# Short instructions on using Weka

G. Marcou[[1]](#footnote-1)

Weka is a free open source data mining software, based on a Java data mining library. Free alternatives to Weka exist as for instance R and Orange. The current version of Weka (3.7.6) can be downloaded from <http://www.cs.waikato.ac.nz/ml/weka>. Weka has a book companion[[2]](#footnote-2), describing the version 3.6 of the software, which is therefore tagged as stable. This document is organized as a set of basic question/answers that can be used to build and validate models during the tutorials.

*How to launch Weka on Windows?*

Weka can be launched either from the main Program menu (Weka.bat file) or by double-clicking on any *\*.arff* file. In the former case, the default working directory is the installation directory. In the latter case, the default working directory is the location of the selected arff file.

The maximum amount of heap memory Weka uses is limited. The larger the memory is the bigger amount of data Weka can manage. To change the default heap size, one should edit the [*C:\Program*](file:///C:\Users\C:\Program) *Files\Weka3.7.6\RunWeka.ini* file. The line containing the word *maxheap* should be:

maxheap=1600m

in order to set the heap to a maximum.



Figure 1: Weka start menu

*How is organized Weka?*

When Weka starts, 4 working modes are proposed:

* **Explorer**: This mode is used to explore data interactively. *This is the main mode to use*.
* **Experimenter**: This mode is used to compare automatically the performances of several methods on different datasets.
* **KnowledgeFlow**: This mode allows one to treat a dataset visually organizing processes in a graph. It is conceptually similar to KNIME and PipelinePilot.
* **SimpleCLI**: a command line interface for Weka. It is useful to help writing scripts calling for Weka methods.

*I don't understand Weka vocabulary. Are there some definitions?*

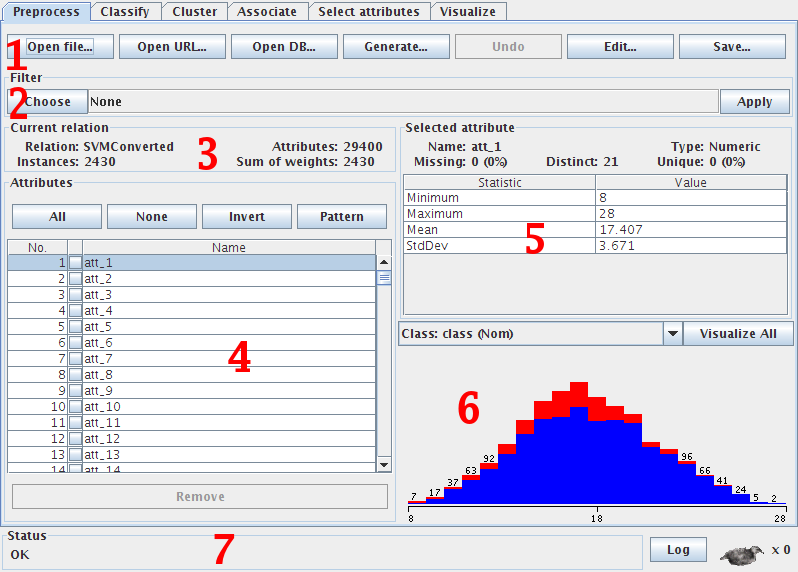
Weka uses Data Mining vocabulary. Briefly, the main key words are:

* *Relation*: this is a dataset. It can be seen as a spreadsheet with columns containing molecular descriptors and rows containing molecules.
* *Attribute*: a part of the description of an object. Here it can be understood as a synonym of molecular descriptors. It is a scalar quantity with a given *type*.
* *Instance*: an object of the dataset. Here it can be understood as a molecule represented by its molecular descriptors.
* *Class*: it is the target of the modeling. Here it can be taken as a synonym of molecular property.
* *Type*: it is a qualification for an attribute or a class. It can be one of the following words:
* *Numeric*: the value is a real/rational or integer number.
* *Binary*: the value is False (f) or True (t).
* *Unary*: the value is 1 or is missing.
* *Nominal*: the value is word in a finite set of word.
* *String*: the value is a character string.
* *Date*: the value is a date.
* *Relational*: used in the frame of multi-instance learning only. It relates to the notion of relationship in a relational database.
* *Missing*: there is no value.
* *Empty*: the object has no description.
* *Supervised/Unsupervised*: a qualification distinguishing two situations: using or not informations about the class (property) of the objects (molecules).

