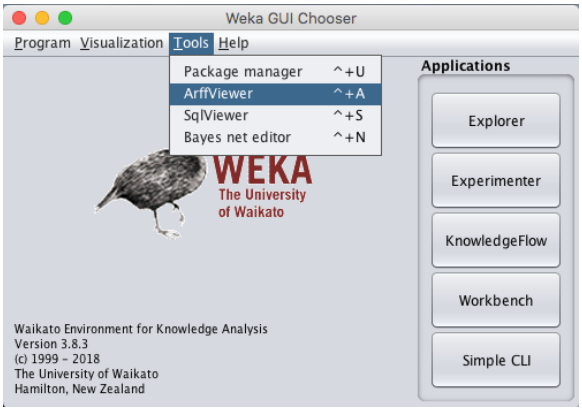
Convert CSV to Arff

An ARFF (Attribute-Relation File Format) file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files were developed by the Machine Learning Project at the Department of Computer Science of The University of Waikato for use with the Weka machine learning software.

To convert your csv file to arff go in Tools -> (ArffViewer or Ctrl+A). Then open your CSV file.



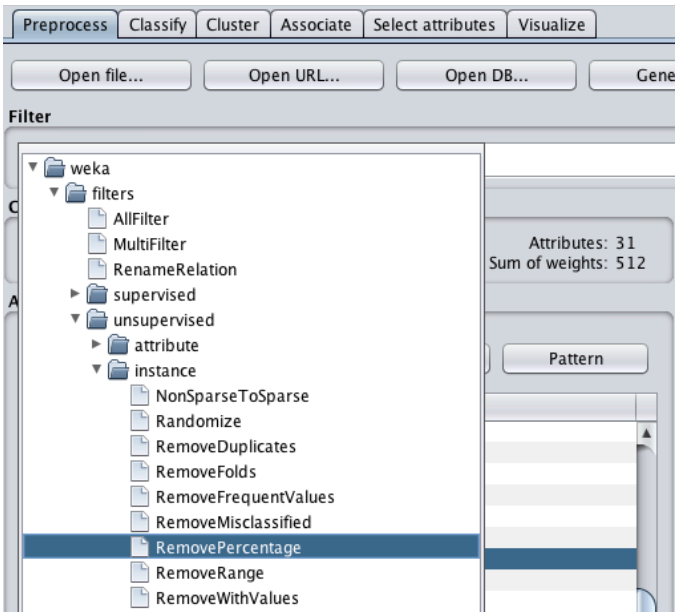
Next, go to File -> Save as... and select *Arff data files* (should be selected by default).

PS.: Note that your fields in csv files must be separated with a comma **“,”** and not a semicolon “;”.

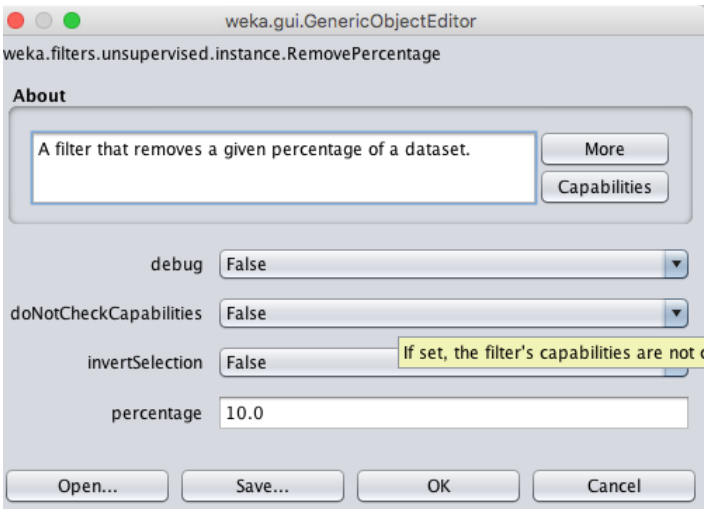
Reduce the Cases

Gets a reduced representation of the data series that is much smaller in volume but yet produces the same (or almost the same) analytical results.

Filter -> Unsupervised -> Instance -> RemovePercentage



Choose the percentage of reduction

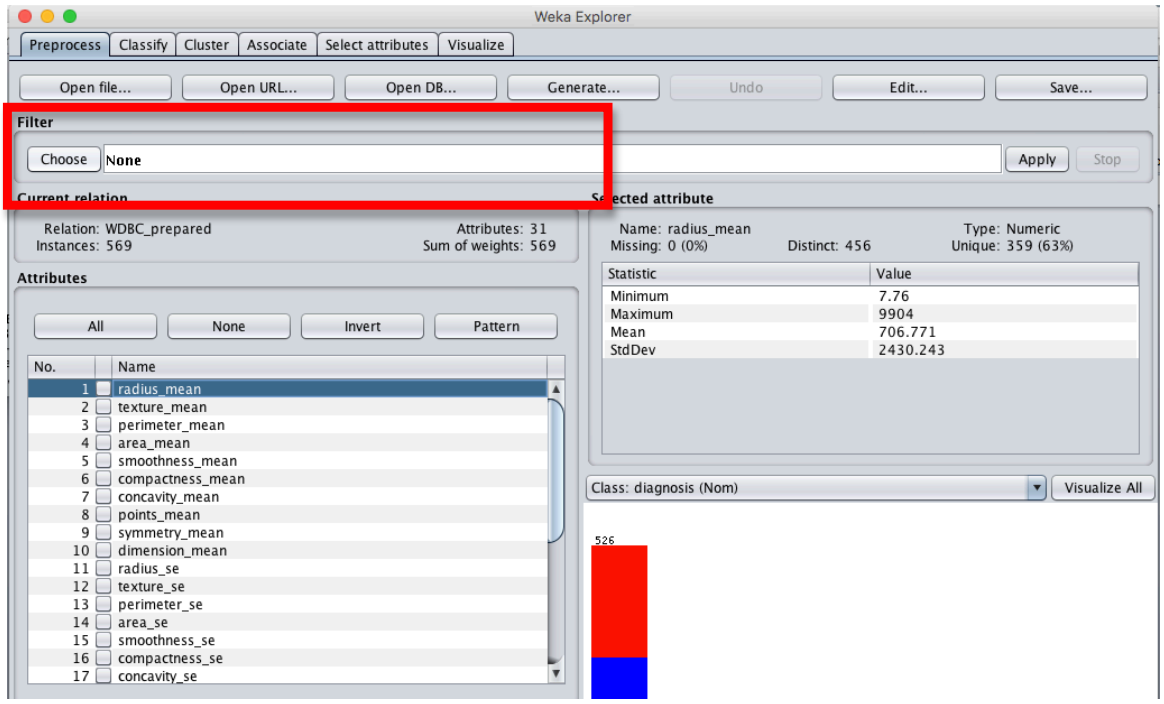


Convert Nominal to Numeric

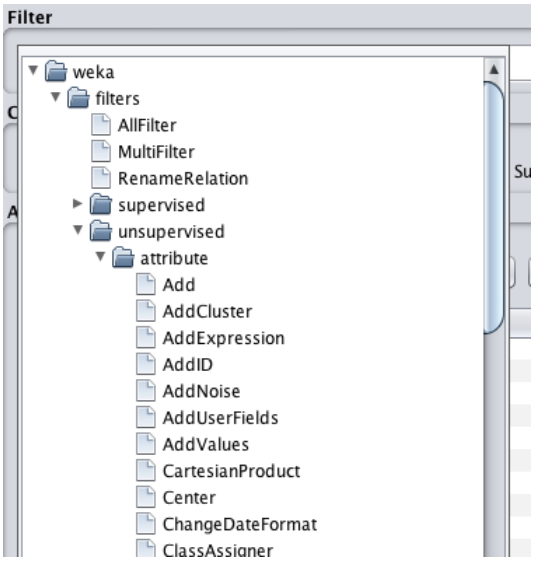
A nominal attribute can take on two or more states . For example, colors are a nominal attribute that may have five states: RED, YELLOW, GREEN, BLUE, and BLACK.

