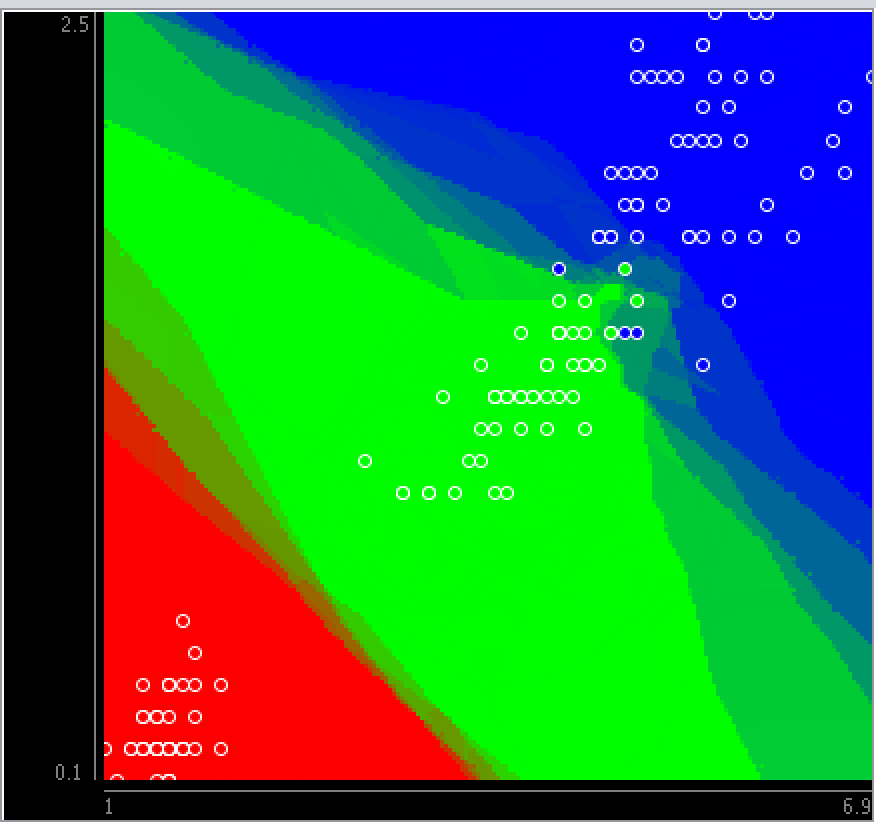
From the Confusion Matrix, we know that there are at most 6 trans-class elements in neighboring class. The valud of can be less than 6.