*How do I use Weka Explorer Preprocess interface?*

The first view of Weka Explorer is the **Preprocess** tab. This interface is designed to manage the data sources. It is divided into 7 frames (see the red numbers on Figrue 2).

1. This frame allows to load a dataset using the **Open File** button, to connect to data sources on the web or to a database, or to generate a random dataset. A **Undo** button can be used to reverse accidental manipulations. The **Edit** button opens a very rough spreadsheet to manually edit the data. The **Save** button outputs the current state of a dataset into a file.
2. The *Filter* frame is the core of data transformations. The **Choose** button allows to select an appropriate transformation method which then appears in the text box of this frame. A click on the text box pops up the configuration interface of the selected method. A method is applied with the **Apply** button.
3. This frame summarizes the dataset characteristics: the name of the dataset (relation), the number of molecular descriptors (attributes), the number of molecules (instances). Since, in the tutorial, instances are not weighted, the sum of weight is always equal to the number of instances.
4. Each line of this frame correspond to a molecular descriptor. Selecting a line in this frame updates the informations displayed in frames 5 and 6. Is is possible to remove manually any molecular descriptors using this interface.
5. The frame *Selected attribute* summarizes some statistics about the molecular descriptor selected in the area 4: minimal and maximal value observed, mean value and standard deviation.
6. The frame 6 provides an histogram of the observed values of the selected attribute. It can be colored using informations about the class (the property) of each molecule.
7. The *Status* frame is used to display informations about the current run. If some calculations are ongoing, the small bird will dance left and right.

Figure 2: Weka Preprocess interface. Refer to the text for explanations about each frame labeled with a red number.

*How are organized methods in Weka?*

Many Weka interfaces will provide a button **Choose**. These buttons typically open a selection menu organized in folders and subfolders, each folder mapping a major concept in data mining.

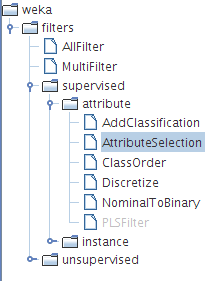


Figure 3: Example of folder hierarchies open by a button Choose.

*How do I use Weka Classify interface?*

The **Classify** interface is designed for supervised modeling. It gives access to many methods, validation heuristics and management of models. It is divided into 6 parts indicated in red on the Figure 4.

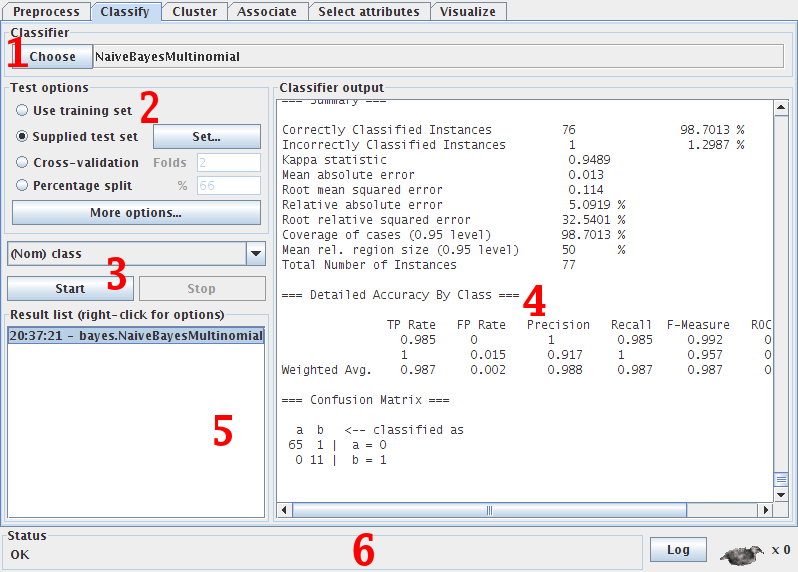


Figure 4: View from the Weka Classify interface. Refer to the text for explanations about the frames tagged with red numbers.