Converting nominal attributes to Numeric in Weka

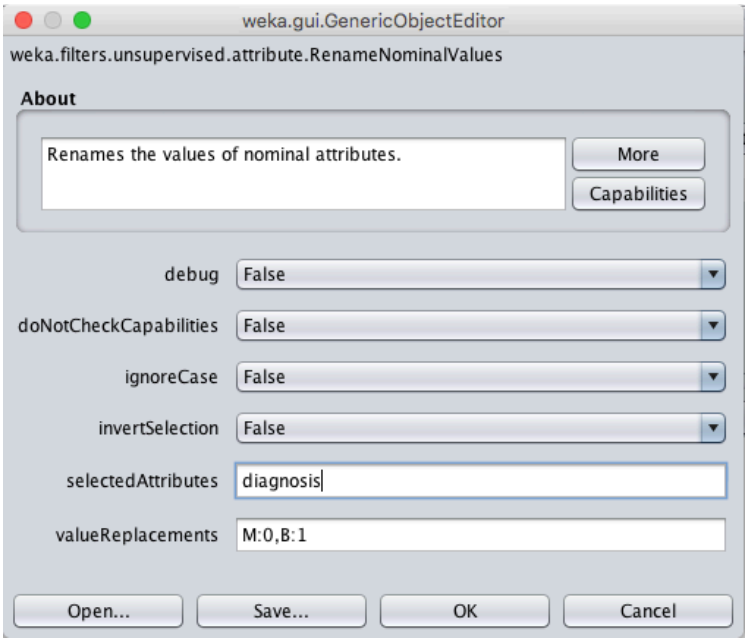
Open arff file, choose filter.



Choose the filter weka.filters.unsupervised.atribute.RenameNominalValues.



Click two times in the filter. It will open a new window. In selectedAtrributes type the name of attribute, “diagnosis” and in the valueReplacements, define the actual value and the new one of each instance value.

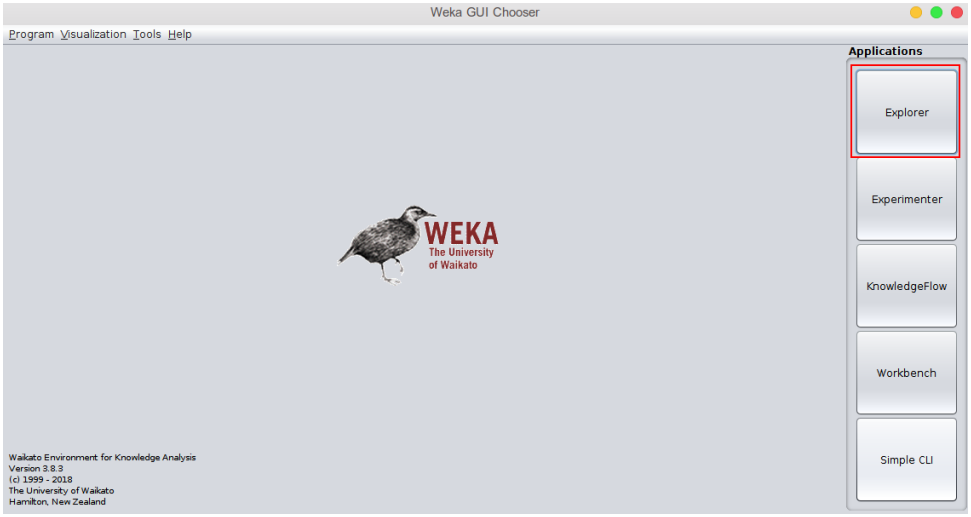


OK and Apply.

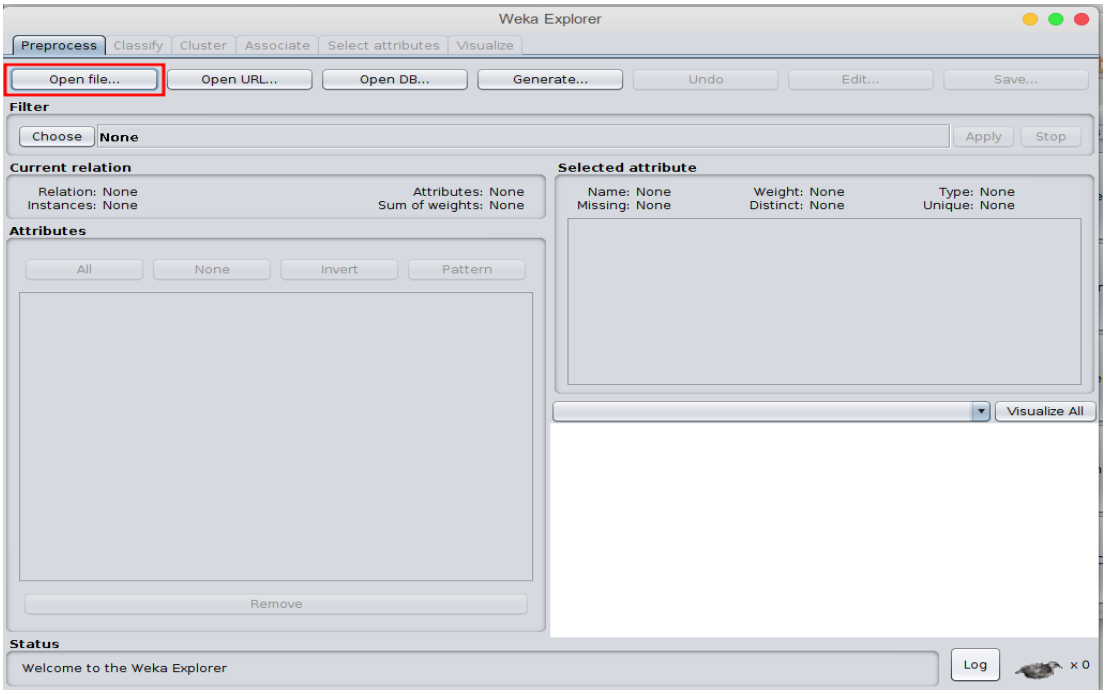
Font: https://stackoverflow.com/questions/27121001/how-to-perform-nominal-to-numeric-conversion-of-attributes-in-weka

