BOSTON UNIVERSITY

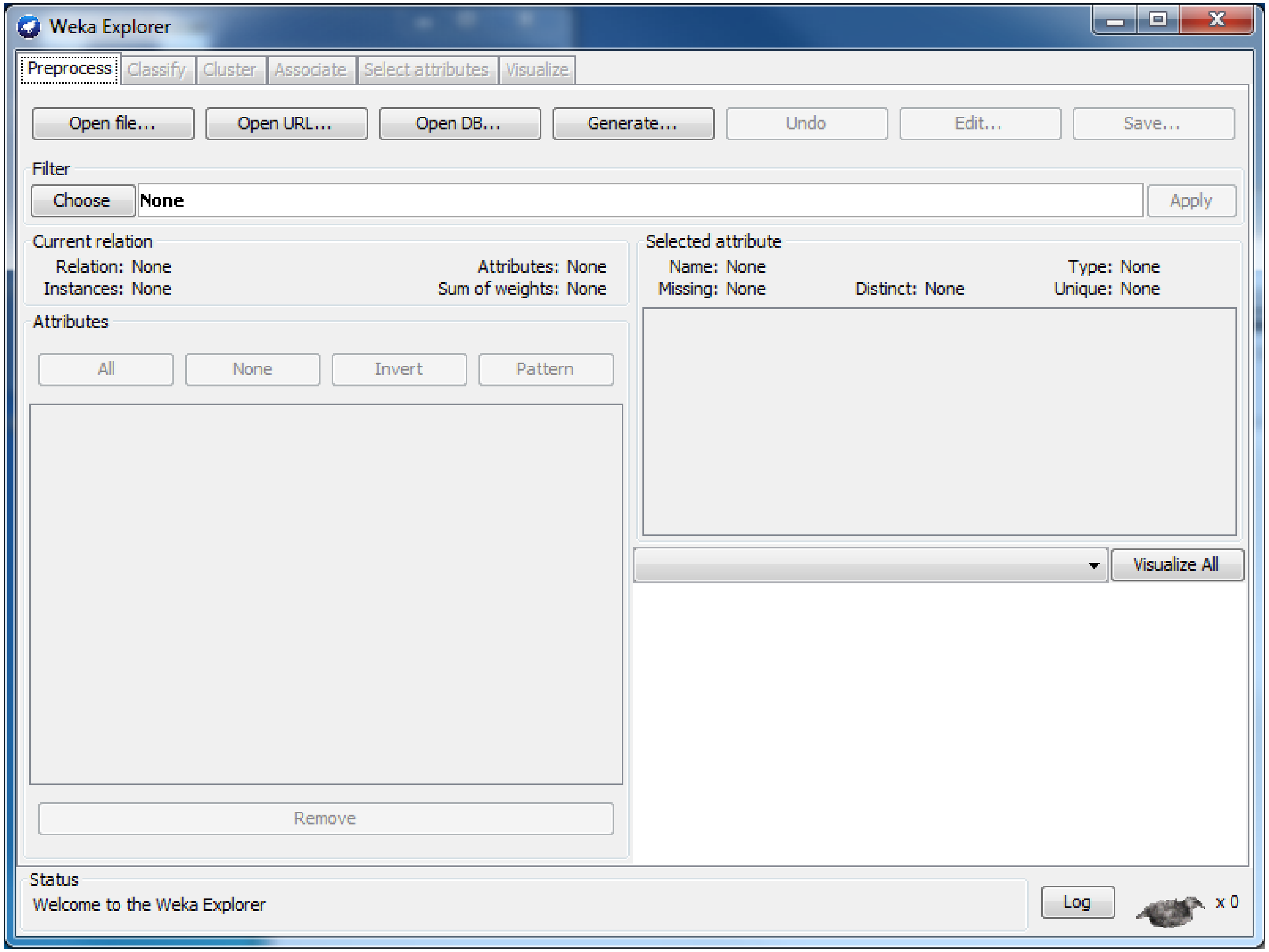
WEKA The University of Waikato

Data mining with WEKA

A use-case to help you get started

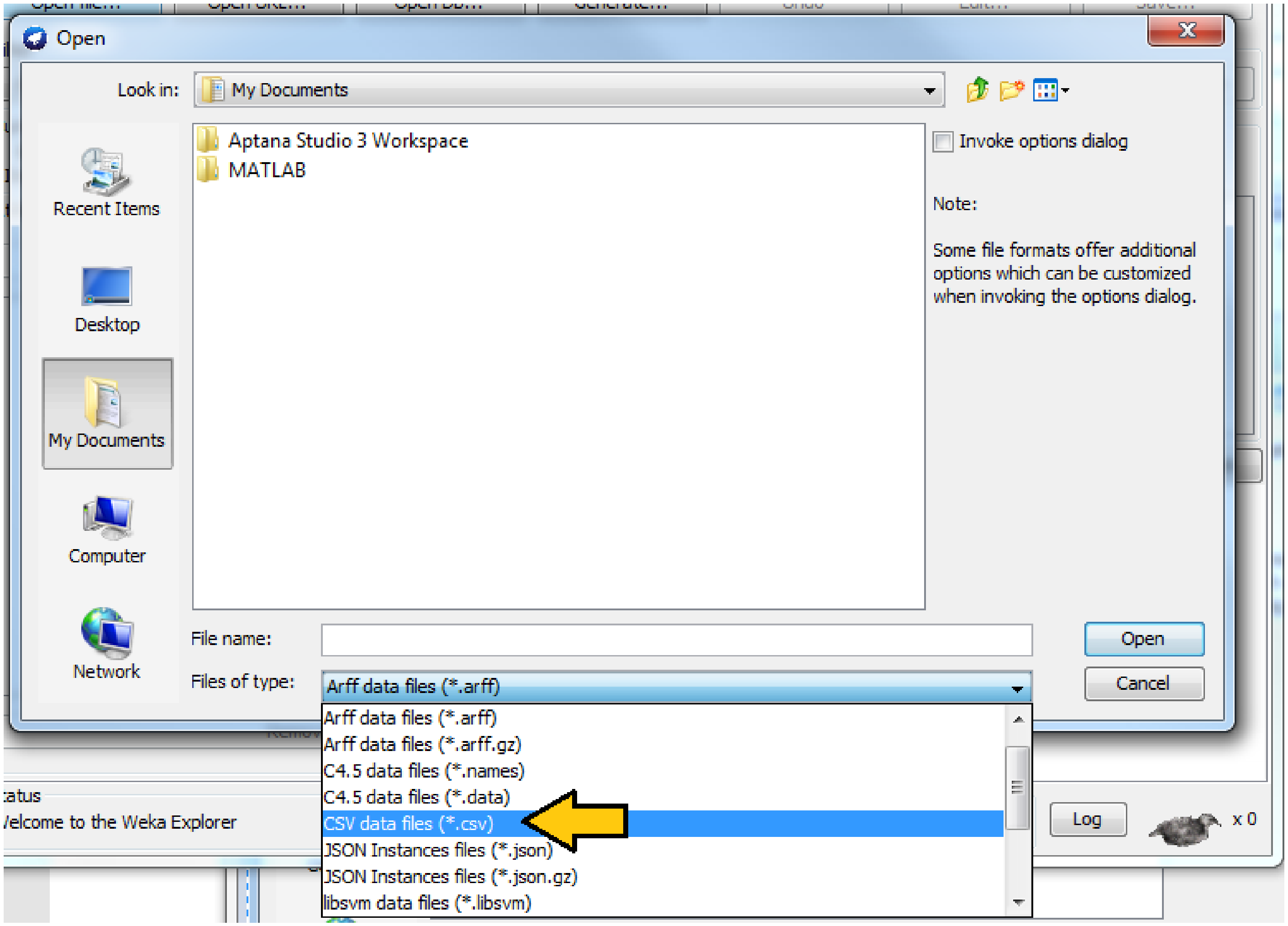
Charalampos Mavroforakis
BU CS105, Fall 2011

