Algorithms from Machine Learning – interesting for CAPRI?

-by Wolfgang Britz, September 2011 –

Inhalt

Background

A serious challenge for large-scale economic models is the dimensionality of the results generated by model runs. These reflect the high level of dis-aggregation in different dimensions and the many aspects dealt with in these tools, such as relating to economic, social and environmental indicators. A single simulation run with CAPRI based on the farm type modules produces over 20 Mio non-zeros. Clearly, any of these numbers is generated by a computer based model and should hence be a non probabilistic outcome depending on the input and the code used. Specifically, the relation between the input and any single number outputted is determined by the model structure and parameterization, and pre and post-processing code. It must hence be possible to track any change quantitatively back to the shock analyzed.

But that rather theoretical point of view has very little to do with the task at hand when one has to distill from a set of model outcomes an analysis. The questions here are: what are the most important results, i.e. salient to the questions underlying the analysis and large enough to matter, and how can they be explained? For the client, the story behind the results is often at least equally important as the results themselves. If the story is well told, the “black box” character of the tool is

removed and its usefulness in depicting major cause-effect relations becomes evident. Telling a good and right story requires however often quite some time in analyzing results in a systematic way.

The user will hence have to decide for which items of the huge data set a thorough analysis of underlying drivers is advisable. Limited time and human resources will set tight limits to the extent of such systematic analysis. Typically, in any report, only a few dozen key results (perhaps complemented with a few maps showing several hundredths numbers) will be presented. But these key results, such as changes in aggregate welfare, farm income, GHG emissions or the nitrogen balance are calculated from thousands of simulated items. How can we discover "the story behind the results", i.e. which regions, activities, price or policy changes etc. are most important for the aggregate changes communicated?

The exploitation tools developed for CAPRI with a flexible on-the-fly approach to produce tables, graphs and maps had been an important step to improve the efficiency in exploiting and analyzing results. But in parallel, CAPRI has grown in scope and scale. It might be the time now to consider new approaches to analyze model outcomes. Before discussing the integration of machine learning in the exploitation tools, we will quickly review the current approaches based on the current exploitation tools.

Using the CAPRI exploitation tools for systematic results analysis

A basic idea when using the CAPRI exploitation tools is go top-down from key aggregate results to the underlying drivers. The starting point of the analysis can be e.g. changes in farm management (crop shares, stocking densities), a welfare analysis or environmental impacts at aggregate level. From there, one can track e.g. down the changes to specific sectors/activities or regions by using more detailed tables or maps. These approaches had been presented in several training sessions.

Recent additions to the GAMS code further support result analysis:

- Decomposition of aggregate yield changes (http://www.capri-model.org/docs/endog_yields.pdf)
- Sensitivity analysis for endogenous features with the supply model (http://www.capri-model.org/docs/Sensitivity_analysis_for_model_features_in_the_CAPRI_supplymodel.pdf)
- Decomposition of changes in behavioral functions of the market part (http://www.capri-model.org/docs/Decomposing_market_model_results.pdf)

All these approaches built on known structural features of the model. The now added "Machine Learning" package aims to add more data driven approach applicable also with less a priori knowledge.