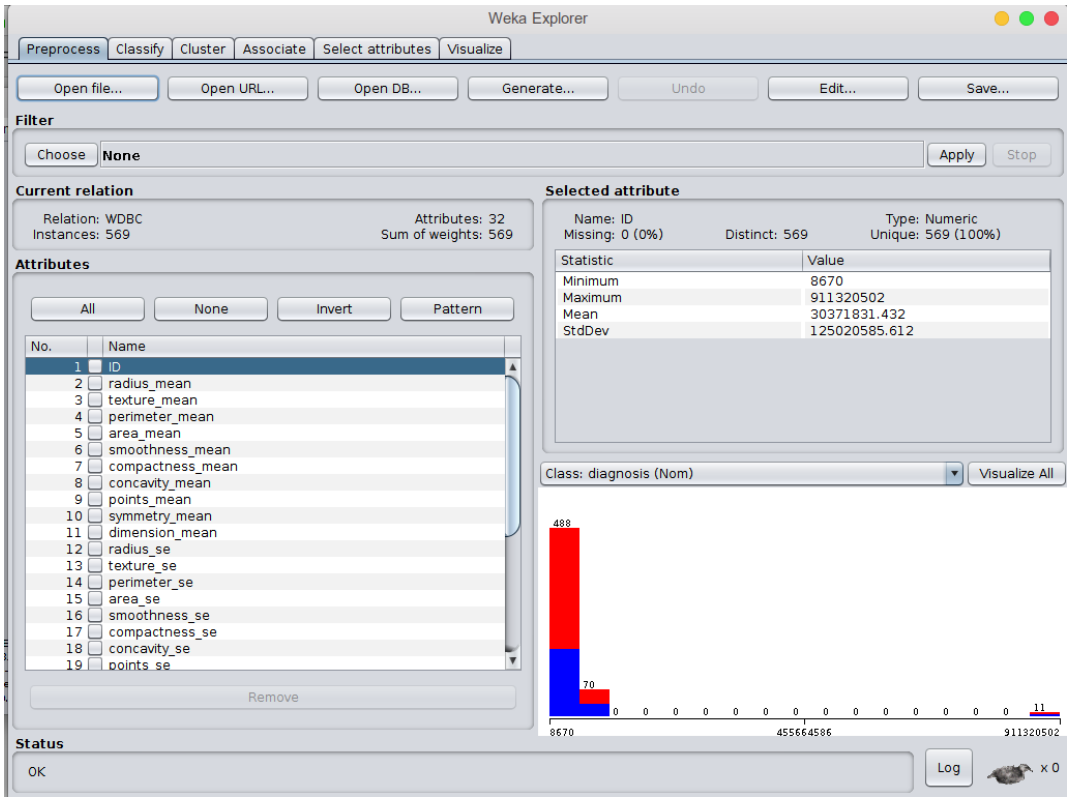
Normalize and Standardize Your Machine Learning Data in Weka

Normalization is useful when your data has varying scales and the algorithm you are using does not make assumptions about the distribution of your data, such as k-nearest neighbors and artificial neural networks.

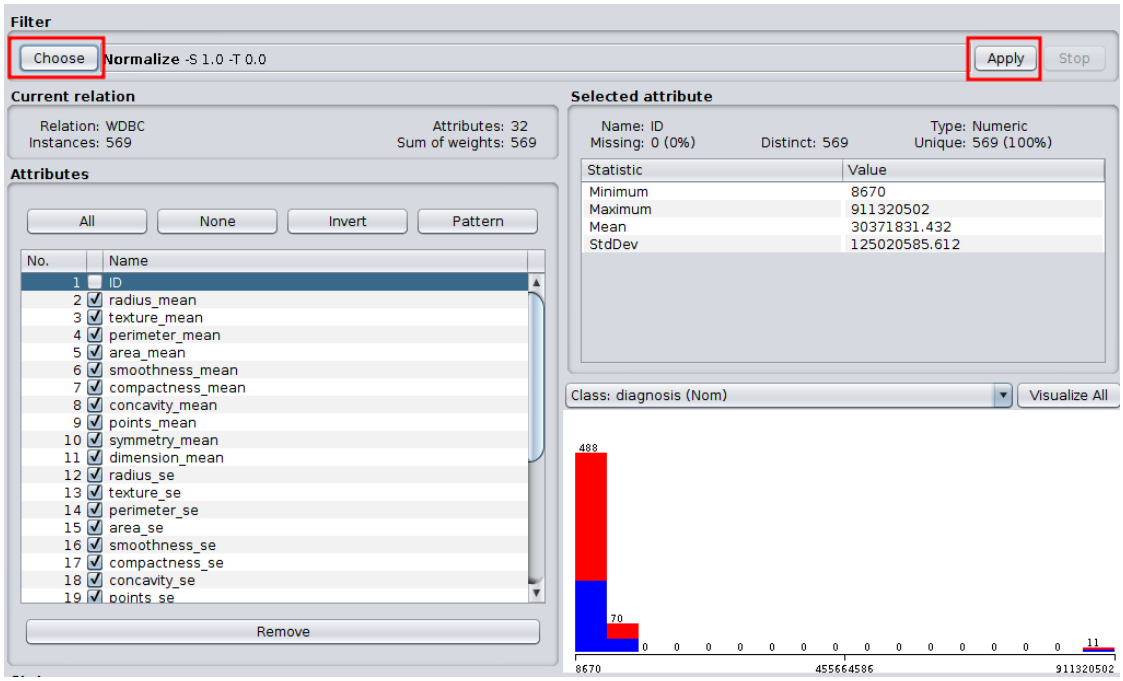


Open the Arff file.


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Stop
Current relation
Relation: None
Instances: None
Attributes: None
Sum of weights: None
Selected attribute
Name: None
Missing: None
Weight: None
Distinct: None
Type: None
Unique: None
Attributes
All
None
Invert
Pattern
Remove
Visualize All
Status
Welcome to the Weka Explorer
Log
x 0


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Stop
Current relation
Relation: WDBC
Instances: 569
Attributes: 32
Sum of weights: 569
Selected attribute
Name: ID
Missing: 0 (0%)
Distinct: 569
Type: Numeric
Unique: 569 (100%)
Statistic
Value
Minimum
8670
Maximum
911320502
Mean
30371831.432
StdDev
125020585.612
Attributes
All
None
Invert
Pattern
No.
Name
1 ID
2 radius_mean
3 texture_mean
4 perimeter_mean
5 area_mean
6 smoothness_mean
7 compactness_mean
8 concavity_mean
9 points_mean
10 symmetry_mean
11 dimension_mean
12 radius_se
13 texture_se
14 perimeter_se
15 area_se
16 smoothness_se
17 compactness_se
18 concavity_se
19 points_se
Remove
Class: diagnosis (Nom)
Visualize All
488
70
11
8670
455664586
911320502
Status
OK
Log
x 0