Starting WEKA



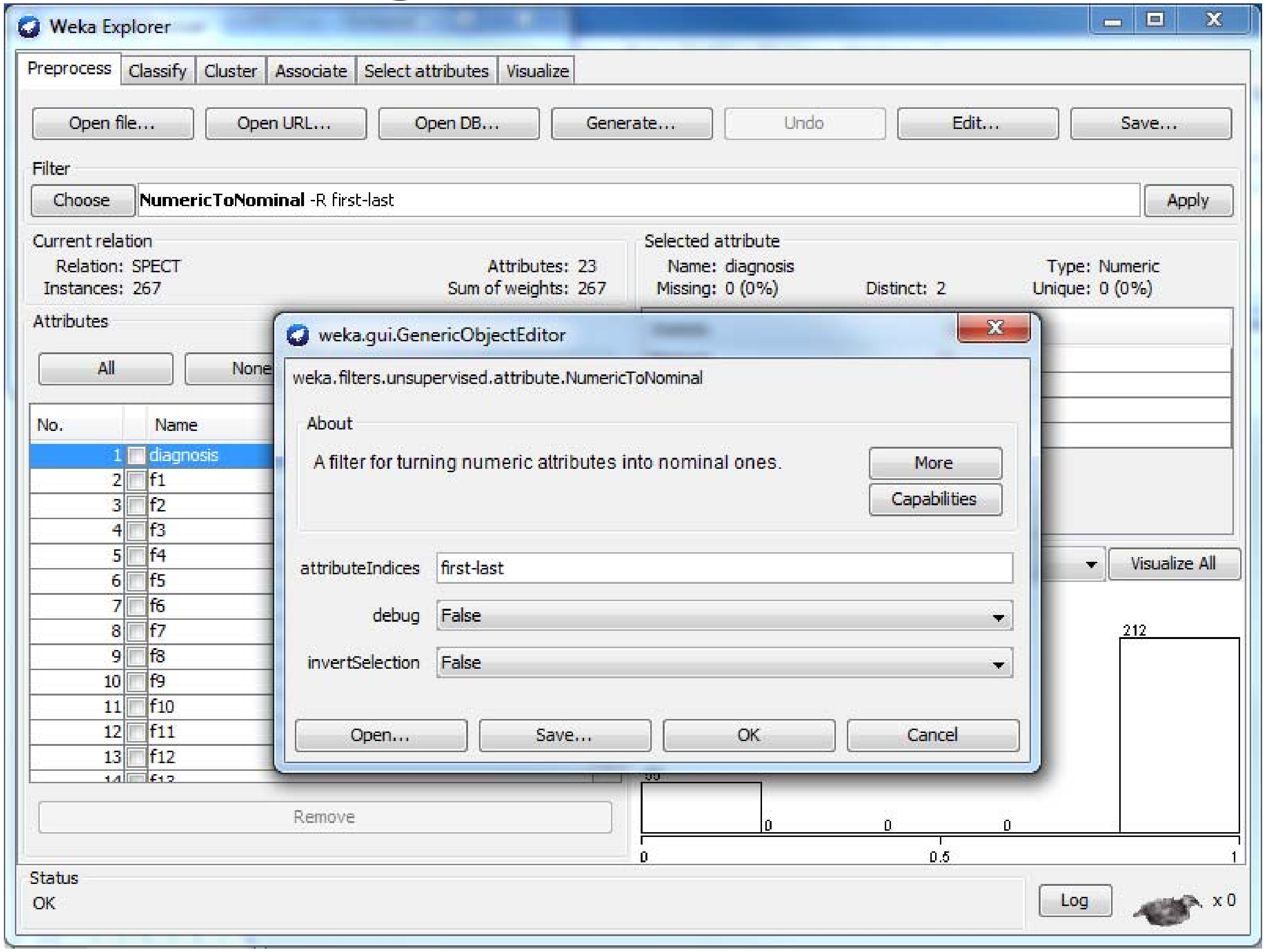
Open Weka : Start > All Programs > Weka 3.x.x > Weka 3.x
From the "*Weka GUI Chooser*", pick "*Explorer*". This is the main WEKA tool that we are going to use.