**# 17.3.5**

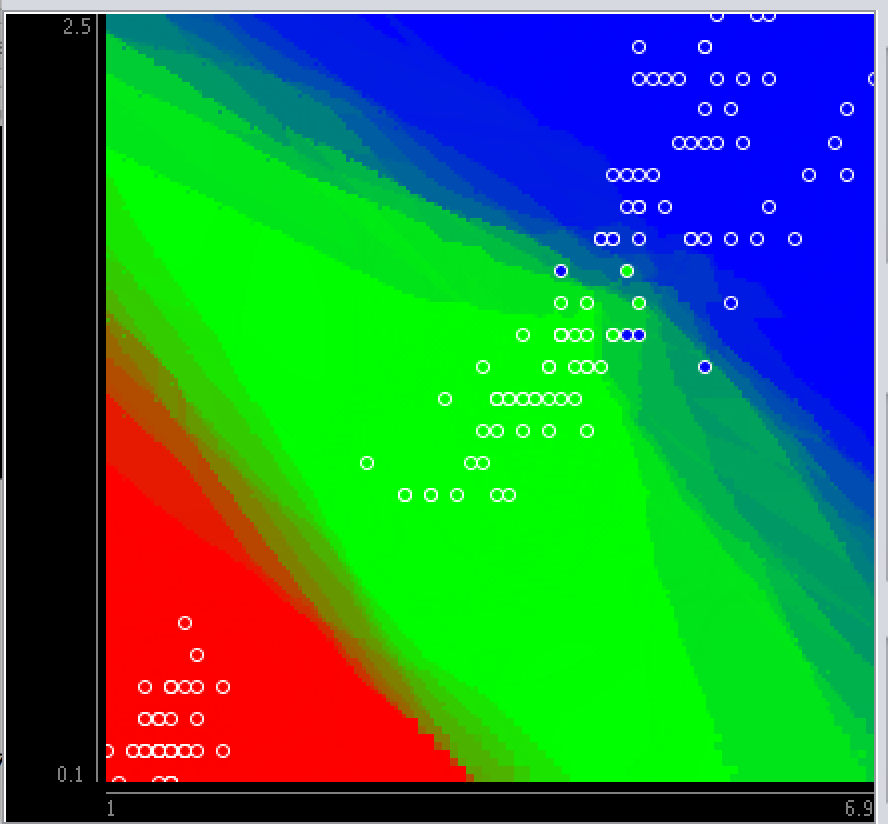


From the visualization of the classification of iris.2D.arff with IBk(k=1), we find that there are 3 large distinct regions of red, green and blue. Blue and green region are superimposed as some blue sample and green sample are superimposed on each other.