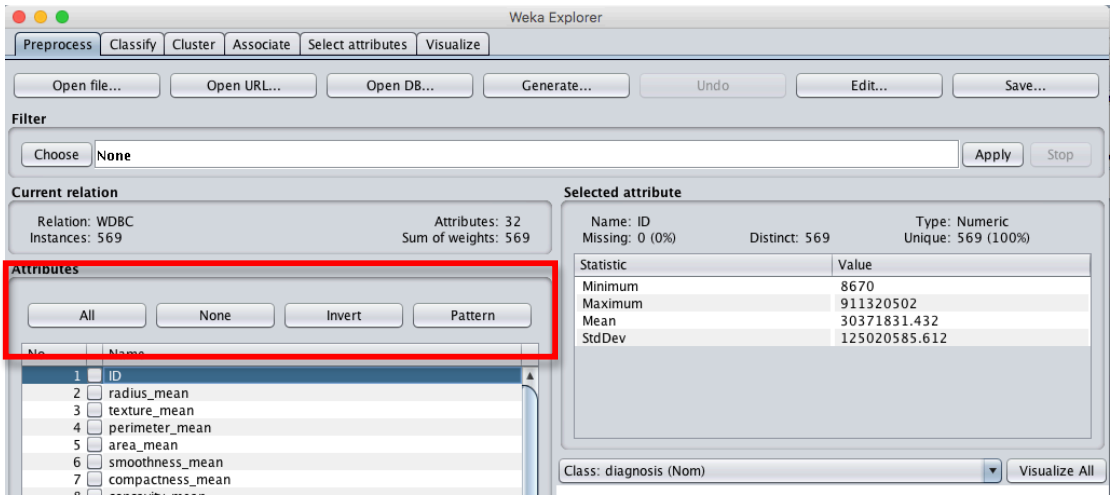
Click the “Choose” button to select a Filter and select ***unsupervised.attribute.Normalize.***



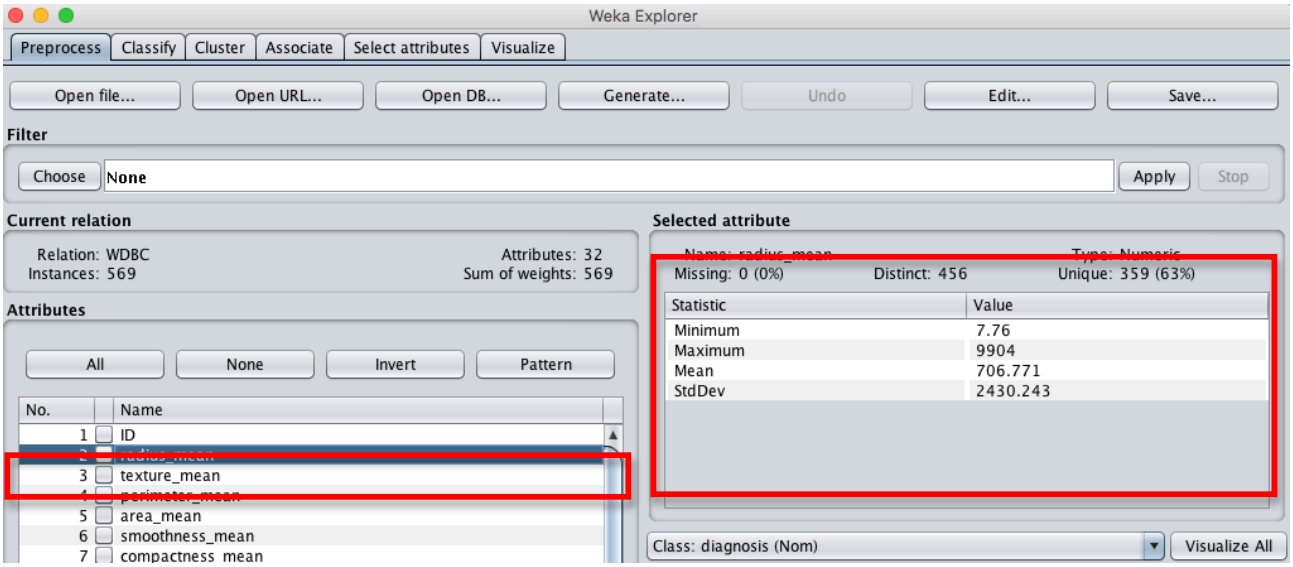
Click the “Save” button and type a filename to save the normalized copy of your dataset.

Better Understand Your Data With Descriptive Statistics

Firstly, note that the dataset summary in the “Current Relation” section. This panel summarizes the following details about the loaded datasets:



Click on one attribute in the dataset in the “Attributes” panel.

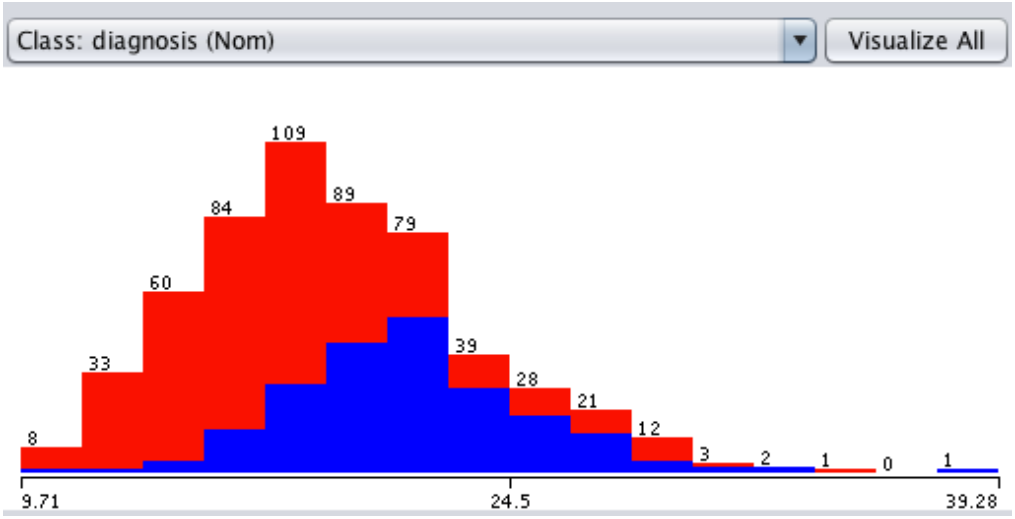


You can learn a lot from this information. For example:

The presence and ratio of missing data can give you an indication of whether or not you need to remove or impute values.
The mean and standard deviation give you a quantified idea of the spread of data for each attribute.
The number of distinct values can give you an idea of the granularity of the attribute distribution.

Univariate Attribute Distributions

The distribution of each attribute can be plotted to give a visual qualitative understanding of the distribution.