1. The *Classifier* frame is the place used to select a particular data mining method. The **Choose** button will open tree architecture of folders in which methods are classified. A click on the text box will open the configuration interface of the selected method. The **Start** button (in part 3) launches calculations.
2. The *Test options* frame is dedicated to different types of model validation. It is possible to assess the performances of a model on training set, on supplied test set (in that case it is necessary to select a file or a data source to which apply a model), in cross-validation (the number of folds must be set), or split the initial dataset into training and test using a given percentage of data for training. The button **More options** opens another interface (Figure 5), controlling the output in area 4 (see Figure 4). For further processing of the results, it is recommended to pay attention to the following points:

* **Store predictions for visualization** will use more memory but is mandatory to use Weka plotting facilities.
* **Output predictions** will print out the predictions for molecules. However, the order of the molecules is not preserved unless the predictions concern a supplied test set or the box **Preserve order for % split** is ticked. It is not possible to relate these outputs to the molecules in a molecule file in the other cases.
* The **Random seed for Xval / % Split**, is the seed of the random number generator used into the heuristics. It is important when looking at a model using cross-validation or percentage split, to validate the model several time changing this seed. The performances of the model shall change, giving a perception of the variance of the performances that shall be expected for the model.

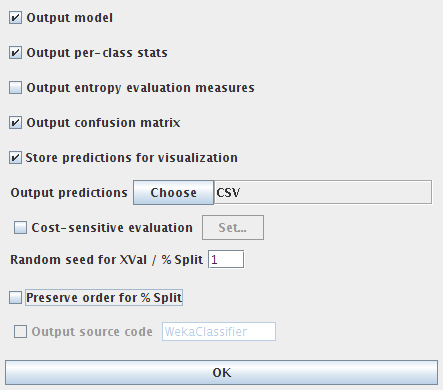


Figure 5: View of the More Options interface of the tab Classify.

1. This part contains the button **Start**, **Stop** and the **Class** selection. The **Class** selection allows one to use any attribute as the target for modeling. The **Start** button actually launches the model building and validation process. The **Stop** button is then activated in order to interrupt calculations at any time.
2. The frame *Classifier output* reports all messages generated during the building, validation or application of a model.
3. The frame *Result List* is a key part in this interface. All models and messages are kept in memory. The whole modeling history is recorded in this area and one can come back to any earlier modeling study using this interface. However, a right click on any line in this area will open a pop-up menu (Figure 6) which offers many important options:

* **Save result buffer** writes the content of the area 4 to text file. It is recommended to give to this file the extension *.out*.
* **Save model** saves the current model into a model file in Object Model File. It shall be possible to use the more conventional format PMML for interpretability with other data mining software.
* **Load model** loads a model which will then be ready to be applied to new data.
* **Re-evaluate a model on current test set** applies the current model (just generated or loaded from a file) to a dataset. The dataset is the one loaded using the button next to **Supplied test set** in part 1.
* **Visualize** items generate several plots useful to understand the strengths and weaknesses of a model. In particular, for regression models, the **visualize classifier errors** item generates experiment versus prediction scatter plot. The size of each point is proportional to the error. Also, the **visualize threshold curve** item generates a plot very similar to the ROC curve for classification models.

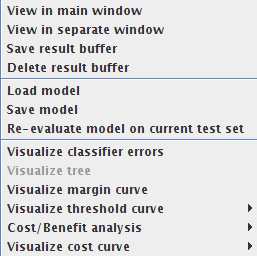


Figure 6: View of the menu appearing when right click on the Result list.

1. The *Status* frame displays informations about the current run.

*I do not find my favorite data mining method. Is it missing in Weka?*

Weka is very complete and it is likely that your favorite algorithm was not shipped with the default install setup. You can search for it in the Package Manager (figure 7 and 8). It requires an internet connexion.