Opening a dataset



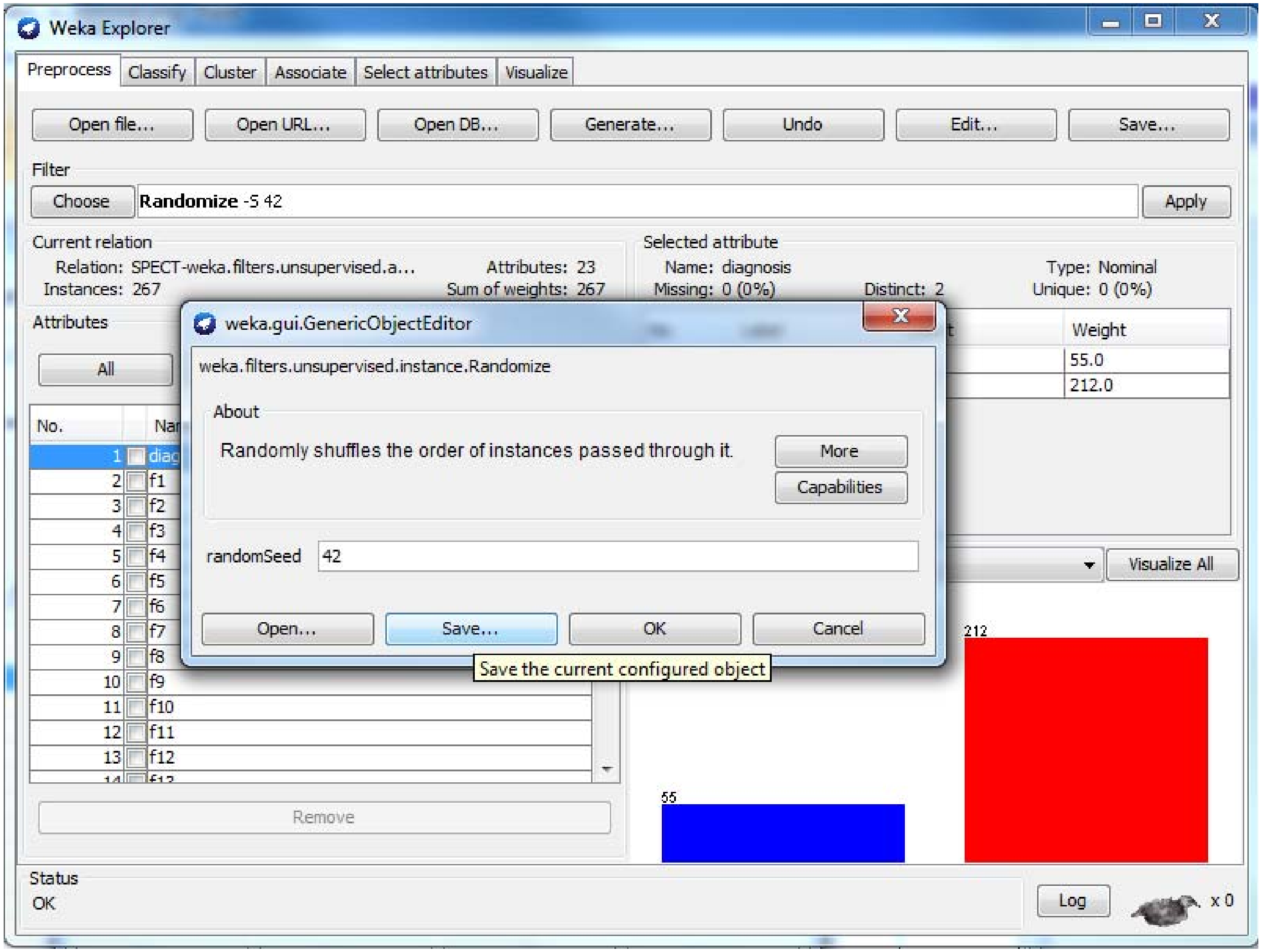
To open a dataset (a .csv file in our case), we click "Open file ..." in the *Preprocess* tab and open the file that contains our data. **Remember** that in the open menu you have to choose csv if your file was saved as such. Let’s open SPECT.csv

Transforming values to nominal (if needed)



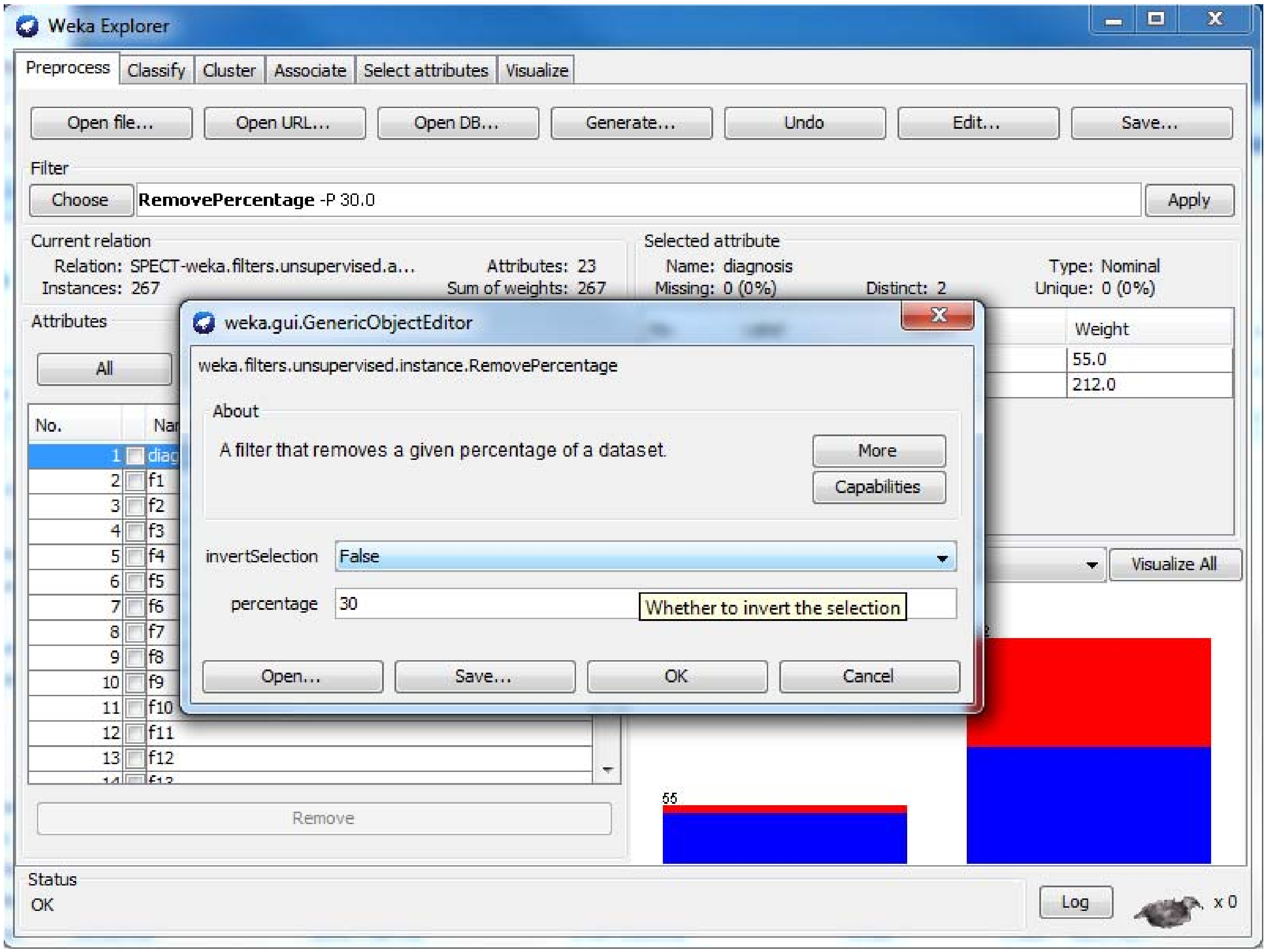
Weka classified every attribute in our dataset as numeric, so we have to manually transform them to nominal. To do so, we will use a filter. We navigate to *NumericToNominal*, which is in *Unsupervised > attribute*. If we click on that, we will get to the options of that filter. Mainly, the most interesting one here is the *attributeIndices*, which enumerates all the attributes that you want the filter to be applied on. To finish, we click *Apply.*