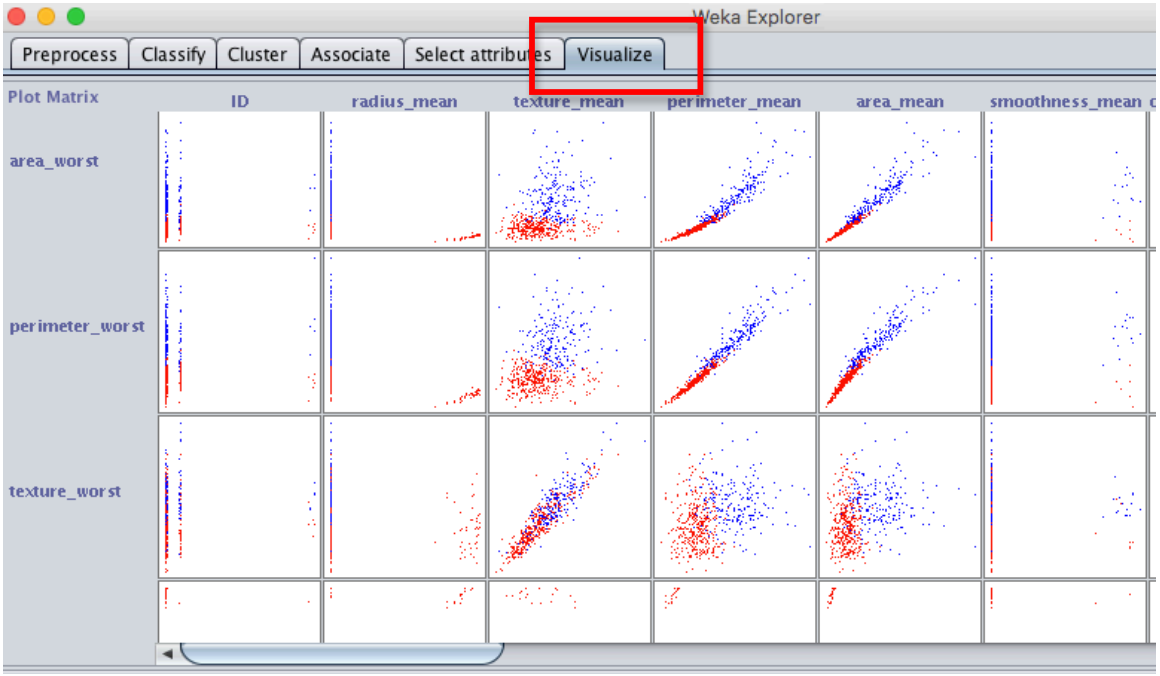
You will see the distribution of preg values between 9.71 and 39.28 along the x-axis. The y-axis shows the count or frequency of values with each texture-mean value.

Note the red and blue colors referring to the e classes of câncer dataset classification. **Malignant** and **Benign** classes respectively. The colors are assigned automatically to each categorical value.

This is useful to get a quick idea of whether the problem is easily separable for a given attribute, e.g. all the red and blue are cleanly separated for a single attribute.

Visualize Attribute Interactions

When attributes are numeric we can create a scatter plot of one attribute against another. This is useful as it can highlight any patterns in the relationship between the attributes, such as positive or negative **correlations**.



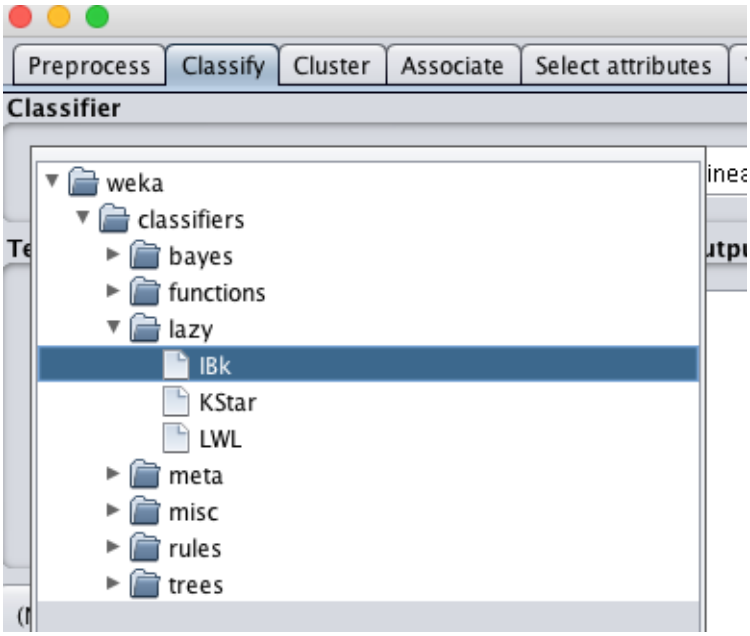
Classification Supervised Models on Weka

K-NN

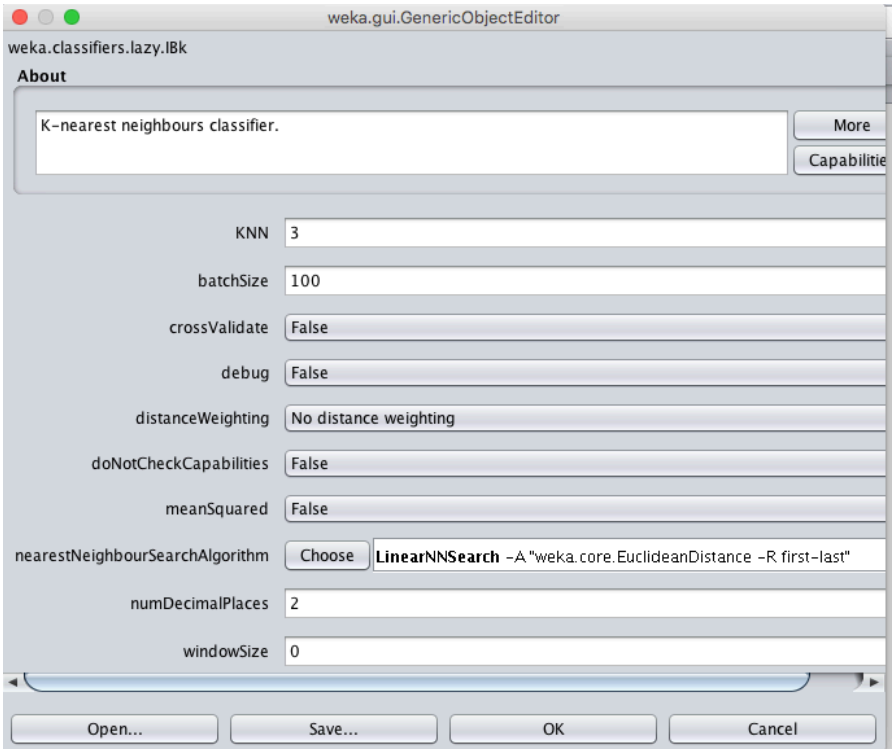
Simple instance-based learner that uses the class of the nearest k training instances for the class of the test instances.

the “IB” stands for Instance-Based, and the “k” allows us to specify the number of neighbors to examine

In the tab “Classify” choose lazy -> IBk

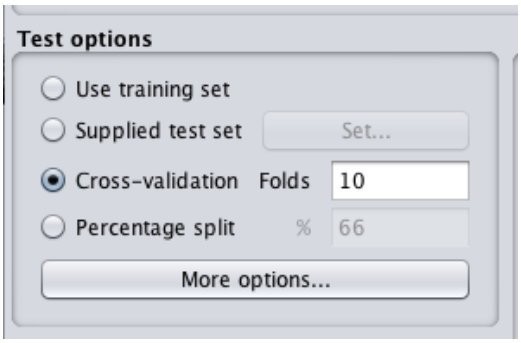


Double click on the classify to configure it properties



In bellow of the classify type, choose the type of separation of data set of training and test.