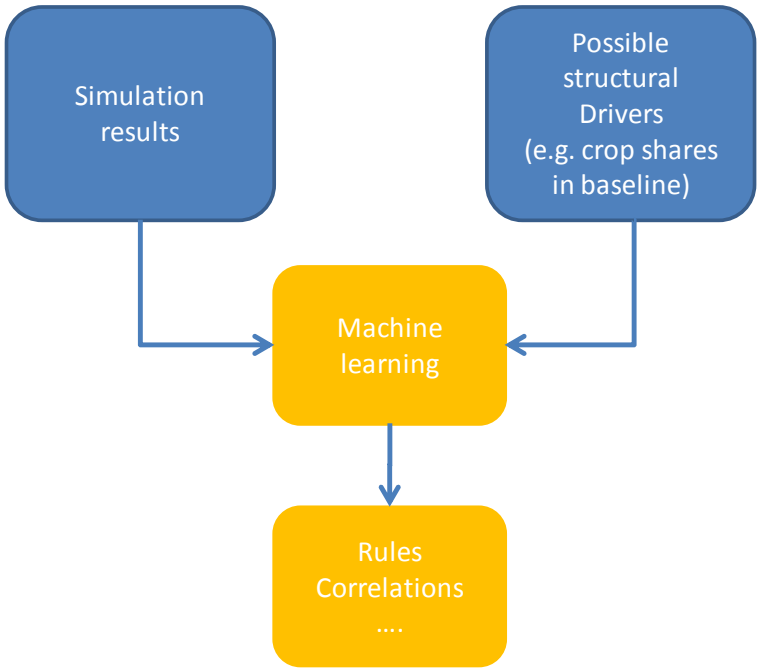
Machine learning

Wikipedia gives the following definition: **"Machine learning**, a branch of artificial intelligence, is a scientific discipline concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data, such as from sensor data or databases. Machine Learning is concerned with the development of algorithms allowing the machine to learn via inductive inference based on observation data that represent incomplete information about statistical phenomenon. Classification which is also referred to as pattern recognition, is a important

task in Machine Learning, by which machines "learn" to automatically recognize complex pattern, to distinguish between exemplars based on their different patterns, and to make intelligent decisions."

That is naturally a very general description. Machine learning has been widely in a wide range of application fields. A typical example is the analysis of which clients of a bank has been given credits. We have many observations with "credit granted" or "credit refused", and probably a longer list of attributes of the clients (age, sex, income, amount of the credit asked for, time since being a customer with the bank, past bookings …). Machine learning could be applied to define a set of rules which based on past decisions predict if a credit would be granted for a new application or not. Machine learning will in many cases also be able to tell something about the possible error range linked with the decision. That could e.g. allow the banks to make fast decisions in many cases, and spend more time on the tricky cases. The book by Witten et.al. 2011 gives many interesting examples.

Now, we can e.g. see the income changes in each farm types in a simulation compared to the baseline as an outcome we want to predict, and their production program and changes in prices and premiums as the attributes used to explain that outcome. Some farm types might exhibit very large income changes, other little ones. What are common characteristics of the one and the other group? Machine learning might then come up with a "pattern" (e.g. based on a regression model) which determines the most important attributes impacting income changes in a given simulation. Machine learning has thus a lot of similarities with statistics – indeed many methods can also be found in statistical packages - but the focus to decide upon which attributes and relations matters is shifted to a certain extent from the human being to the computer. And, the tool box used in machine learning differs to a certain degree from classical statistics. And, not of least, many of the algorithms had also been developed keeping computing time in mind.



Implementation in CAPRI

The implementation in CAPRI is based on the existing exploitations tool of the CAPRI GUI and the WEKA machine learning library (Witten et.al. 2011) which is also integrated into other well known packages such as RapidMiner. Thanks to the GNU license including full access to the underlying Java source code, it was possible to integrate the functionality of WEKA into the CAPRI exploitation tools. Only a few code changes were necessary to pass data from the tables and maps shown in the CAPRI

GUI to the WEKA library (see below). That is done automatically in the background with the aim to reduce user input in the process.