Splitting the dataset



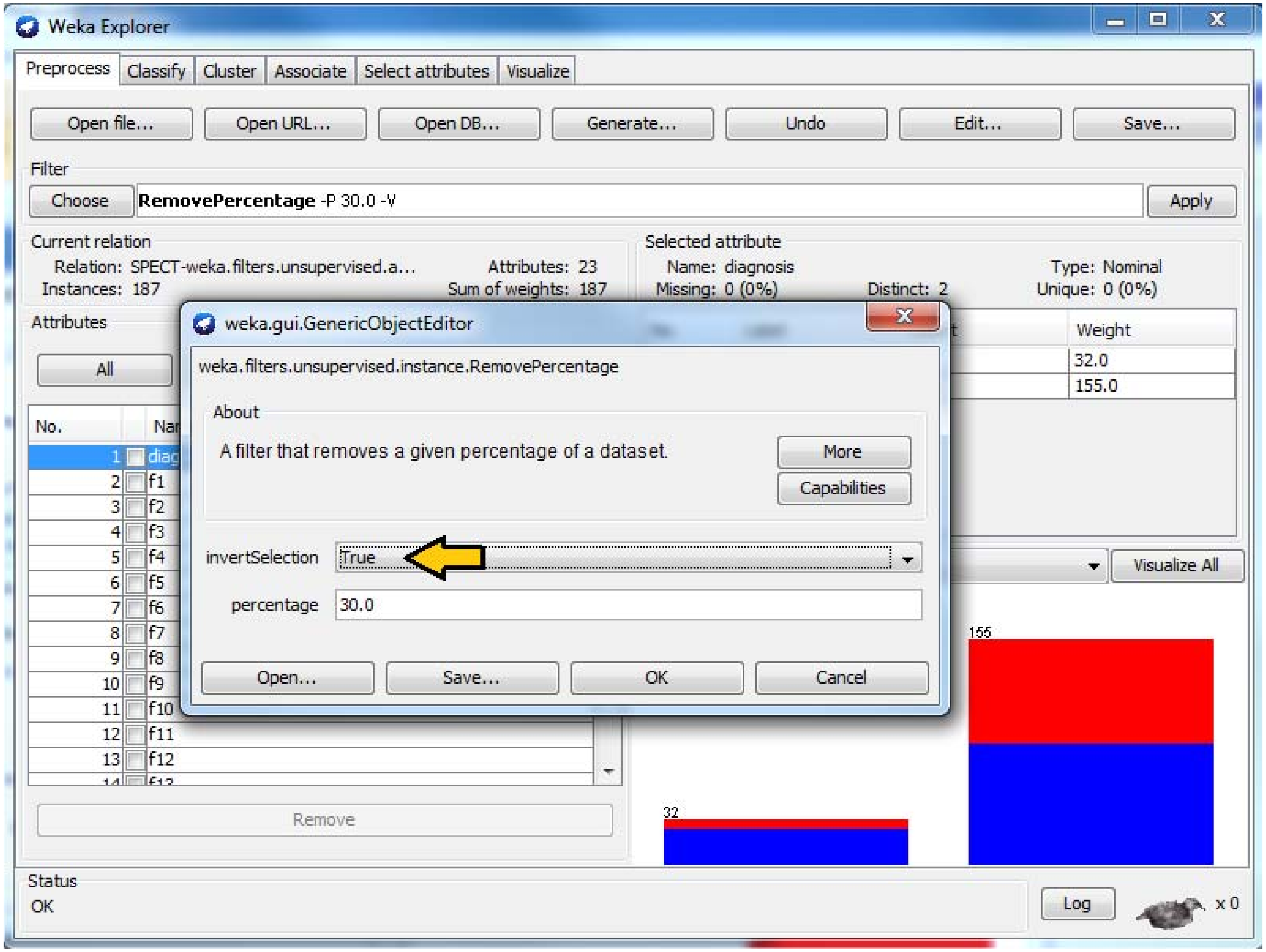
We have to split the dataset into two, 30% testing and 70% training. To do that, we first *Randomize* the dataset (Unsupervised > Instance), so that we create a random permutation.