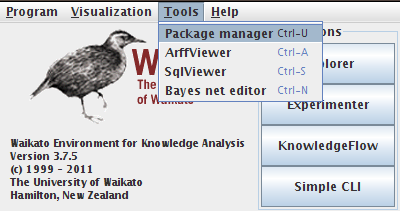
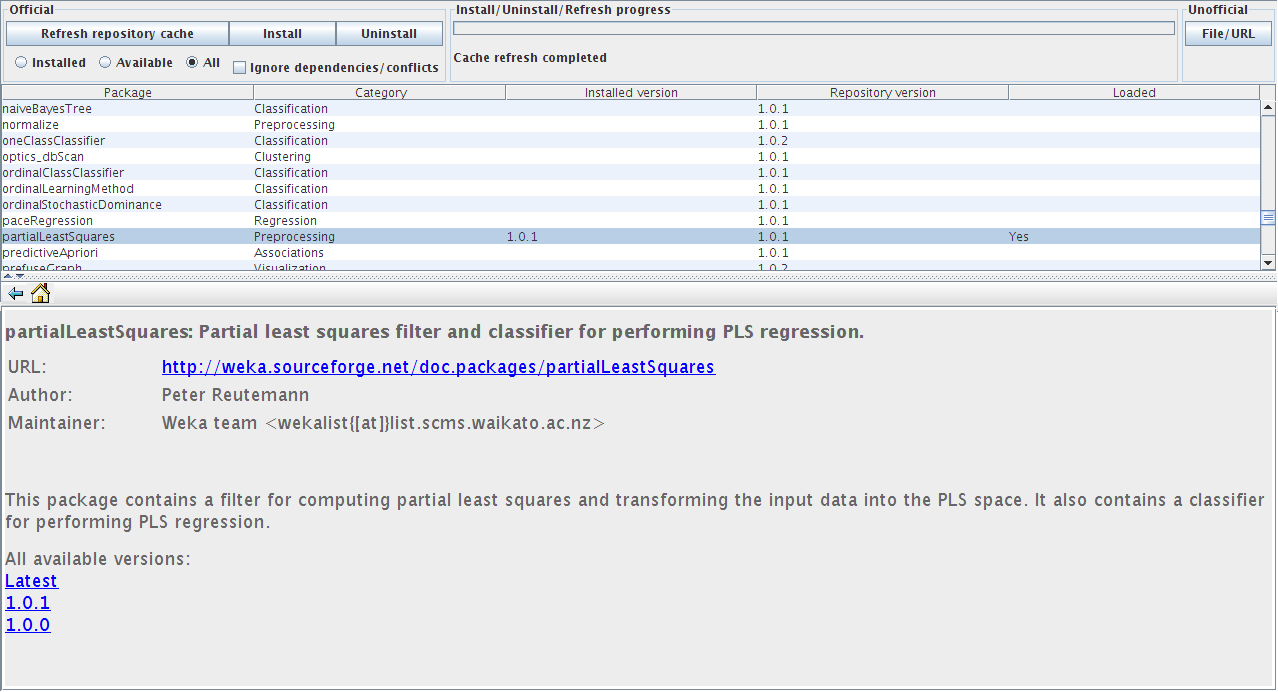


Figure 7: Location of the Package Manager in the Weka main frame.

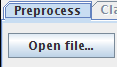
Figure 8: The Weka package manager to install or uninstall data mining methods.

*Sample run for Classification models of A2A binding.*

This section concernes an example of obtaining and validation of classification model for A2A binders. All files are in the folder Classification.



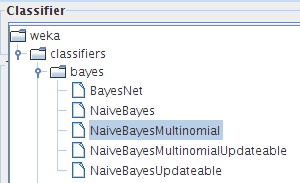
1. Open the Weka **Explorer** interface.



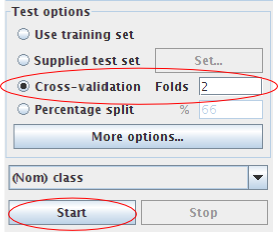
1. In the **Preprocess** tab, load the file train.arff.
2. Change for the **Classify** tab.