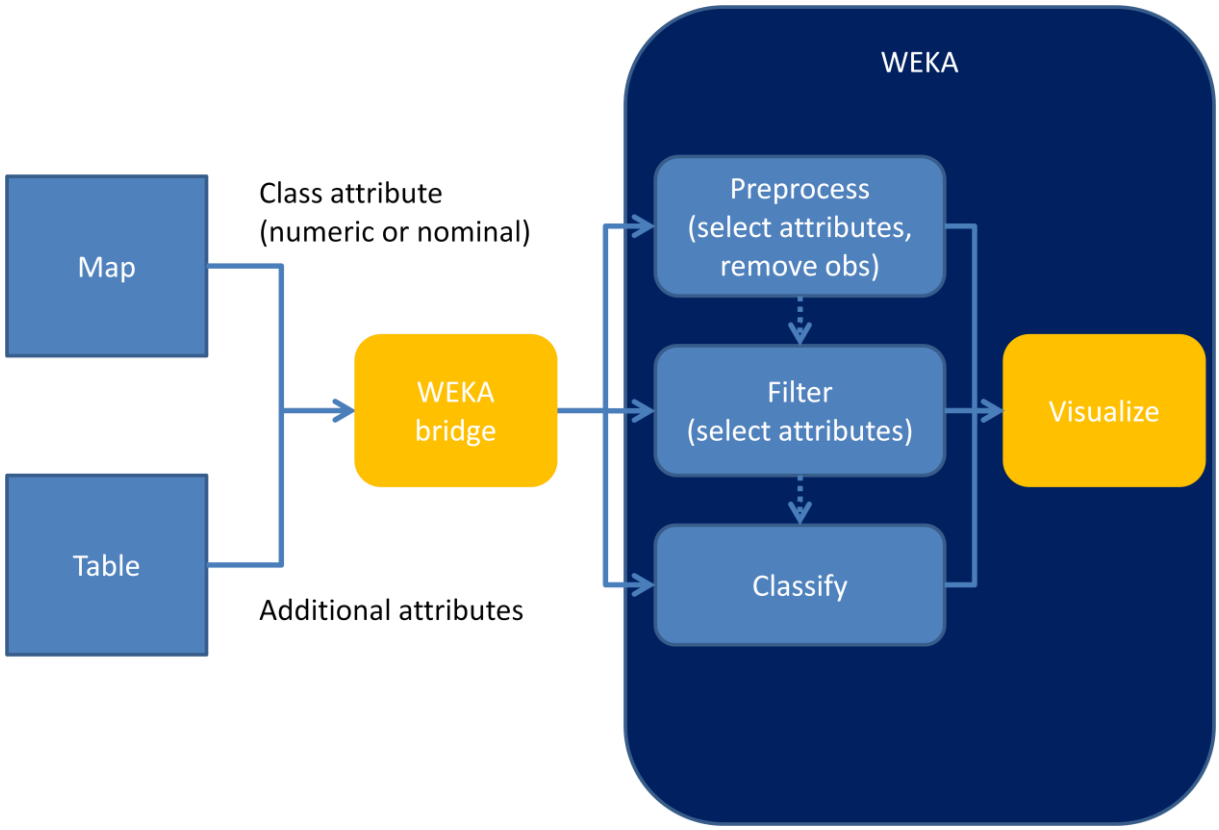
As a consequence, a very powerful set of filtering and classification as well as related visualization tools from machine learning can be applied to the result sets from CAPRI inside the existing exploitation tools.

The current implementation is based on the interaction of two views:

1. A **map** or a table using classification colors – it defines the class attribute (=dependent variable) of the data to classify. For classification algorithms which require nominal values, the assigned class from the classification is used.
2. A **table** with the "explanatory" attributes.

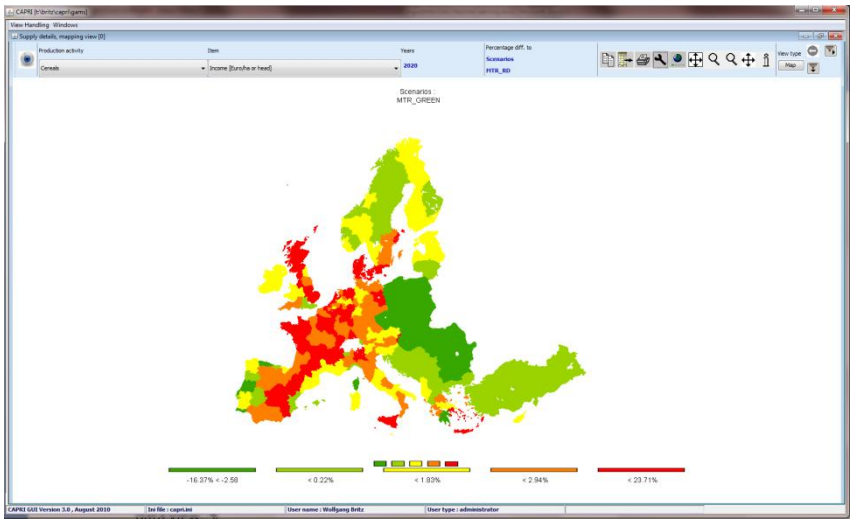
Both tables must be, as conventionally in the exploitation tools, the observations in the rows. For maps, each map carries the data for a region. But one might also work with two tables where the observations are not strictly geo-referenced entities such as farm types.

The CAPRI GUI will automatically send new data to the WEKA GUI if either the map (or the table using classification colors) or the table is updated by a user action. The basic data flow is shown in the graphic below.