Splitting the dataset



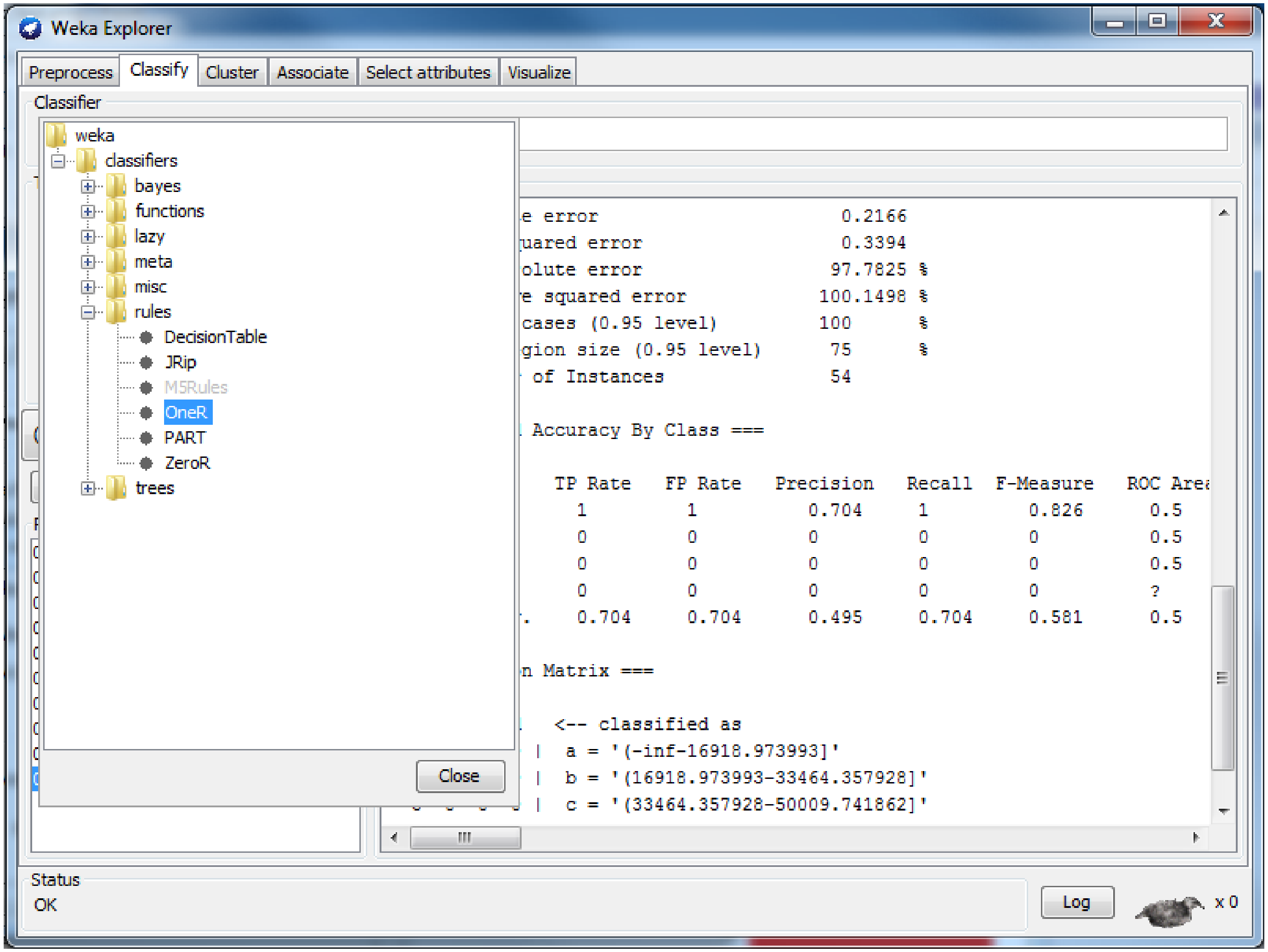
Then we apply *RemovePercentage* (Unsupervised > Instance) with percentage 30 and save the resulting dataset as training.

Splitting the dataset



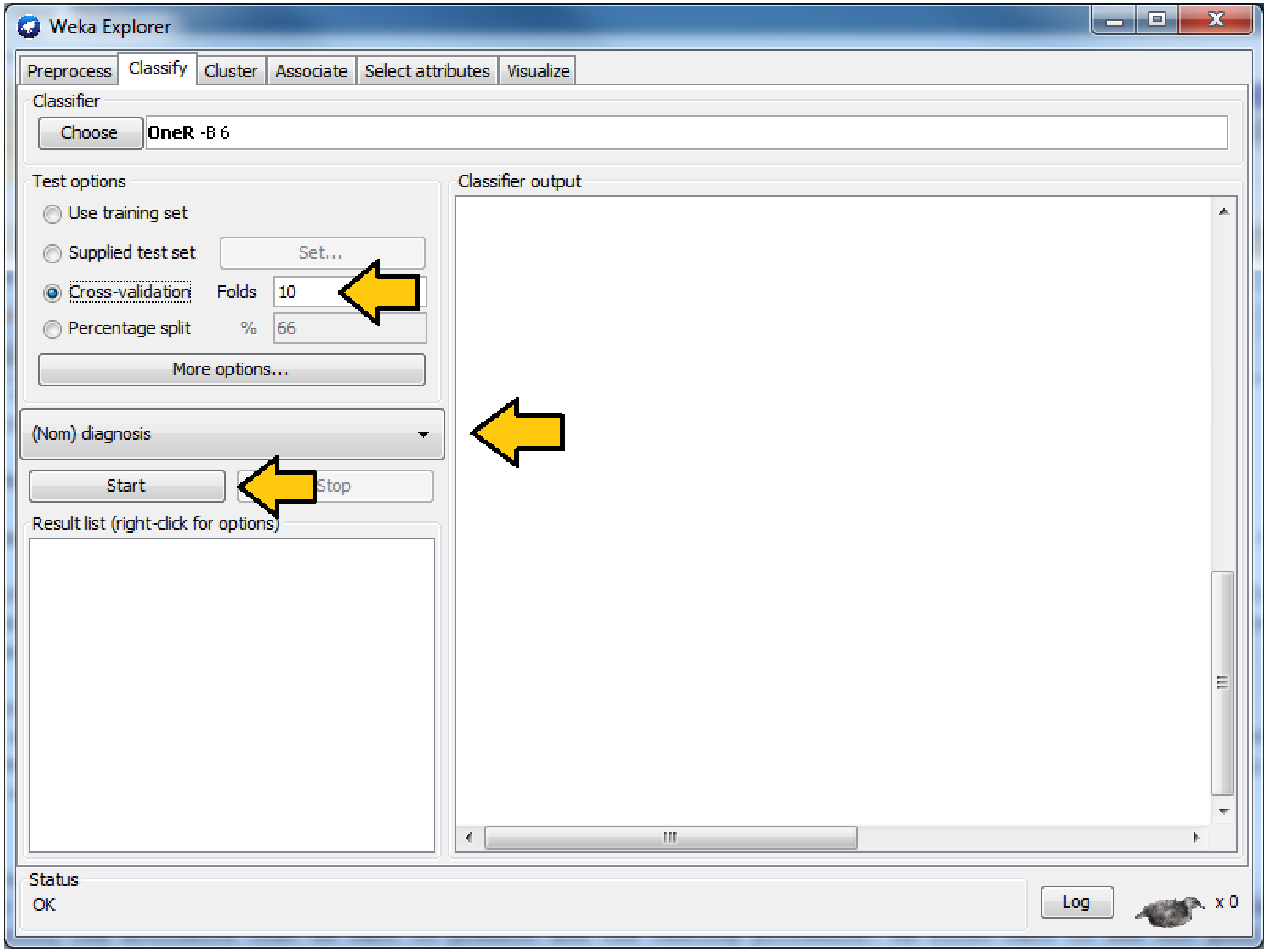
After that, we undo and apply the same filter choosing *invertSelection* this time. This will pick the rest of the data (30%) so we save them as the testing.