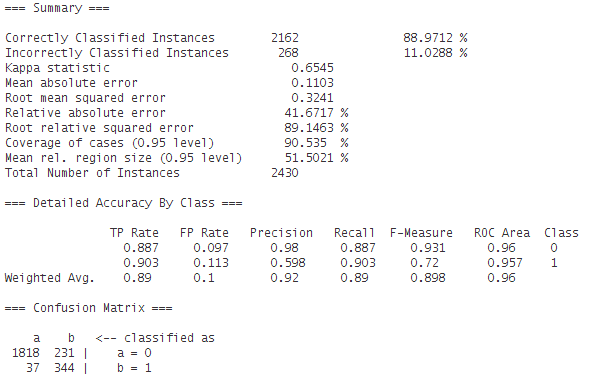


1. Click the button **Choose** and select the method *Bayes->NaiveBayesMultinomial*.

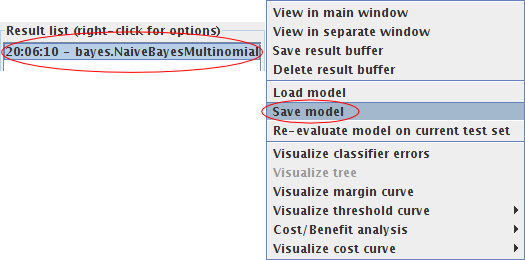


1. Click on **Cross-validation** and set the number of fold to 2. Then click on **Start** button.

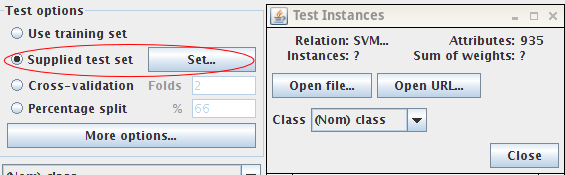




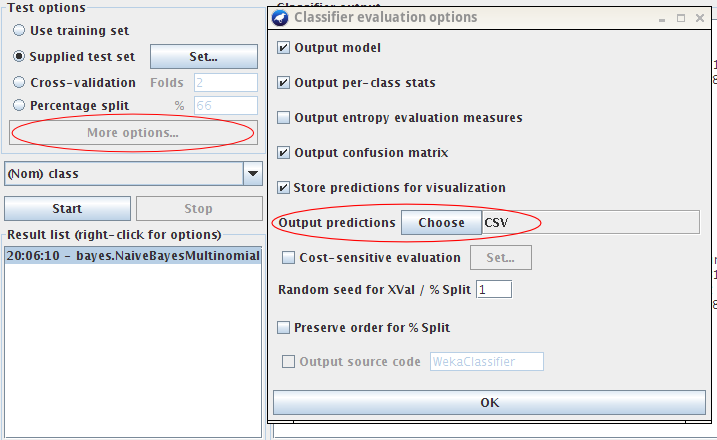
1. Right-click on the last line of the *Result list* frame and save your model as A2AC.model.

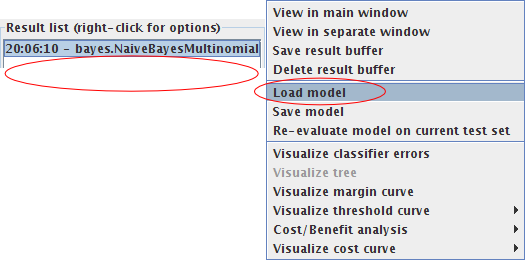


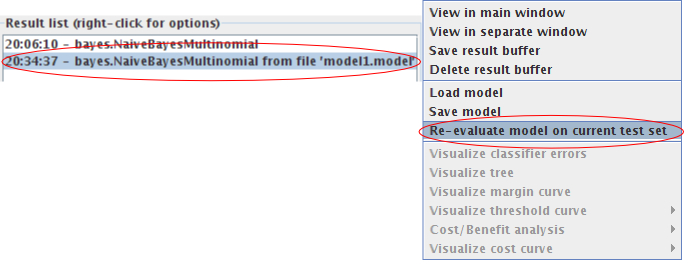
1. Click on **Supplied test set**, then on the button **Set** and select the file test.arff.