**# 17.3.6**



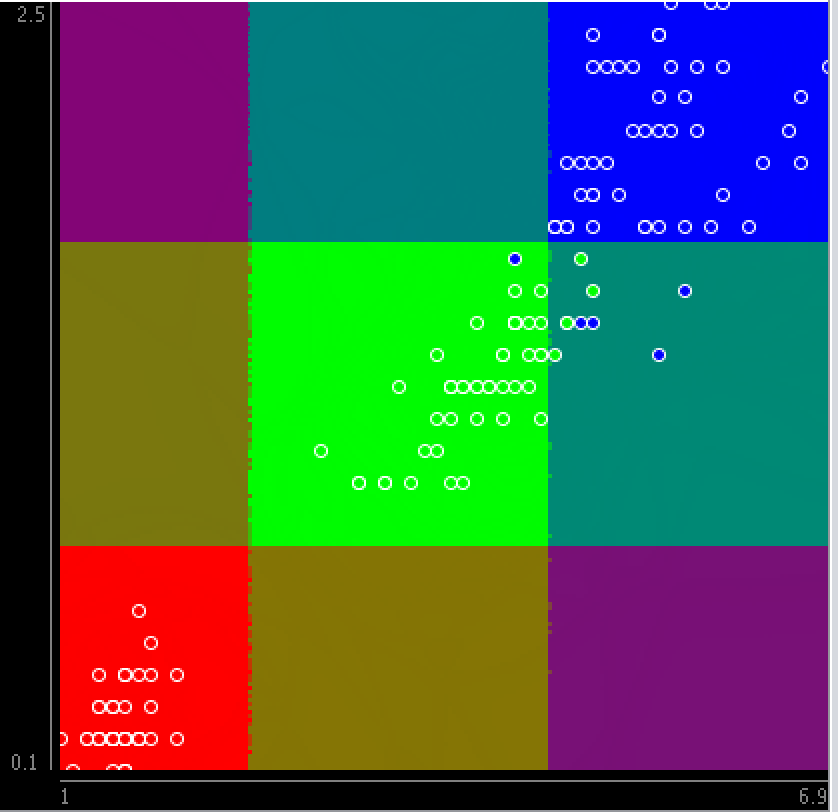
K=5



K=10

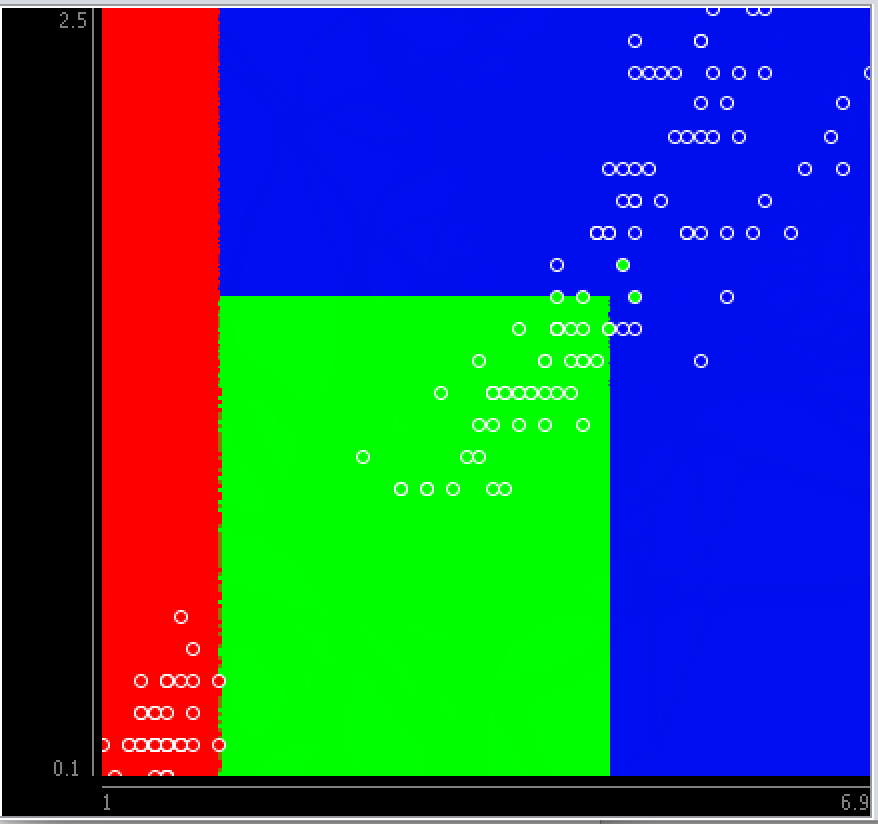
In IBK-k, when we increase the k, we would consider more neighbors for each center. For lower k, it tends to be overfitting and the model is with low bias and high variance. ; For higher k, we will get a smoother model with low variance but high bias.

**# 17.3.7**



Different from other classification visualization, the visualization of Naïve Bayes classification shows 9-colored grids. For each grid, its color is determined by the multiplication of the sample present on the axis and the sample present on the axis.

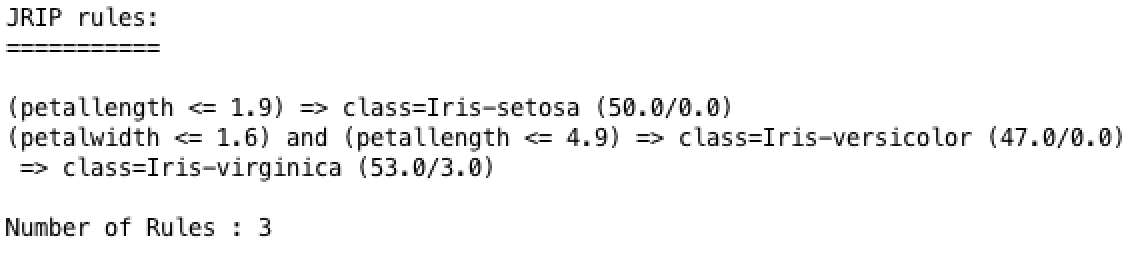
**# 17.3.8**



JRip visualization and JRip rules:

There are 3 regions representing the 3 classes.

There are 3 rules for this classifier.



