Neural Network (Regression)

It handles the task of **regression**: the *Neural Network* in the output data set will provide the forecast value for one or several variables that depend on the input parameters set.

Before forecasting, it is required to train the algorithm using the training data set, namely, the training sample. Each row of such sample contains the following data:

- a set of input parameters in the fields marked as the **input** ones;
- the dependent variables values mapping the input parameters in the fields marked as the **output** ones.

Technically, the training consists in determination of *weights* — coefficients of links between neurons. In the process of training, the neural network enables to detect complex dependences between input and output parameters, and also to perform generalization. It means that on condition of the successful training, the *Neural Network* can return the correct result based on the data that was absent in the training sample, and also incomplete and/or "noisy", partially distorted data. The Quasi-Newton Broyden — Fletcher — Goldfarb — Shanno method is used for training with limited use of L-BFGS memory.

Only the fields with continuous data kind can be the **output** ones in the regression task (as opposed to the classification task). Data kind of input fields is not regulated.

Note: One input will be created for each continuous parameter in the *Neural Network* structure, whereas each discrete one will be provided with the inputs the number of which will comply with the number of different unique values of this parameter.

Ports

Input

- — **Input data source** (data table).

Requirements to the Received Data

The input data set fields that will be used as the **input** or **output** ones, must not contain null values. If this requirement is not met, error message will appear when node activating.

Output

- — **Neural network output** (data table). Also refer to output data set structure.
- — **Model quality indicators**.

Wizard

Step 1. Usage Type of Input Columns

It is required to set the usage type of the input data set fields at the first stage. It is required to select one of the following usage types for each of the fields:

- **Input**: the field contains the values of one of the input parameters.
- **Output**: the field contains the class values.
- **Unused**: the field is not included into processing. It is set for other fields by default.

Step 2. Configuration of Normalization

At this stage, the input data is set to the normal kind. It is transformed from the natural values to the non-dimensional ones to ensure that the data with wide scatter of values does not prevail over the data with less significant scatter of values. Usage of **normalization** increases the quality and speed of the *Neural Network* training.

Step 3. Partitioning

At this stage, the input data can be divided into the **test**, **training** and **validation** sets (samples).

Partition Method

- Random: the data for the test and training sets is formed from the whole volume of the input data in random order and proportion.
- Consequent: data ratio for the test and training sets is customized. In this case, it is possible to select the order in which the data is selected first for the test set and then for the training one, or vice versa.

Validation Method

- No validation: validation of the Neural Network training is not performed.
- K-fold **cross validation**: the input data is divided into K folds (parts). The neural network is trained on the basis of all folds, with the exception of one that is used for validation. The training process is repeated K times, and a new fold is used for the validation purposes for

each iteration.

- **Monte Carlo**: the input data is randomly divided into the training and validation set according to the set proportion. The training process is repeated according to the set count of resampling iterations, and a new validation set (resample) is generated for each iteration.

Step 4. Configure Neural Network Parameters

Neural Network Structure

- Number of hidden layers is selected from the list:
  - No hidden layers.
  - One hidden layer (used by default).
  - Two hidden layers.
- Number of neurons in the first hidden layer — integer >= 1 (by default = 10).
- Number of neurons in the second hidden layer — integer >= 1 (by default = 10).

Output Value Limit

The limit type defines the form of **activation function** of the output Neural Network layer:

- No - linear.
- Interval — hyperbolic tangent.
- Bottom only — exponent truncated from below.
- Top only — exponent truncated from above.

It is also possible to set the upper and lower bounds of the activation function form limit. As a rule, this parameter directly correlates with the normalization parameter of the input data.

Training Parameters

- Number of restarts: number of attempts of the *Neural Network* training (using one and the same set) based on the random initial values of the network weights. Upon completion of all restarts, it is required to select the network that provides the least root-mean-square error of the training set. It must be integer>= 1 (by default = 10).
- Decay degree: the degree of the network weights dependence from each other. The higher this dependence, the stronger impact exerts one input parameter on the other ones. The decay enables to decrease the effective number of the model degrees of freedom, thereby avoiding overfitting. The following options are available:
  - None (0).
  - Very weak (20).
  - Weak (40).
  - Mean (60).
  - Strong (80).
  - Very strong (100).

- Continue training: selection of this checkbox enables to start the model overfitting not from the random values of the Neural Network weights, but from the ones obtained during the last training. In this case, the "Number of restarts" parameter is ignored.

