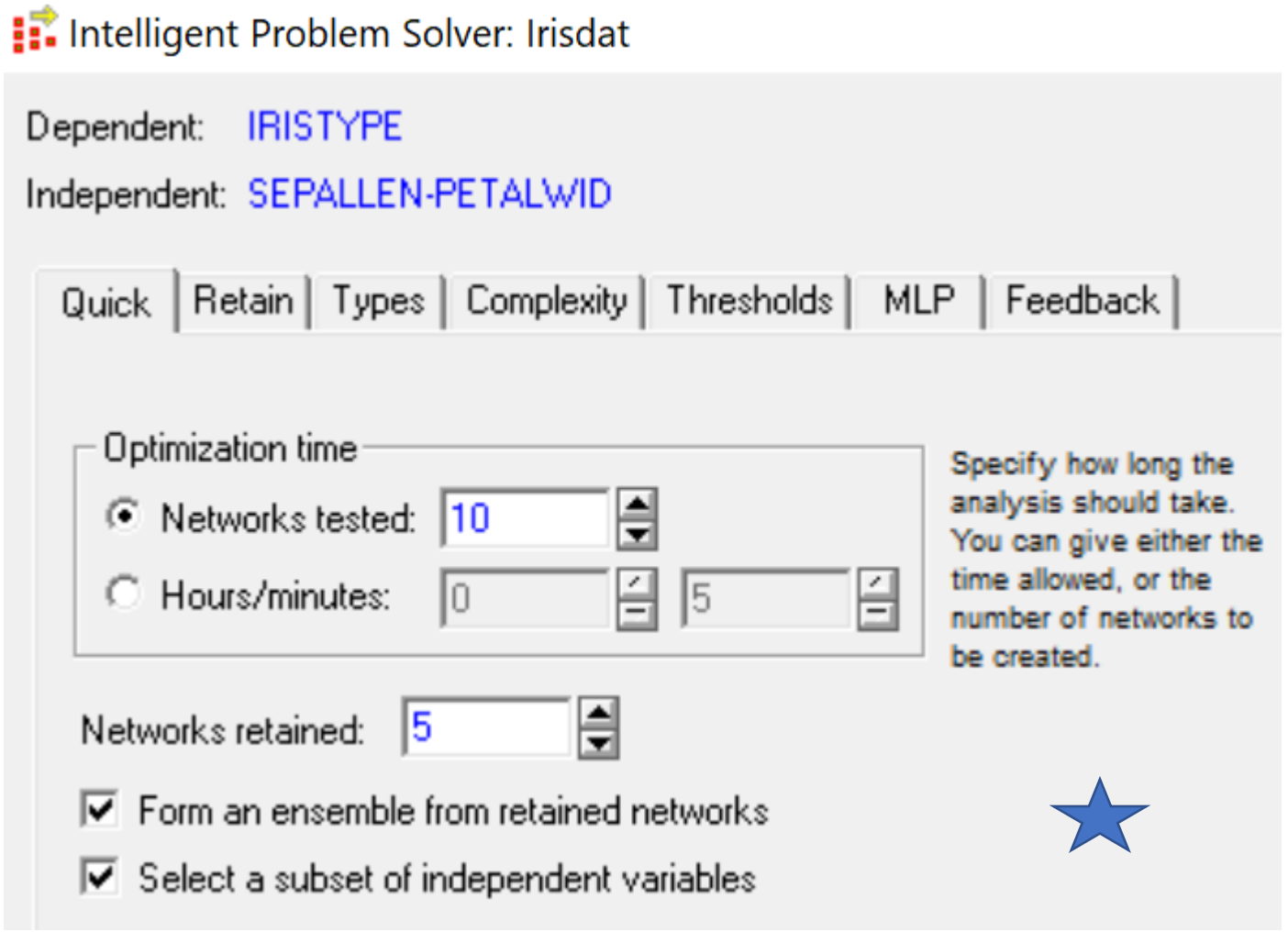
SNN Example 18: Ensembles and Sampling

One of the most prevalent problems with neural networks is the tendency to overfit the data. A range of strategies can be used to combat overfitting, and many of these are available in *STATISTICA Neural Networks*, including **early stopping** and **weight decay regularization**.

Another important approach is to deploy **ensembles of networks**. Rather than making predictions using a single neural network, we can average across the predictions of a number of networks. This can improve the reliability of the prediction, especially if we resample the training set on each network.

Ensembles are effective at improving generalization performance because the networks are optimized against a finite data set. The error made by a neural network can be divided into three parts. First, some error is due to intrinsic noise on the function being modeled - this error is unavoidable. Second, the network may implement a function that is, on average across possible data sets, systematically different to the underlying function. This is known as bias, and is a particular problem if your neural network is insufficiently complex to capture the underlying function. Third, the neural network may be particularly sensitive to the particular choice of data set, and the error may fluctuate accordingly - this is known as variance. More complex models tend to exhibit higher variance, as they can fit to the variance-induced noise in the data set. Selecting the neural network's complexity (i.e. number of hidden units) is actually a choice about the trade-off between bias and variance errors.
Ensembles provide a very simple way of improving performance, because averaging across different neural networks lowers the expected variance. Arguably, you can even afford to deploy more complex models (i.e. more weights) with lower bias, on the basis that the resulting variance can be removed via the ensemble. An important piece of theory (Bishop, 1995) shows that the expected performance of an ensemble is at least as good as the average performance of the members. (Central Limit Theorem)

Creating an ensemble using the Intelligent Problem Solver.