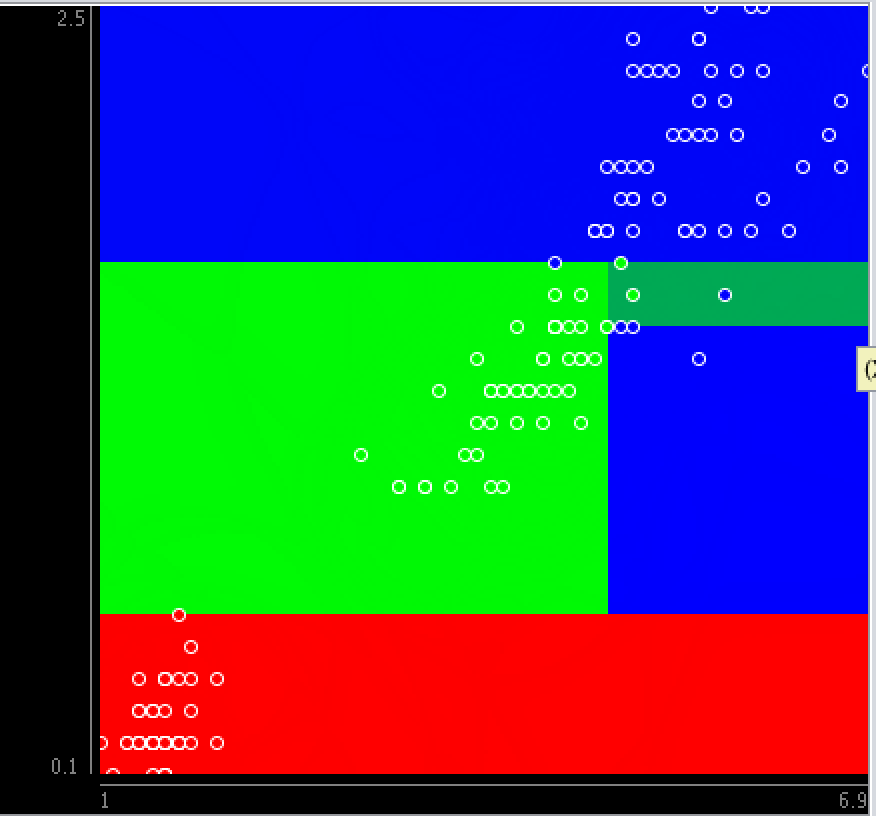
**# 17.3.9**

-(petallength <= 1.9) => class = iris-setosa

-(petallength <= 1.6) and (petallength <= 4.9) and (petallength > 1.9) => class = iris-versicolor

-(petalwidth > 1.6) and (petallength > 4.9) => class = iris-virginica

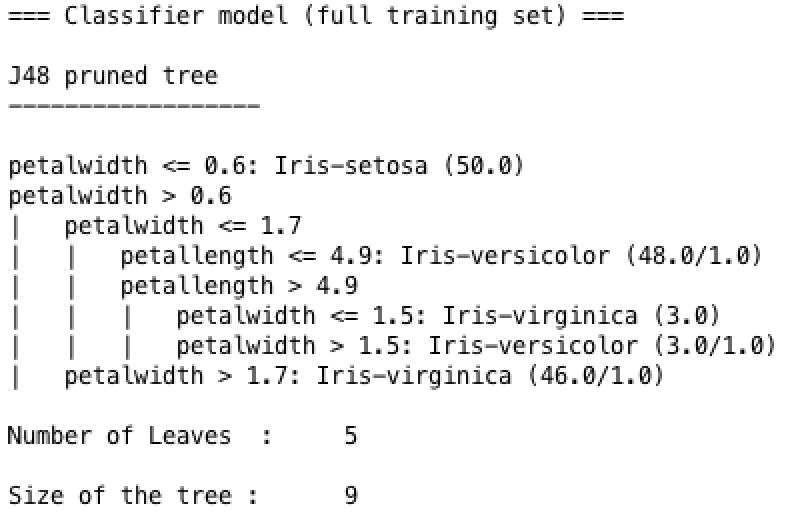
**# 17.3.10**



J48

Compared with JRip, we find that there are more regions in J48 as there are more classification conditions.

There are 5 leaves with 50, 49,47， 3 and 4 samples respectively. Leaves with 3 and 4 sample can be grouped together in “petallength > 4.9”group.



**# 17.3.11**

For 1 leaf node: the value of

For 2 leaf nodes only: the value of as already have 50 cases.