Stop Criteria

The network is trained in the iterative manner. The whole training data set is read for each iteration, and the *Neural Network* weights are changed. This process continues until the relative weights changes are less than the set threshold, or iteration count exceeds the set value.

- Minimum weight change threshold: if the relative change of the weights vector norm is less than the threshold at another training step, the training stops. By default = 0.01.
- Maximum number of epochs: the maximum count of the algorithm training iterations. This parameter is disabled by default. If it is required to limit the training process in time, in this case, it will be stopped upon the set number of epochs even if the training has not reached the optimal point, namely, the minimum weight change threshold has not been reached yet.

Step 3. Configure Auto Selection of Neural Network Parameters

Three structure related parameters can be selected for the neural network:

- Number of hidden layers (0, 1 or 2).
- Number of neurons in each hidden layer.
- Decay degree enables to adjust the model stiffness.

Common Parameters

- **Structure autofit** provides the auto selection of the Neural Network structure:
  - **Start with the specified structure**: usage of the values set on the page of the Neural Network parameters configuration as the initial parameters (refer to Step 2).
- **Decay degree autofit** provides the auto selection of the Neural Network decay degree.
  - **Start with the specified decay degree**: usage of the value set on the page of the Neural Network parameters configuration as the initial *Decay degree*.

> **Note:** If it is required to select parameters for the large input volumes or complex models, it is possible to enable only the structure selection, or only decay degree reducing the time spent on training.

Sampling Parameters

To speed up the autofit process, it is required to set the subsample in which it will be performed:

- **Use a subset of the training set** enables to use a subsample of the **training set**.
  - **Percent sample size**: the size of the training set subsample.

- **Maximum sample size**: the maximum size of the training set subsample.

Auto Stop Criteria

The auto selection process is stopped by default if it is not possible to find better parameters as compared with the detected ones. To limit the operation time, at the same time it is possible to limit also the number of autofit stages and autofit time.

- **Autofit stages not more**: the maximum number of the algorithm steps (0 — restrictions are disabled).
- **Autofit time not more (s)**: the maximum algorithm operation time (0 — restriction is disabled).

> **Note:** In the course of the work it is required to take into account that practically both restrictions can be insignificantly exceeded when using subsample for autofit, as the best full set Neural Network will be trained at the last unrestricted stage.

The optimizer is stopped in some specific cases if, from a theoretical standpoint, the best result is reached. It is the root-mean-square error of the training set equal to 0 both for regression networks, and for the classifier.

By default, *Stop at zero classification error* is also enabled for the classifier:

- **Stop at zero classification error** enables to stop autofit when reaching zero classification error.

> **Note:** *Stop at zero classification error* option can be disabled because the correct classification of all examples does not always mean the best structure of the Neural Network, namely, the optimizer can be allowed to select the network with the best generalization ability (for example, with the least number of neurons or more regularizable) but not necessarily with zero classification error.

Optimization Strategy

The root-mean-square error of the training set is a target function for the optimizer. Under these circumstances, to take into account the cases when several networks provide the results comparable in accuracy, in order to select the network with the simplest structure, the target function value is additionally fined by the multiplier (1+1e-8) slightly different from 1 for each hidden neuron.

The following optimization strategy is used:

- If it is required to select only the decay degree for the set structure:
  1. If the starting point is not set, the decay degree is selected using the "golden section search" method, otherwise, the "hill climbing" method is used.
- If it is required to select only the structure without changing the decay degree.

    1. If the initial structure is not set, it is selected in two stages. First, the number of hidden layers is selected (0, 1 or 2), then, the size of hidden layers is approximately selected using the "golden section search" method if the previous stage result is not equl to 0. In this case, the number of neurons for 2 hidden layers becomes the same at this stage.
    2. The structure is selected for all three parameters at once (number of layers, number of neurons) using the "hill climbing" method from the set or selected starting point.
- If autofit of structure and decay is required:
    1. The structure is selected as in the previous clause. In this case, if the initial decay value is set, it can be used, otherwise, the decay option is disabled.
    2. If the initial decay value is not set, it is selected using the "golden section search" method.
    3. The final autofit stage is implemented using the "hill climbing" method for all four parameters.

The flowgraph (transition graph) of the implemented autofit strategy is shown on the figure below.

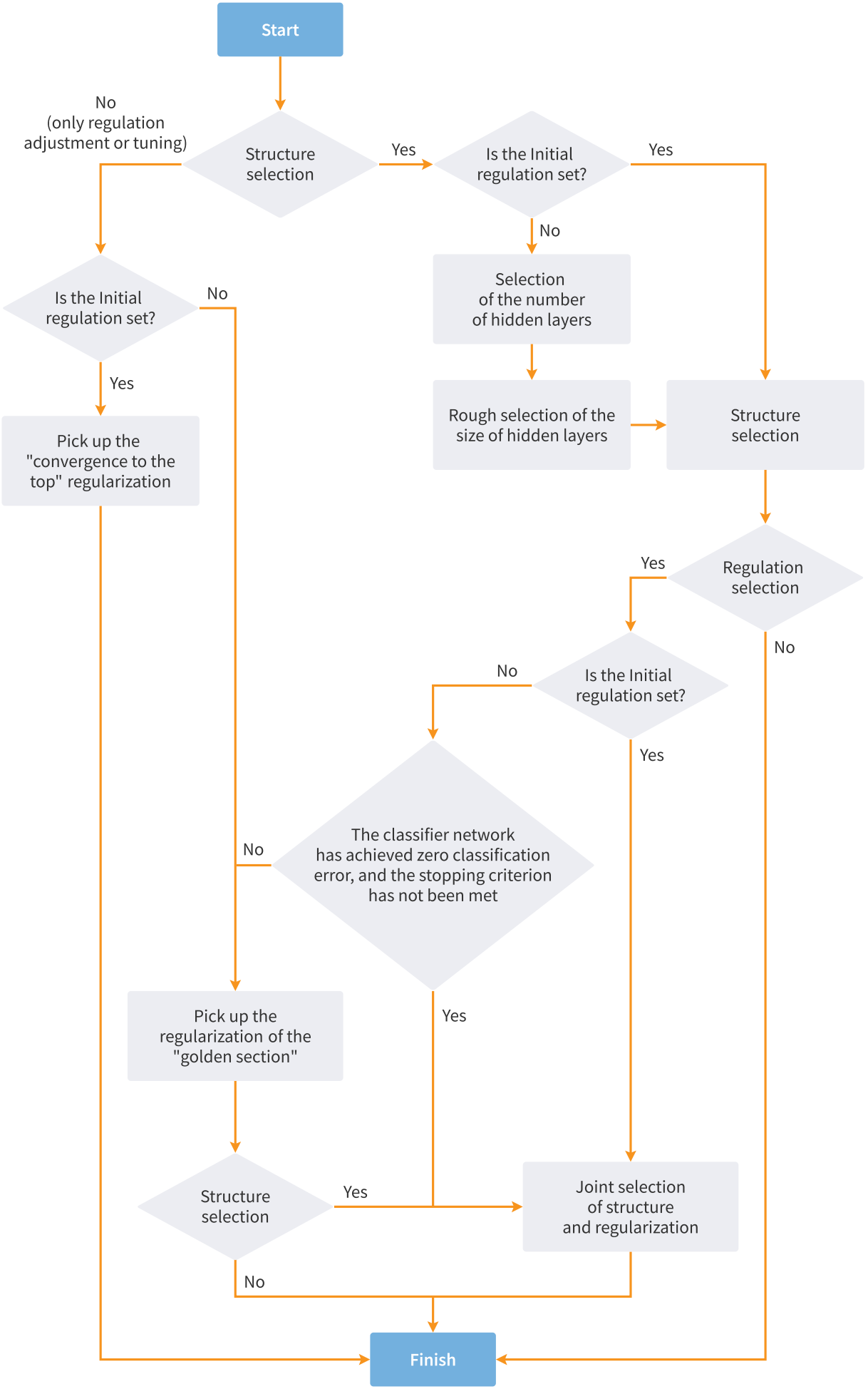
Start
Structure selection
No (only regulation adjustment or tuning)
Yes
Is the Initial regulation set?
Yes
No
Selection of the number of hidden layers
Rough selection of the size of hidden layers
Structure selection
Regulation selection
Yes
No
Is the Initial regulation set?
No
Yes
Is the Initial regulation set?
No
Yes
Pick up the "convergence to the top" regularization
The classifier network has achieved zero classification error, and the stopping criterion has not been met
No
Yes
Pick up the regularization of the "golden section"
Structure selection
Yes
Joint selection of structure and regularization
No
Finish

Figure 1. Auto Selection Algorithm

Articles in Section:

- Report