Interaction between CAPRI GUI and WEKA

Let's construct an example: we want to check if the income change in cereals in a simulation depends on the crop shares of cereals and the yields. In order to do so, we first render our map as usual (table "Farm details, mapping view", use the option dialogue to show percentage changes against the baseline):



The regions shown are our instances and the value plotted for a region defines the class attribute we want to analyze. Any one instance consists of a vector of attributes of which one is the “class value”, i.e. the value to classify, which can be numeric or nominal. The other attributes are used for classification or clustering and stem from a second table (see below). Classification methods which use nominal values can also be sued. In that case, the class chosen for the region, as seen from the color in which is drawn, defines the class attributes. In our example above, each region would fall into one of five classes.

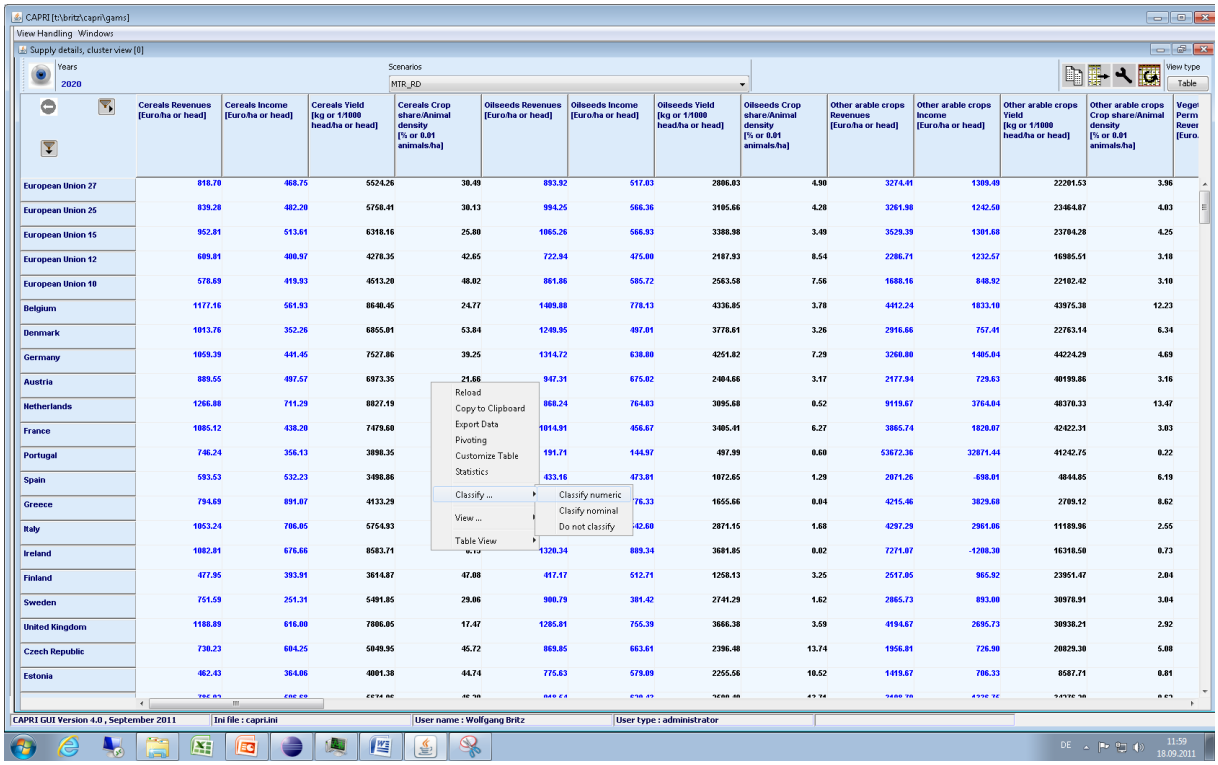
Next, we open a second table with the data we want to use as explanatory attributes. The latest trunk comprises the table “Supply details, cluster view” which comprises promising attributes which are possible candidates to explain many changes in a simulation (for all activity aggregates: crop shares/stocking densities, revenues, income, yields).