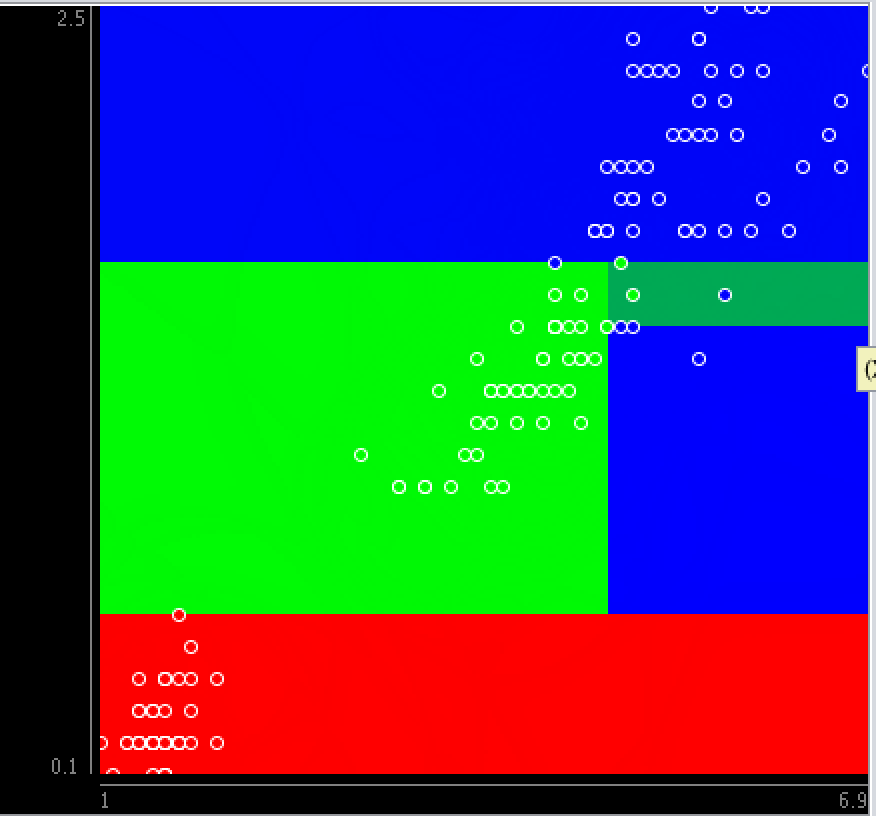
For 3 leaf node: the value of as has group 50 cases in one branch; Then contains 47 cases in second branch; Finally, the rest will be group together as the third group.

**# 17.3.12**

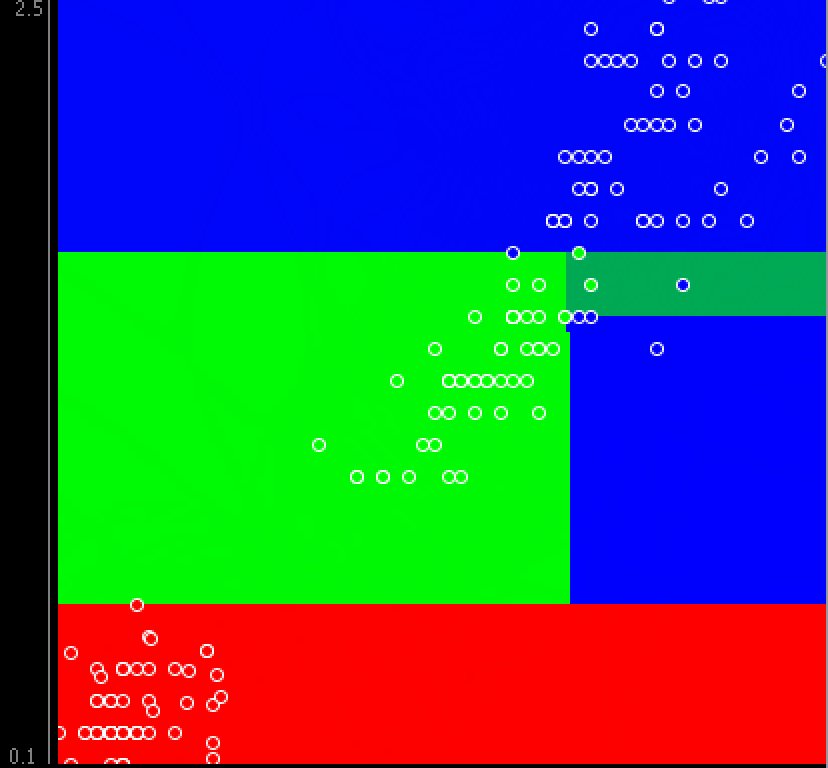
When we add some noise, there are two possible cases that can be observed

1. If the noise points are added in its right region, then the visualization will not change a lot;
2. If the noise points are added in the different region, it will change the visualization

Take J48 as an example, before we add noise points, its visualization graph is below:



1. If we add some iris-setosa noise points in original red area



1. If we add some iris-setosa noise points in green region:

