ANN - Classification

Menu:	QCExpert	Predictive methods	Neural Classification

This module uses neural network model to classify non-numerical (categorial, or factor) response Z based on one or more numerical predictors X. The response has usually text values, like "PETER", "JOHN", etc. called "levels". The levels have no order (one cannot say whether one level is greater than another). The factor must have at least two levels, for example: (A, B, C, D, E); (YES, NO); (green, blue, yellow); ("Elytrigia repens", "Lolium perenne", "Phleum pratense"), etc. If there are numbers in the factor columns, they are interpreted as text with no numerical value. The goal of the classification module is to find any possibly existing relationship between the predictors and the response. Examples of applications is industrial fault identification, diagnosis support from a blood analysis, and many applications in psychology, chemical and environmental research, biology and life sciences, assessing and predicting activity of drugs in pharmaceuticals, and so on.

The following table shows typical data in classification.

Predictor					**Response**
X1	**X2**	**X3**	**X4**	**X5**	
2.4	3.2	3.9	3.8	2.2	NY
-1.2	2.6	3.6	3.7	2.1	CA
3.5	1.4	6.9	4.3	0.1	WA
1.8	1.3	3.3	0.6	1.1	WA
-0.4	2.6	1.8	2.2	2.8	FL
1.2	0.5	4.6	3.1	3.7	NY
0.1	2	4.2	2.6	3.9	CA
4.7	2.3	7.5	3.6	1.9	CA
2.8	2.1	4.8	1.4	-1	WA
0.8	1	1.3	2.6	4.2	FL
3.3	3.2	4.8	5.2	2.6	NY
…	…	…	…	…	…

Here, the response factor is in the column "Response", levels of the factor are NY, CA, WA, FL, number of factor levels is R=4. Number of independent variables is M=5. If there is any dependence of response on predictors the classification model will try to find it and possibly use it for prediction of an unknown response level based on known predictor values. The prediction takes the form of R probabilities p_1, p_2, .. p_R of each level (p_i sum to 1). So, the prediction gives the probability for each level to occur at a given point of the M-dimensional predictor space given by the M predictor values. The level with the biggest probability is the predicted response level. This module uses R artificial dummy variables representing the response in a binary form.

Data and parameters

The data structure is described above. Data consists from one or more columns of independent variables (predictors) and one column of non-numeric factor (dependent variable). In the dialog window, the independent and dependent variables are selected. The "new" independent variable columns may be selected after checking the *X-Prediction* checkbox. The number of columns for prediction must be the same as the number of independent variables chosen. The rest is similar as in the previous ANN modules. The

computed model can be saved for later use by clicking on the button *Save model* after finishing optimization.

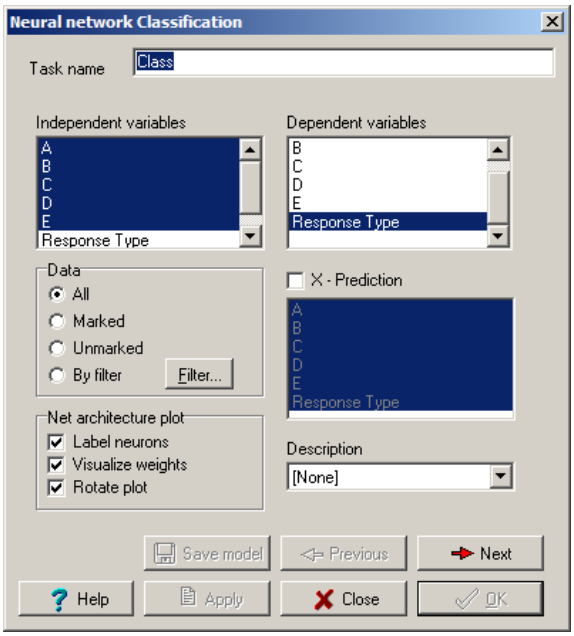


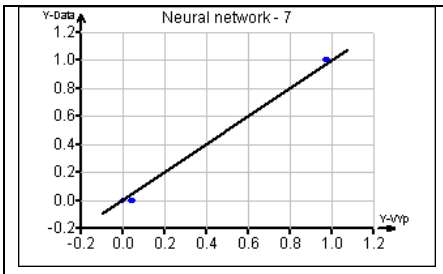
Fig. 1 Setting the parameters for ANN Classification