Supply details, cluster view [0]

Years: 2020 — Scenarios: MTR_RD

	Cereals Revenues [Euro/ha or head]	Cereals Income [Euro/ha or head]	Cereals Yield [kg or 1/1000 head/ha or head]	Cereals Crop share/Animal density [% or 0.01 animals/ha]	Oilseeds Revenues [Euro/ha or head]	Oilseeds Income [Euro/ha or head]	Oilseeds Yield [kg or 1/1000 head/ha or head]	Oilseeds Crop share/Animal density [% or 0.01 animals/ha]	Other arable Revenues [Euro/ha or h
European Union 27	818.70	468.75	5524.26	30.49	893.92	517.03	2806.03	4.90	
European Union 25	839.28	482.20	5758.41	30.13	994.25	566.36	3105.66	4.28	
European Union 15	952.81	513.61	6318.16	25.80	1065.26	566.93	3388.98	3.49	
European Union 12	609.81	400.97	4278.35	42.65	722.94	475.00	2187.93	8.54	
European Union 10	578.69	419.93	4513.20	48.02	861.86	585.72	2563.58	7.56	
Belgium	1177.16	561.93	8640.45	24.77	1409.88	778.13	4336.85	3.78	
Denmark	1013.76	352.26	6855.01	53.84	1249.95	497.01	3778.61	3.26	
Germany	1059.39	441.45	7527.86	39.25	1314.72	638.80	4251.82	7.29	
Austria	889.55	497.57	6973.35	21.66	947.31	675.82	2404.66	3.17	
Netherlands	1266.88	711.29	8827.19	12.49	868.24	764.83	3095.68	0.52	
France	1085.12	438.20	7479.60	29.96	1014.91	456.67	3405.41	6.27	
Portugal	746.24	356.13	3898.35	6.25	191.71	144.97	497.99	0.60	
Spain	593.53	532.23	3498.86	20.42	433.16	473.81	1072.65	1.29	
Greece	794.69	891.87	4133.29	18.20	688.18	776.33	1655.66	0.84	
Italy	1053.24	786.05	5754.93	28.54	874.83	542.60	2871.15	1.68	

In order to start the clustering/classification, we click in the table to open its popup-men and then select “Classification”:



We clicking one of the option if we can then decide to:

1. use *numerical classification* methods such as different regression methods. The observations in the map define the dependent variable.
2. Use the class assigned by the maps input into *nominal classification*.
3. To switch classification off.

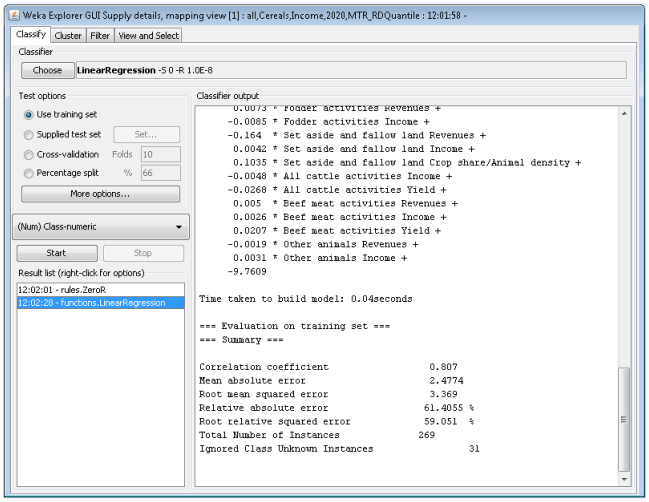
A new window will be opened which shows the WEKA GUI (see below).

The WEKA GUI

The classification is based on the complete functionality of the WEKA GUI regarding attribute selection/visualization, filtering and classification, see http://www.cs.waikato.ac.nz/~ml/index.html. There are very good manuals available from the site (the latest user manual is also available from http://www.capri-model.org/docs/WekaManual-3-6-5.pdf), so that only a few major tips are given below for fast start.

The tabs “Classify”, “Cluster”, “Filter” and “View and select” allow the user to access specific part of the WEKA functionality. The result set from the current classification run can be shown in the lower left panel (result list). For each result set, a popup menu opens options, e.g. to show a graph with the prediction errors.