Training models



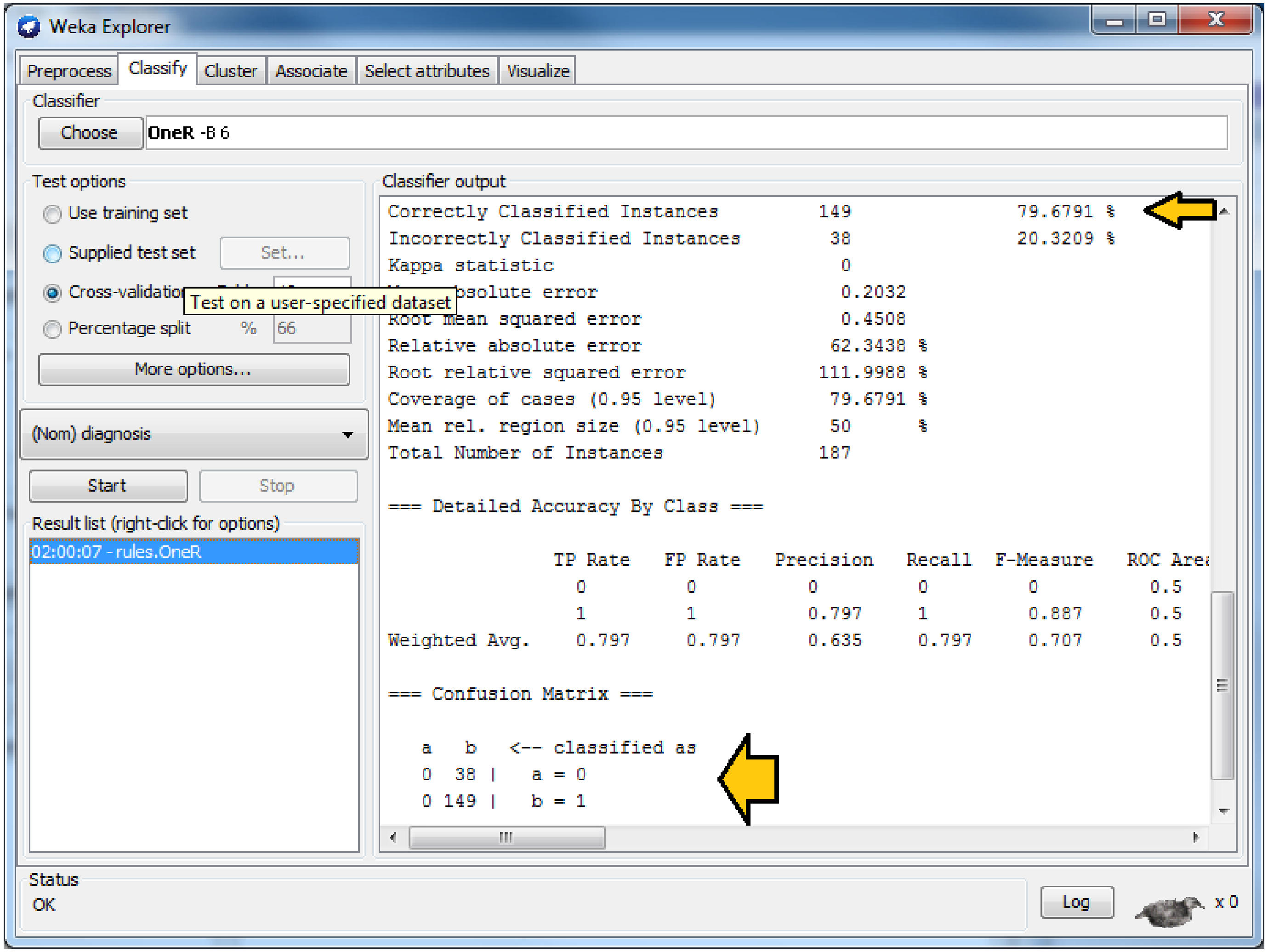
From now on we will be using the training dataset. We switch to the tab "*Classify*" and we pick a classifier. Let's start with *OneR*, which is the same with the one we saw in the class.