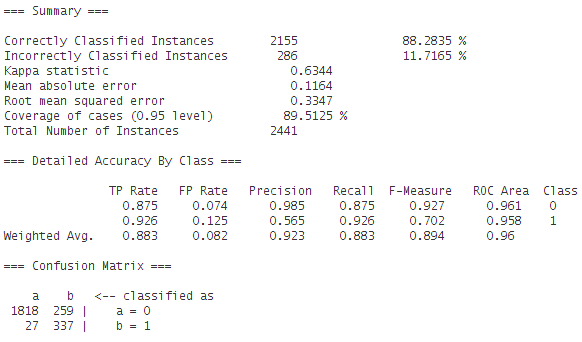


1. Click on the button **More options**. In the interface click on the button **Choose** and select *CSV*.

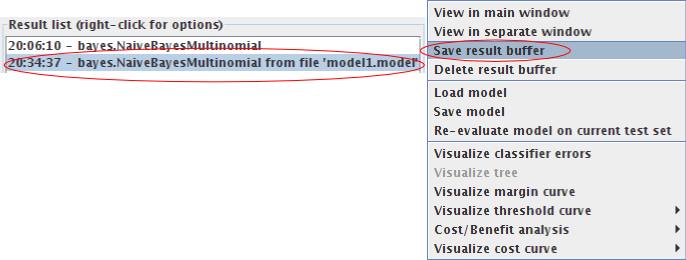


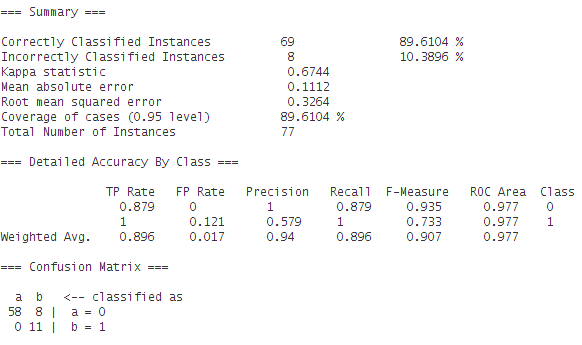
1. Right-click in the Result list frame and click on **Load model**. Then chose your model A2AC.model.
2. Right-click on the last line of *Result list* and click on **Re-evaluate on current test set**.





1. Right-click on the last line of *Result list*, click on **Save result buffer** and name your file test.out.

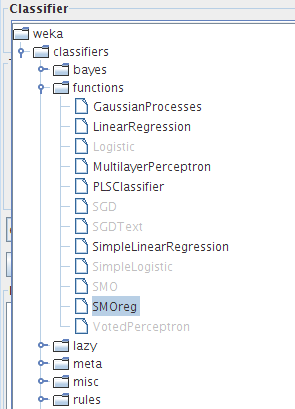


1. Click on **Supplied test set**, then on the button **Set** and select the file external.arff.
2. Right-click on the last line of *Result list* and click on **Re-evaluate on current test set**.
3. Right-click on the last line of *Result list*, click on **Save result buffer** and name your file external.out.

*Sample run for Regression models of A2A binding.*

This section is an example of obtaining and validation of regression model for A2A binding (pKd). All files are in the folder Regression.

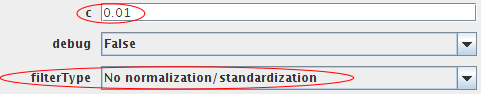
1. Open the Weka **Explorer** interface.
2. In the **Preprocess** tab, load the file train.sdf.
3. Change for the **Classify** tab.
4. Click the button **Choose** and select the method *functions->SMOReg*.



1. Click on the text box near the button **Choose**.



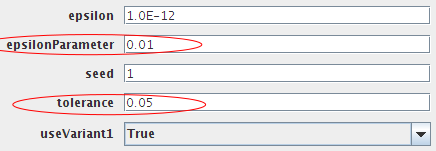
1. Configure the method using C=0.01 and select *No normalization/standardization* as **filterType**.

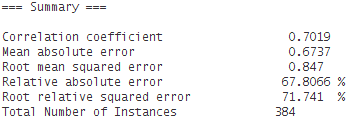
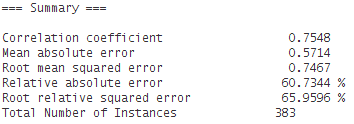


1. Click on the text box near **regOptimizer**.



1. Change the value of **epsilonParameter** to 0.01 and **tolerance** 0.05.



1. Click on **OK** to validate your configuration.
2. Click on **Cross-validation** and set the number of fold to 2. Then click on **Start** button.
3. Right-click on the last line of the *Result list* frame and save your model as A2AR.model.
4. Click on **Supplied test set**, then on **Set** button to select the file test.arff.
5. Click on **More options**. In the interface click the **Choose** button and select *CSV*.
6. Right-click in the *Result list* frame and click **Load model** and chose your model A2AR.model.
7. Right-click on the last line of *Result list* and click on **Re-evaluate on current test set**.
8. Right-click on the last line of *Result list*, click on **Save result buffer** and name your file external.out.

*Link Weka output and your SDF file*

Download from infochim.u-strasbg.fr the applications ModelAnalyzerC and ModelAnalyzerR which relate Weka predictions with molecular structures stored in SD file. Both applications read Weka buffer file (*\*.out*) in which predictions on the test set are stored in CSV format.

1. Université de Strasbourg, Faculté de Chimie, Laboratoire d'Infochimie [↑](#footnote-ref-1)
2. *Data Mining: Practical Machine Learning Tools and Techniques 3rd Edition*, I. H. Witten, E. Frank, M. A. Hall, **Morgan Kaufmann**, ISBN: 978-0-12-374856-0 [↑](#footnote-ref-2)