Classification

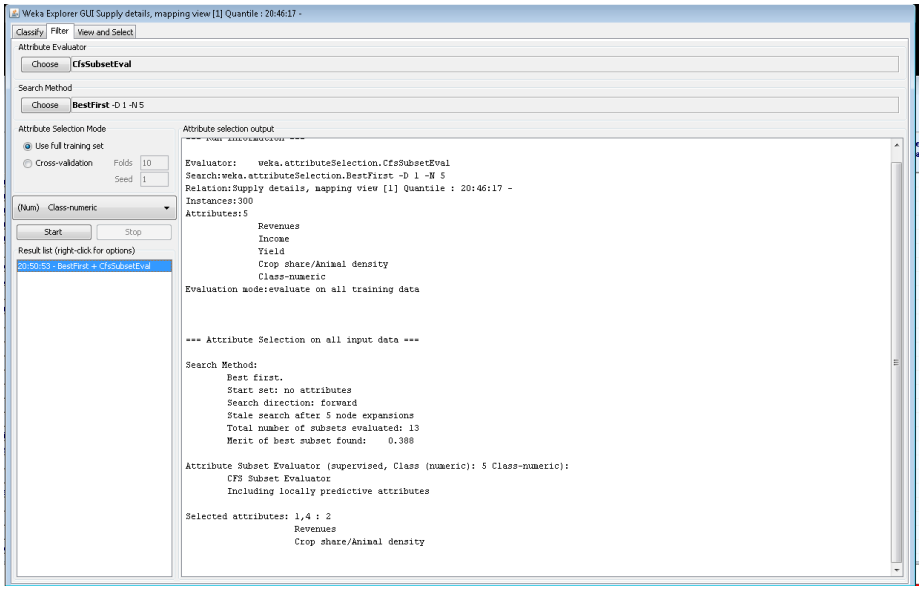


- The "**choose**" button will give access to a wide range of **different classifiers**, many of which have additionally options which can be edited by users. A multiple linear regression using the Akaide criterion for model selection is used as the default, assuming that most people will start with using numerical values as class attributes. Please not that switching between nominal and numerical class attributes might trigger error messages if the currently selected classifier cannot handle the newly selected class attribute type.
- It is recommended for our purposes to use under "Test options" "Use training set" (the default in our implementation) as we are typically not interested in an out-of-sample test of the prediction quality.
- The actual classification can be started with the "start" button. If the data in the background are updated, the actually chosen classifier with the chosen options will be started on the new data set automatically. In absence of errors the "Classifier output" on the RHS will hence typically show results based on the latest selected data.
- The results can be visualized by clicking with the mouse on an item in the result list, the last on in the list always being the newest. If one has tried several classifiers, the old results remain available. However, if the data in the background change, the old results are automatically removed.

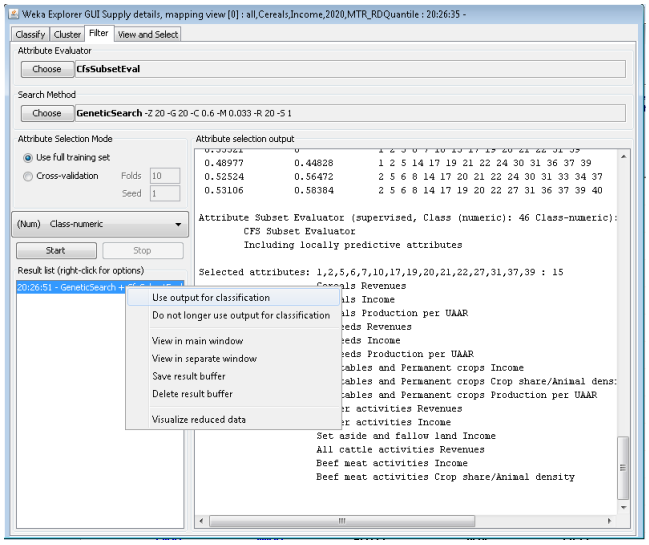
The reader should note that all the functionality described is from the standard WEKA GUI so that the user manual from WEKA can be used for further information.

PS: The cluster panel is not described, it works quite similar. Note however that filters are not applied to the cluster (see below).