Training models



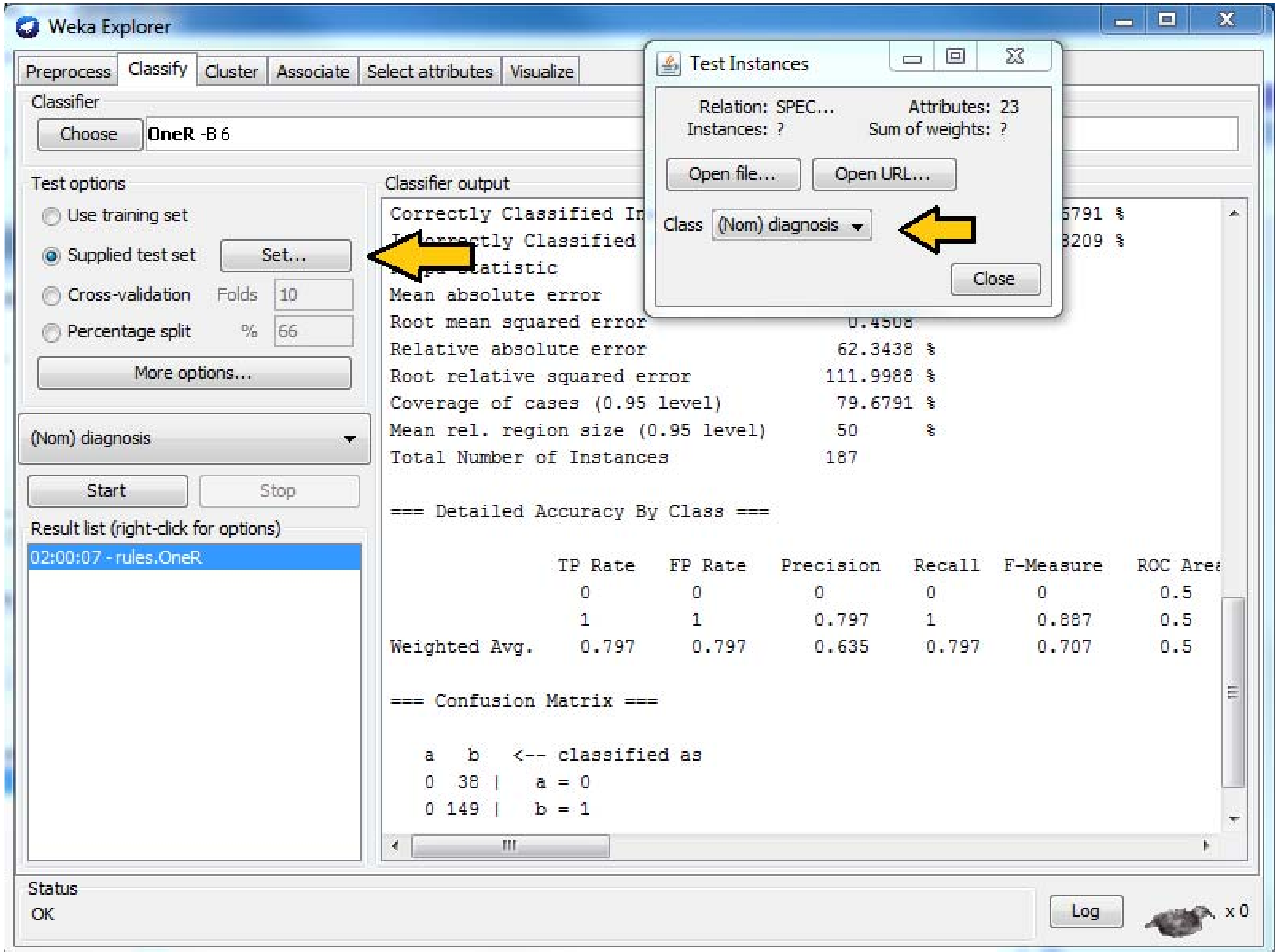
We have to specify the attribute that we want to predict and the testing procedure. We first want to see how good OneR is as a model, so we use cross-validation. , and only after that will we go and check what it predicts on the unseen data.

Training models



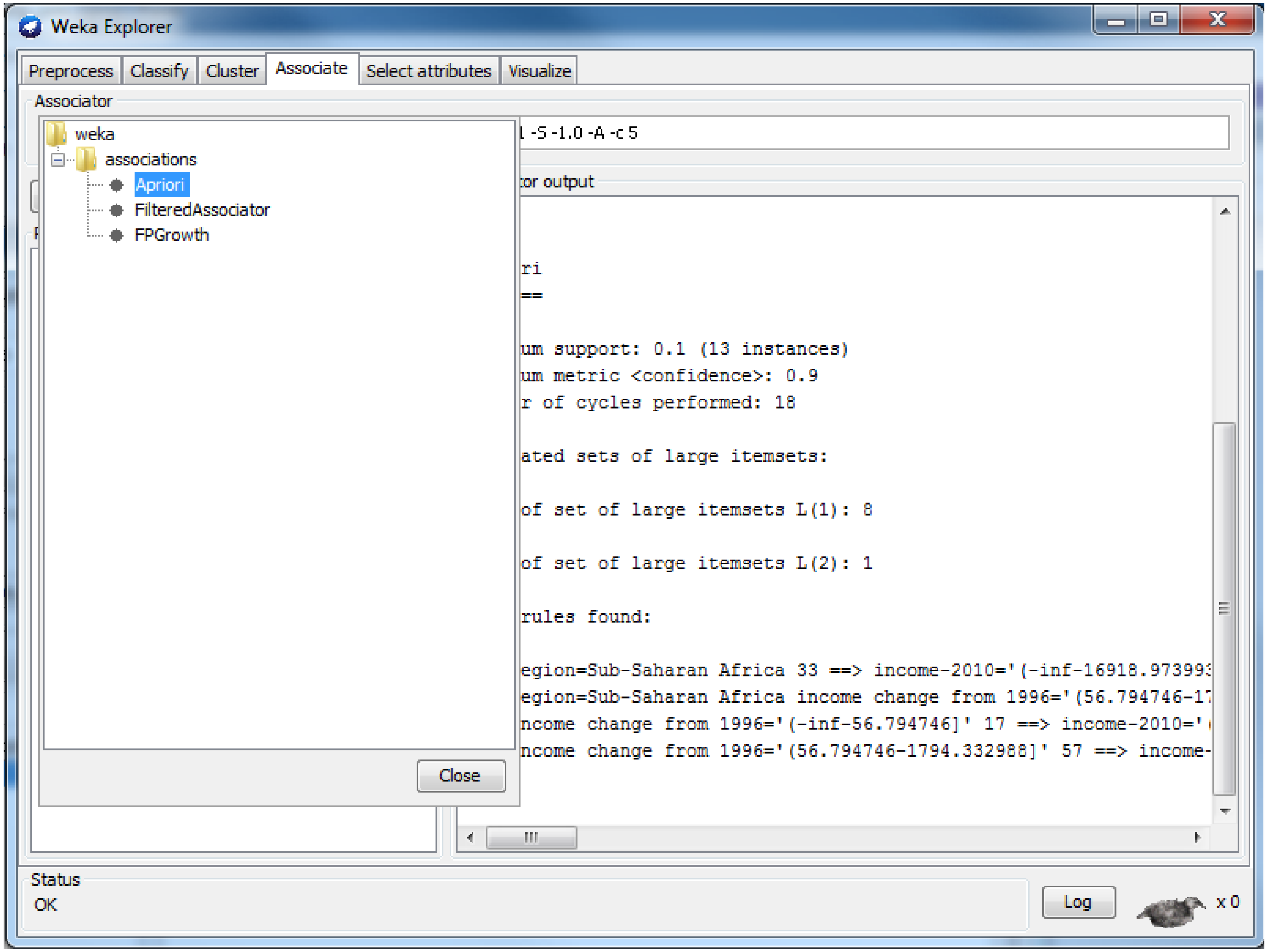
In the output, we get information about the average accuracy and the confusion matrix of our model.