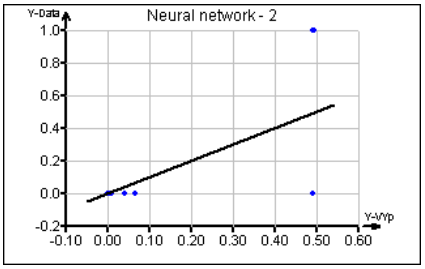
Protocol

Task name	Task name
Data	Selected subset of the data
Independent variable	List of independent variables
Transformation type	Type of used transformation of the independent variables
Dependent variable	The factor response variable
Prediction	Columns (new data) selected for prediction
Layer / Neurons	Number of layers (including input and output) and number of neurons in each layer
Sigmoid steepness	Used steepness of the sigmoidal activation functions in the network
Moment	Used moment (parameter in the optimization algorithm)
Training speed	Used learning rate (parameter in the optimization algorithm)
Terminate when error <	Termination criterion for optimization
Training data (%)	Fraction of data used to learn model (training data)
Termination conditions	User-defined number of iterations
Optimization report	Information about the trained model
No of iterations	Actually performed number of iterations
Max training error	Reached maximal absolute error for training data
Mean training error	Reached mean absolute error for training data
Max validation error	Reached maximal absolute error for testing (validation) data
Mean validation error	Reached mean absolute error for testing (validation) data
Number of cases	Number of data rows times number of factor levels
Number of weights	Number of optimized parameters of the neural network (weights)
Number of rows	Number of input rows

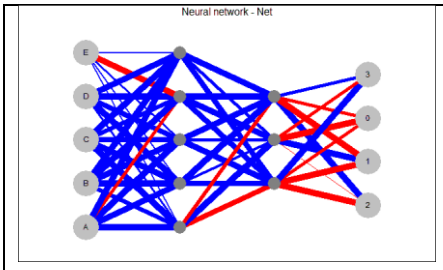
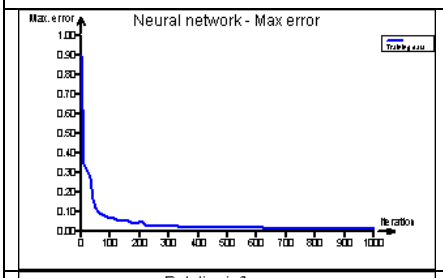
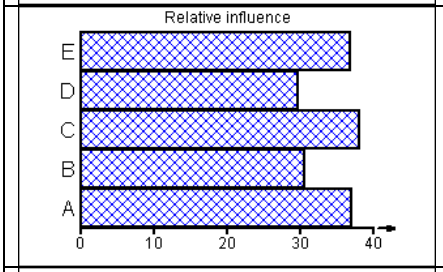
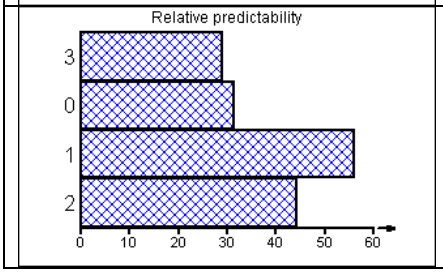
Number of levels	Number of output factor levels
Total sum of squares	Sum of squares without any model
Residual sum of squares	Sum of squared residuals (data minus model)
Explained sum of squares	[Sum of squares without any model] minus [Sum of squared residuals].
F-statistic	Computed F-statistic for the model
F-crit	Critical F-quantile
P-value	(1 – probability of the F-statistic) (if less than 0.05, the model can be assumed significant. Verbal significance statement follows: Model is significant / Model is insignificant
Classification probabilities	Estimated probabilities of each factor level based on the model
Prediction, Data, Misclass	Predicted and actual level of the factor for each case. Misclass is 1 if the prediction is wrong.
Misclassification table	Classification summary, total number of correct and incorrect classifications (Correct / Incorrect) followed by a detailed misclassification table for all factor levels. Number of correctly classified levels are on the diagonal of the square table.
Weights	Table of the optimized values of the neural network parameters
Relative influence	Relative influence of the predictors, a sum of absolute weights from each predictor.
Prediction	If selected in the *ANN Classification* dialog window, a table of predicted levels for given new predictor values is generated.

Graphs





Plot of predicted and actual levels of the factor. Predicted level on x-axis is the predicted probability of the level, on y-axis is the actual value of the factor. Is all points are concentrated around (0,0) and (1,1) then the model is fitting the data perfectly (which is not necessarily the goal, as it may imply worse prediction capability of the model due to its overdetermination).

	Graphical representation of the network architecture. If the checkbox "*Display weights*" was checked the thickness of synapses (connection lines) represent the absolute value of the corresponding weight and thus in a sense the amount of information that flows down between two neurons. From the thickness of the synapses going from the predictors we can assess their significance (the thicker lines the more significant variable). Greater weight values on the input to response nodes (thick lines going to the predictor nodes) suggest the quality of prediction of each dependent variable. Color of synapses shows only sign of the weight (red = negative weight, blue = positive weight), which is of little practical interest in complicated nets, but may be of use in simple ones. Variable nodes and factor levels are labeled by the column names / levels, if the appropriate checkbox was checked.
	Plot of the training (network optimization) process, which decrease generally the sum of squares of differences between prediction and the actual measured values, with the number of iterations on x-axis.
	Relative influence of each of the predictors on prediction computed from the absolute weights for each predictor.
	Relative predictability is the sum of absolute weights for each factor level.