Filtering



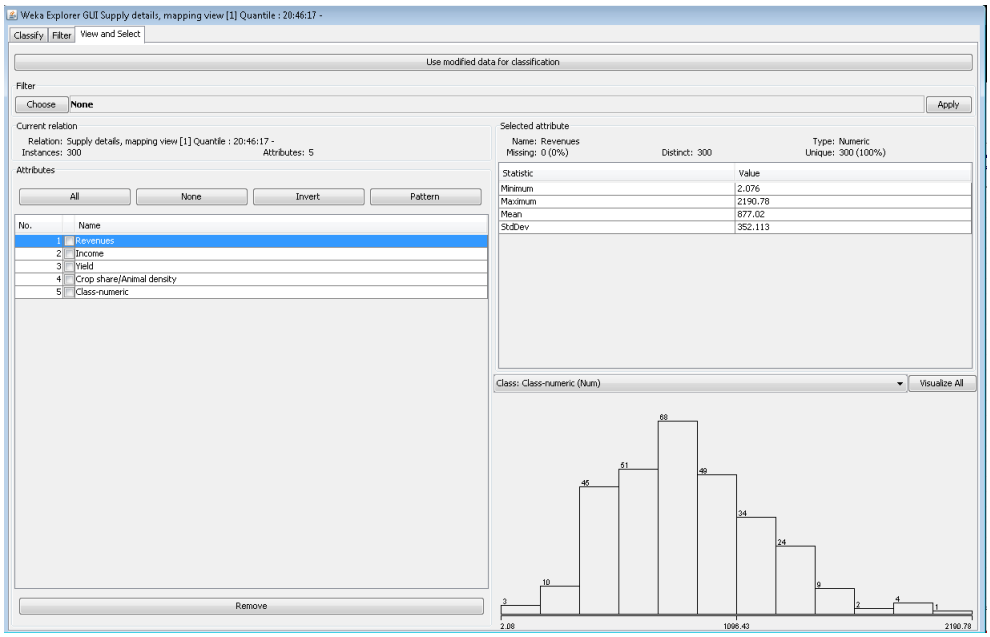
The filter panel allows running different types of filters which remove attributes, in many cases reflecting the correlation between attributes. In order to use the result from the filter run, click on the result set and chose “Use output for classification”:



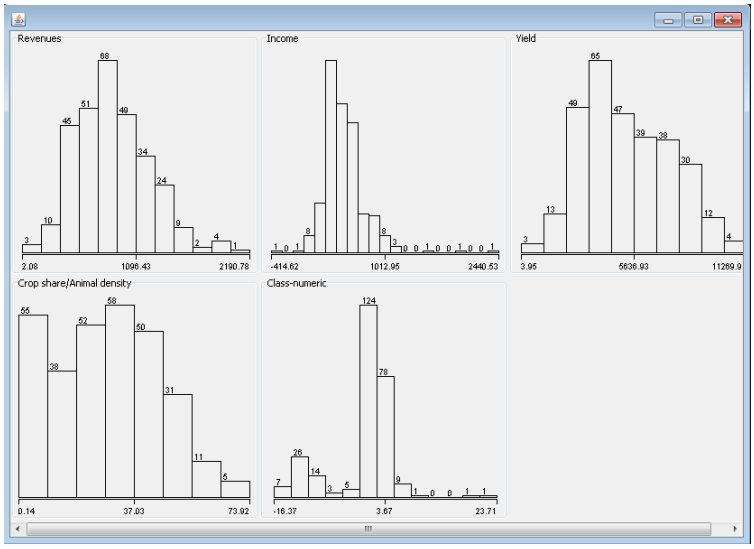
The last selected filter will be automatically restarted if a new data set is implicitly loaded (change of the map or of the data in the cluster table with the explanatory results). In order to switch off the use of the filter, select “Do not longer use output for classification”

Attribute viewing and selection

The last panel available is especially interesting to quickly analyze statistics of the underlying data:



The reader can manually remove attributes and the reduced set of attributes will then passed to the filter and classifier. However, the attribute selection is not maintained when new data are loaded. The “Visualize All” button produces graphs of all current attributes:



Summary

The integration of algorithms from machine learning based on the WEKA library and GUI offers new possibilities to systematic analysis of result sets. Thanks to the open source policy of WEKA, it was possible to integrate these powerful tools transparently in the CAPRI GUI. Depending on the experiences made over the next months, further links might be included (e.g. rending clusters in maps).

References

Ian H. Witten, Eibe Frank, Mark A. Hall (2011). Data Mining Practical Machine Learning Tools and Techniqes. Third edition. Elsevier, Amsterdam. 630 pages

Remco R. Bouckaert, Eibe Frank, Mark Hall, Richard Kirkby, Peter Reutemann, Alex Seewald, David Scuse (2011). WEKA Manual for Version 3-6-5. June 28, 2011, University of Waikato, Hamilton, New Zealand.