Training models



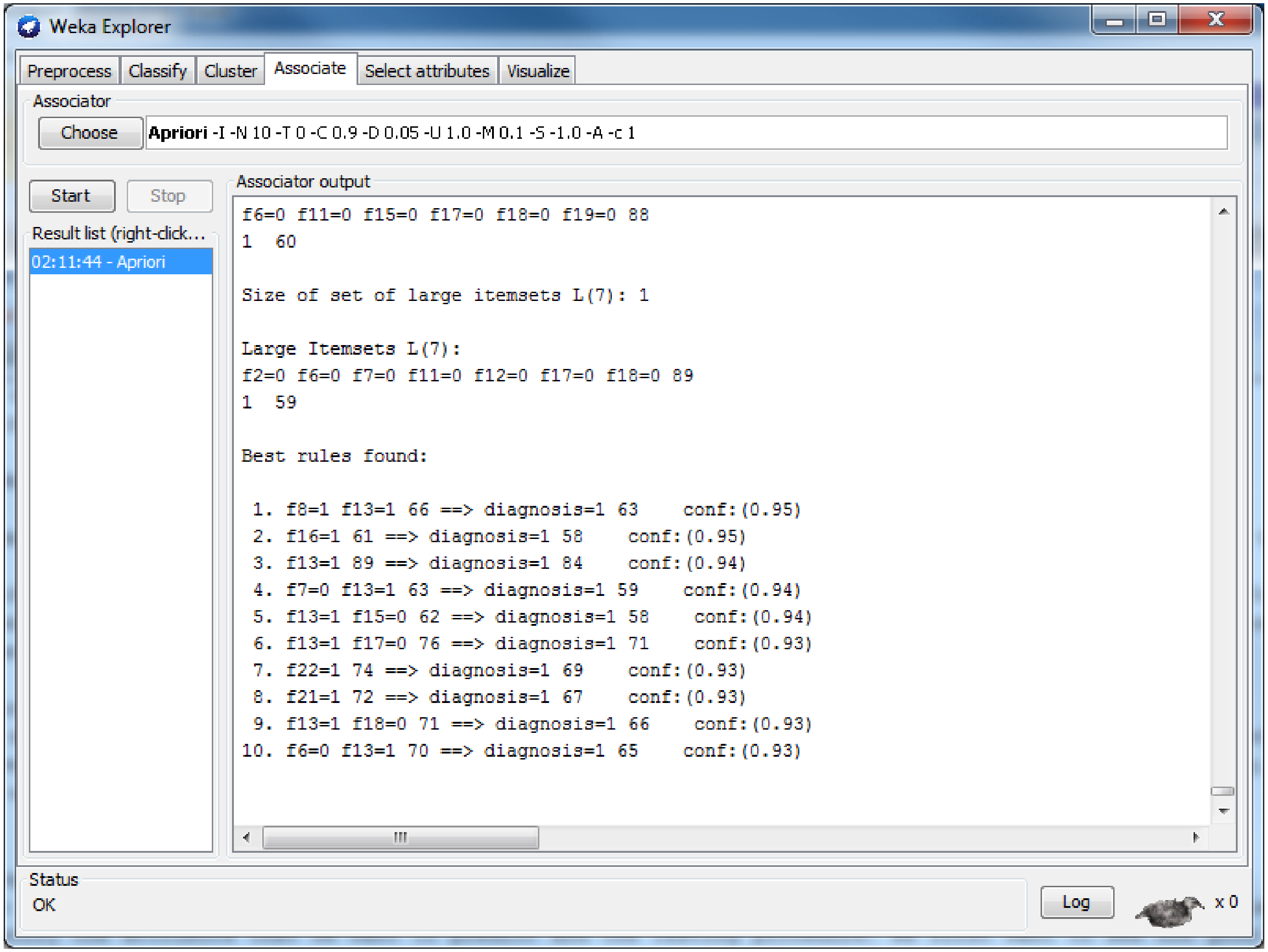
In order to check how well we do on the unseen data, we select "*supplied test set*" ,we open the testing dataset that we have created and we specify which attribute is the class. We run the algorithm again and we notice the differences in the confusion matrix and the accuracy.

Association learning



If all of our attributes are nominal (in case they are not, we can discretize them in the *Preprocess* tab) we can also do association learning. In order to do that, we switch to the *Association* tab and we choose the *Apriori* algorithm. You can play around with its parameters if you want.

Association learning



We could set *car* to True (so that it produces rules that predict the class attribute) and specify the index of the attribute that will be considered as class. *minMetric* sets the threshold of confidence and *numRules* limits the number of rules that will be created. The result will be a set of rules that predict the class, together with their confidence.