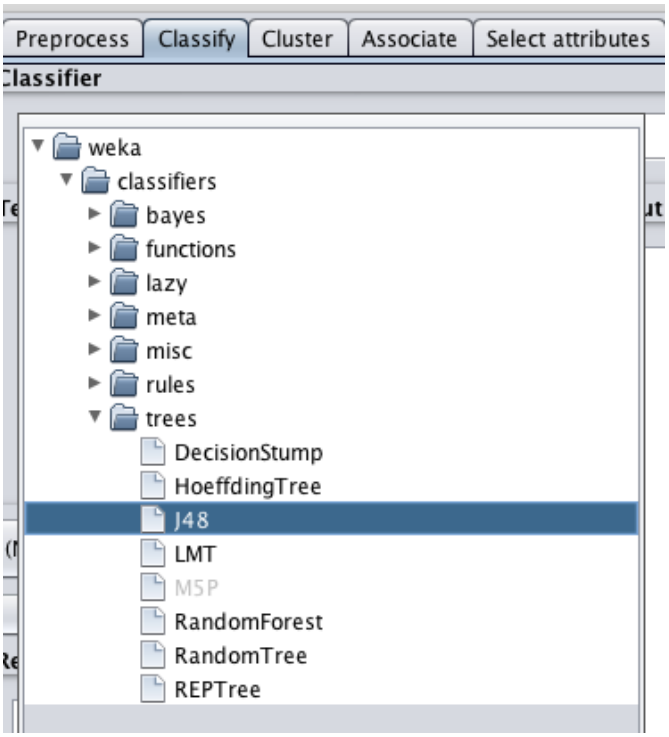
In this case, 10-fold, that consists of dividing the total data set into 10 mutually exclusive subsets of the same size. A subset is used for testing and the remaining 9 are used for parameter estimation and model accuracy is calculated



And click in **start** button.

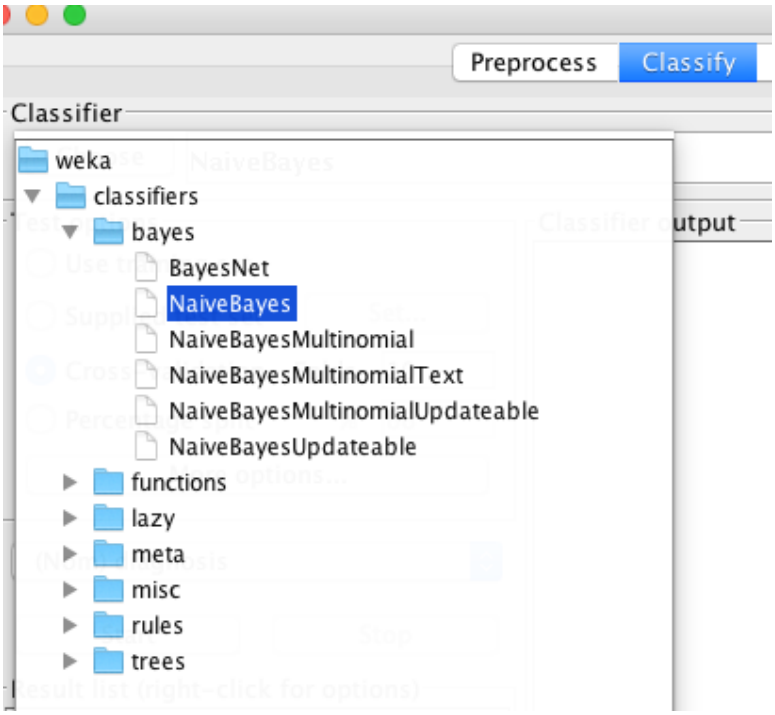
Decision Tree

Decision tree J48 is the implementation of algorithm ID3 (Iterative Dichotomiser 3) developed by the WEKA project team. R includes this nice work into package RWeka



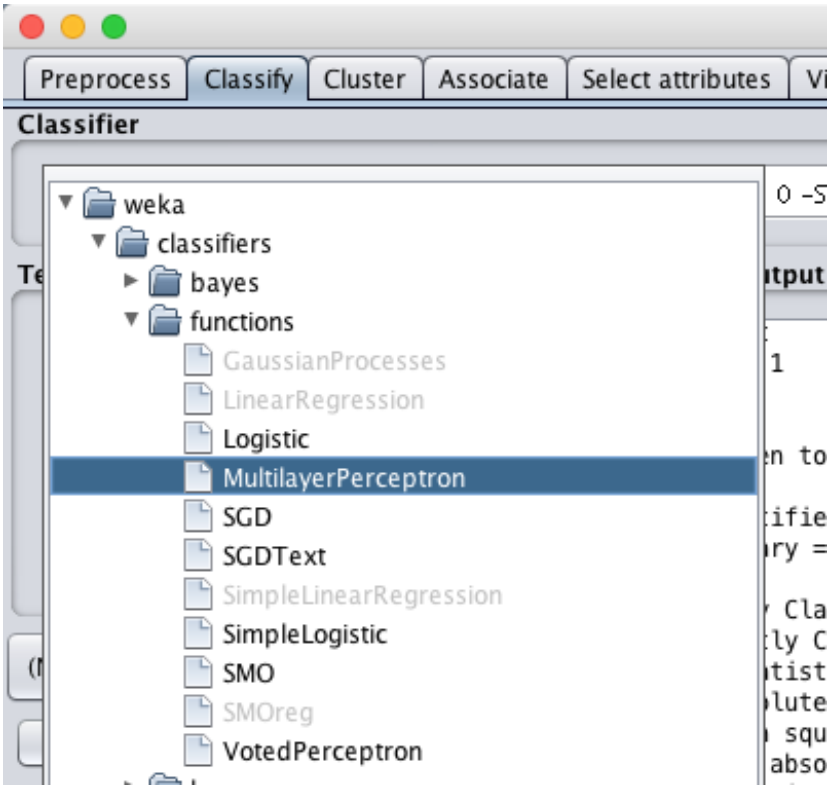
Naive Bayes

In the tab “Classify” choose bayes -> NaiveBayes

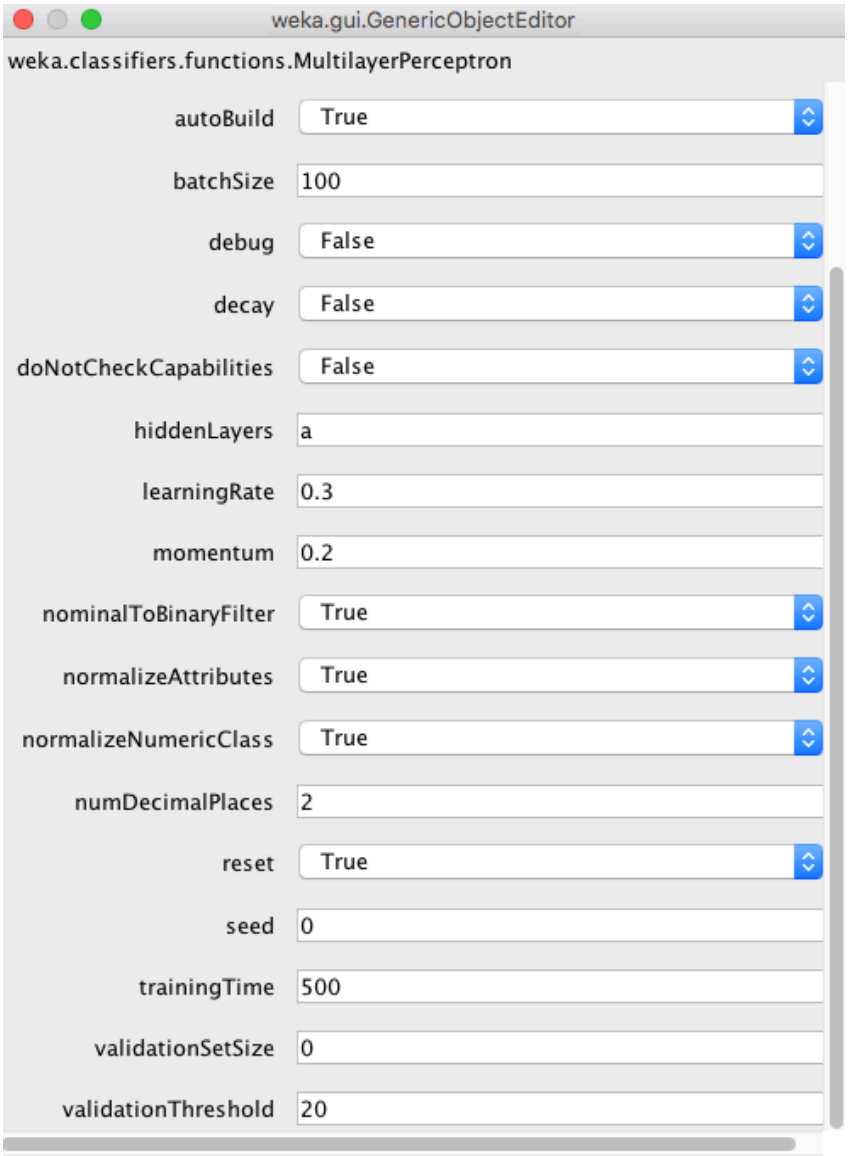


Neural Network

Classify -> Multilayer Perceptron



Double click on the select classify to configure it properties



hiddenLayers = number of neurons in the hiddenLayers. This is a list of positive whole numbers. 1 for each hidden layer. Comma separated. To have no hidden layers put a single 0 here. To have 3 layers with 10, 20 and 10 neurons, put “10,20,10” in this field.

seed = seed to initialize the weights

trainingTime = Number of epochs that the network will execute

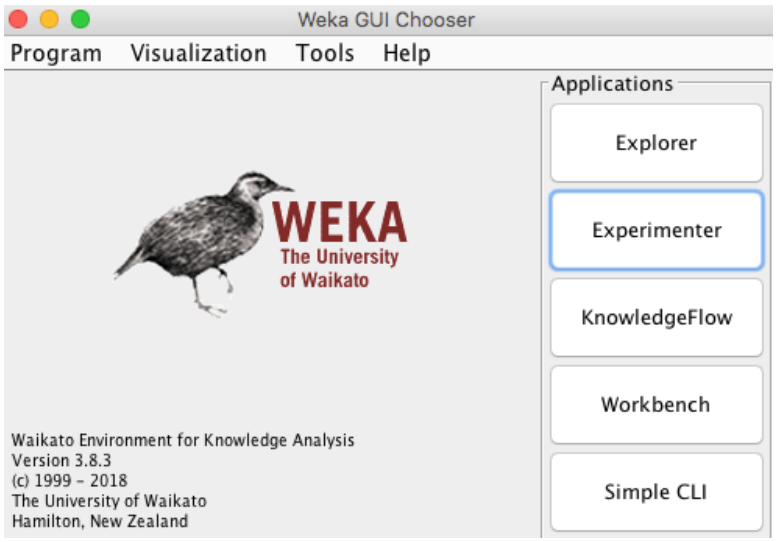
Ensemble in the Weka

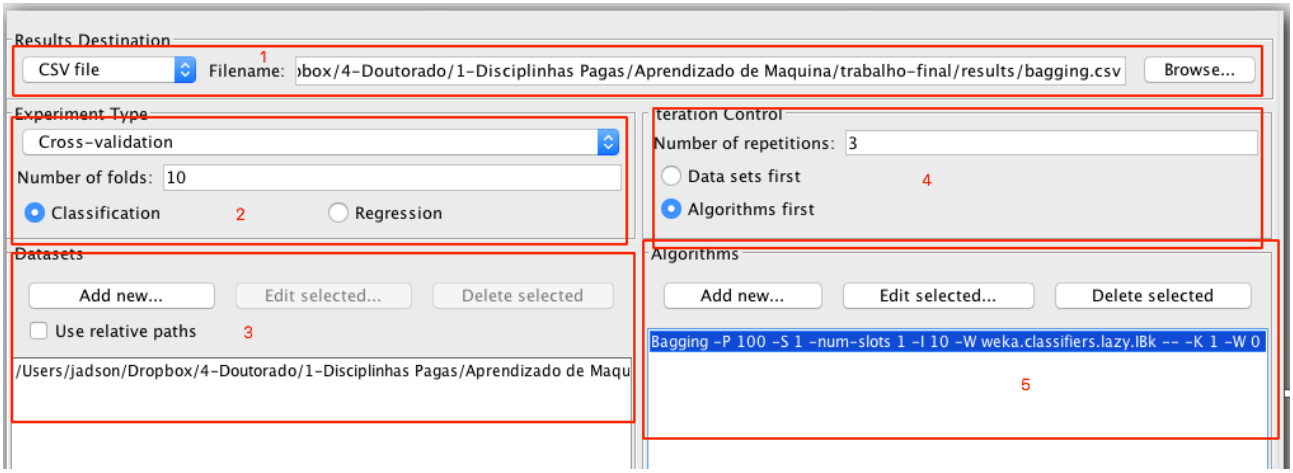
Ensemble algorithms are a powerful class of machine learning algorithm that combine the predictions from multiple models.

In weka Ensemble are also calling as “meta algorithms”

Open the **experimenter** window**.**

Experimenter allow you to run several processes and save the results





1 – The file where the results will be save

2 – Resampling method of the input data (training and test)

3 – Your data set

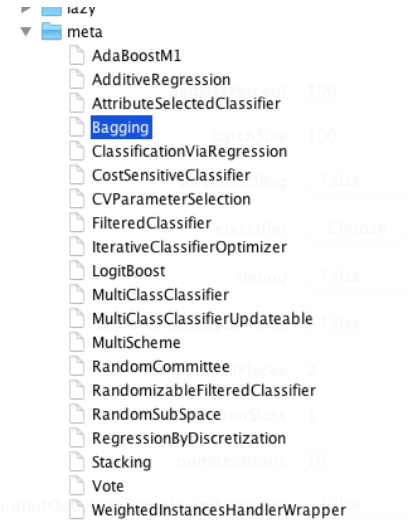
4 – Number of repetitions

5 – The Algorithms, in this case Ensemble algorithms are in the “meta” folder

In Algorithms (number 5), click and “add new …”

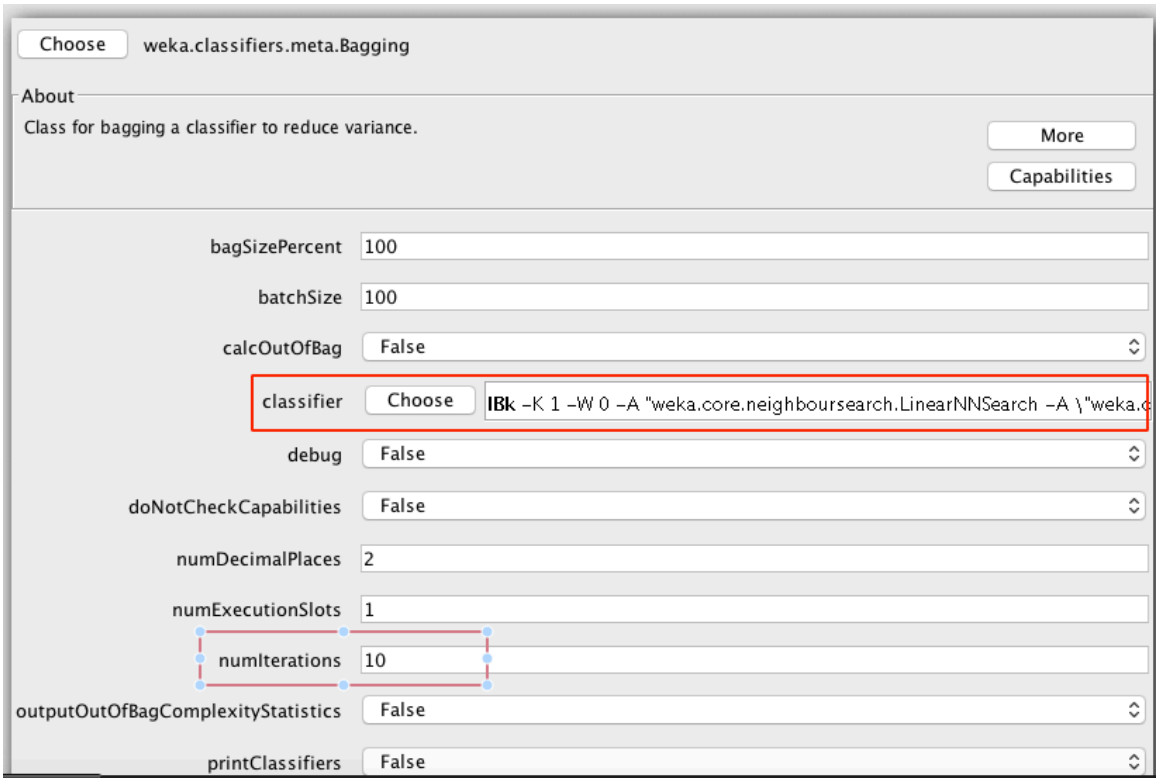
You can choose here any algorithm, like KNN, MLP, Naïve, etc, but to be an ensemble, we we have to choose the “meta” folder.

Configuring the ensemble in Weka



Choose one of the meta algorithms, the more common are Baggind, AdaBoostM1 and Stacking.

When you choose the ensemble, the 2 main parameters that you have to choose is the classifier, KNN, Decision Tree, MLP, etc and the number of classifiers over the ensemble, that is the Weka is call numIterations. ie **numIterations == number of classifiers**



Now you can execute it that the result will be in the file that you choose.

AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN
Num_true_positives	False_positive_rate	Num_false_positives	True_negative_rate	Num_true_negatives	False_negative_rate	Num_false_negatives	IR_precision	IR_recall	F_measure
11.0	0.03333333333333333	1.0	0.9666666666666667	29.0	0.15384615384615385	2.0	0.9166666666666666	0.8461538461538461	0.8799999999999999
11.0	0.0	0.0	1.0	30.0	0.15384615384615385	2.0	1.0	0.8461538461538461	0.9166666666666666
12.0	0.0	0.0	1.0	30.0	0.07692307692307693	1.0	1.0	0.9230769230769231	0.9600000000000001
11.0	0.0	0.0	1.0	30.0	0.15384615384615385	2.0	1.0	0.8461538461538461	0.9166666666666666
9.0	0.03333333333333333	1.0	0.9666666666666667	29.0	0.3076923076923077	4.0	0.9	0.6923076923076923	0.7826086956521738
11.0	0.03333333333333333	1.0	0.9666666666666667	29.0	0.15384615384615385	2.0	0.9166666666666666	0.8461538461538461	0.8799999999999999
11.0	0.03333333333333333	1.0	0.9666666666666667	29.0	0.15384615384615385	2.0	0.9166666666666666	0.8461538461538461	0.8799999999999999
11.0	0.06896551724137931	2.0	0.9310344827586207	27.0	0.15384615384615385	2.0	0.8461538461538461	0.8461538461538461	0.8461538461538461
10.0	0.034482758620689655	1.0	0.9655172413793104	28.0	0.23076923076923078	3.0	0.9090909090909091	0.7692307692307693	0.8333333333333333
11.0	0.06896551724137931	2.0	0.9310344827586207	27.0	0.15384615384615385	2.0	0.8461538461538461	0.8461538461538461	0.8461538461538461
9.0	0.0	0.0	1.0	30.0	0.3076923076923077	4.0	1.0	0.6923076923076923	0.8181818181818181
12.0	0.0	0.0	1.0	30.0	0.07692307692307693	1.0	1.0	0.9230769230769231	0.9600000000000001
10.0	0.06666666666666667	2.0	0.9333333333333333	28.0	0.23076923076923078	3.0	0.8333333333333334	0.7692307692307693	0.8
11.0	0.1	3.0	0.9	27.0	0.15384615384615385	2.0	0.7857142857142857	0.8461538461538461	0.8148148148148148
13.0	0.03333333333333333	1.0	0.9666666666666667	29.0	0.0	0.0	0.9285714285714286	1.0	0.962962962962963
12.0	0.0	0.0	1.0	30.0	0.07692307692307693	1.0	1.0	0.9230769230769231	0.9600000000000001
13.0	0.0	0.0	1.0	30.0	0.0	0.0	1.0	1.0	1.0