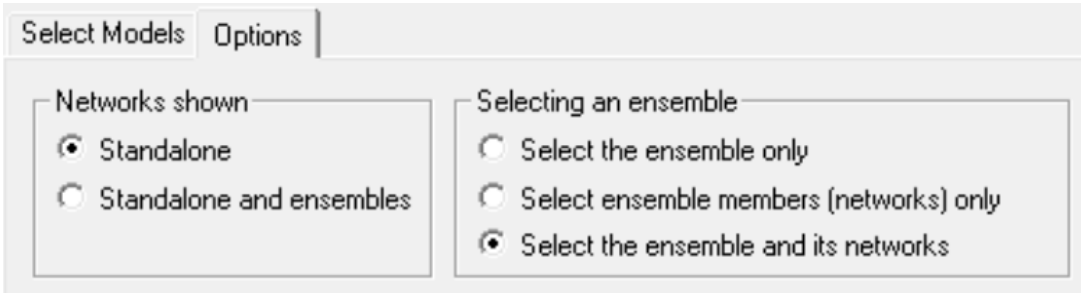
When the Results dialog is displayed, you will see that the summary list contains a single model, with a Profile of Output 4:[5]:1 (or something similar). This signifies an output ensemble, with four inputs, one output, and five member networks.

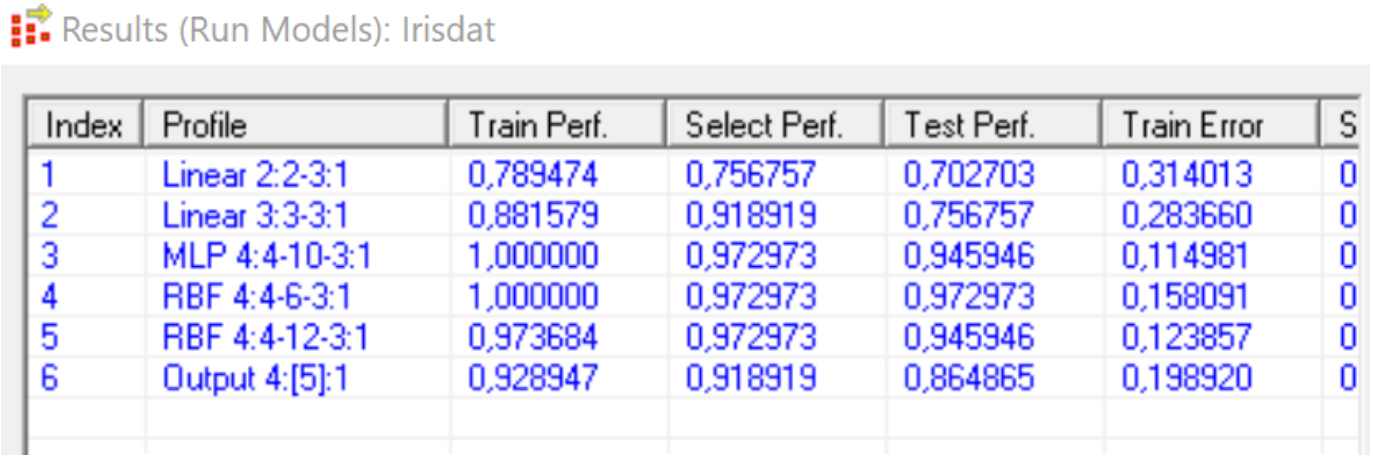
Model Summary Report (Irisdat)

Index	Profile	Train Perf.	Select Perf.	Test Perf.	Train Error	Select Error	Test Error	Training/Members	Note	Inputs	Hidden(1)	Hidden(2)
6	Output 4:[5]:1	0,928947	0,918919	0,864865	0,198920	0,220836	0,278290	1-5		4	5	0

You can execute the ensemble just as you would an individual network, with some minor exceptions. Specifically, and as this is a classification problem, the ensemble forms its prediction using a voting strategy. Each network individually predicts the class of the *IRISTYPE* output. The ensemble predicts the most common class from the members. If there is a tie (disagreement between the members) the ensemble output a missing value (i.e. the class is "unknown"). Due to this voting process, the ensemble cannot output confidence levels (there is a special form of ensemble, a Confidence ensemble, that can, but it requires fairly homogeneous member networks).

The summary statistics displayed for the ensemble at the top of the *Results* dialog are actually the **average statistics** for the member networks. Therefore, an ensemble is also a convenient way to group a set of networks, and generate overall results.
When networks are placed in an ensemble, you can still execute the individual networks.
On the *Results* dialog, click the *Select models* button to display the *Select Networks and/or Ensembles* dialog.

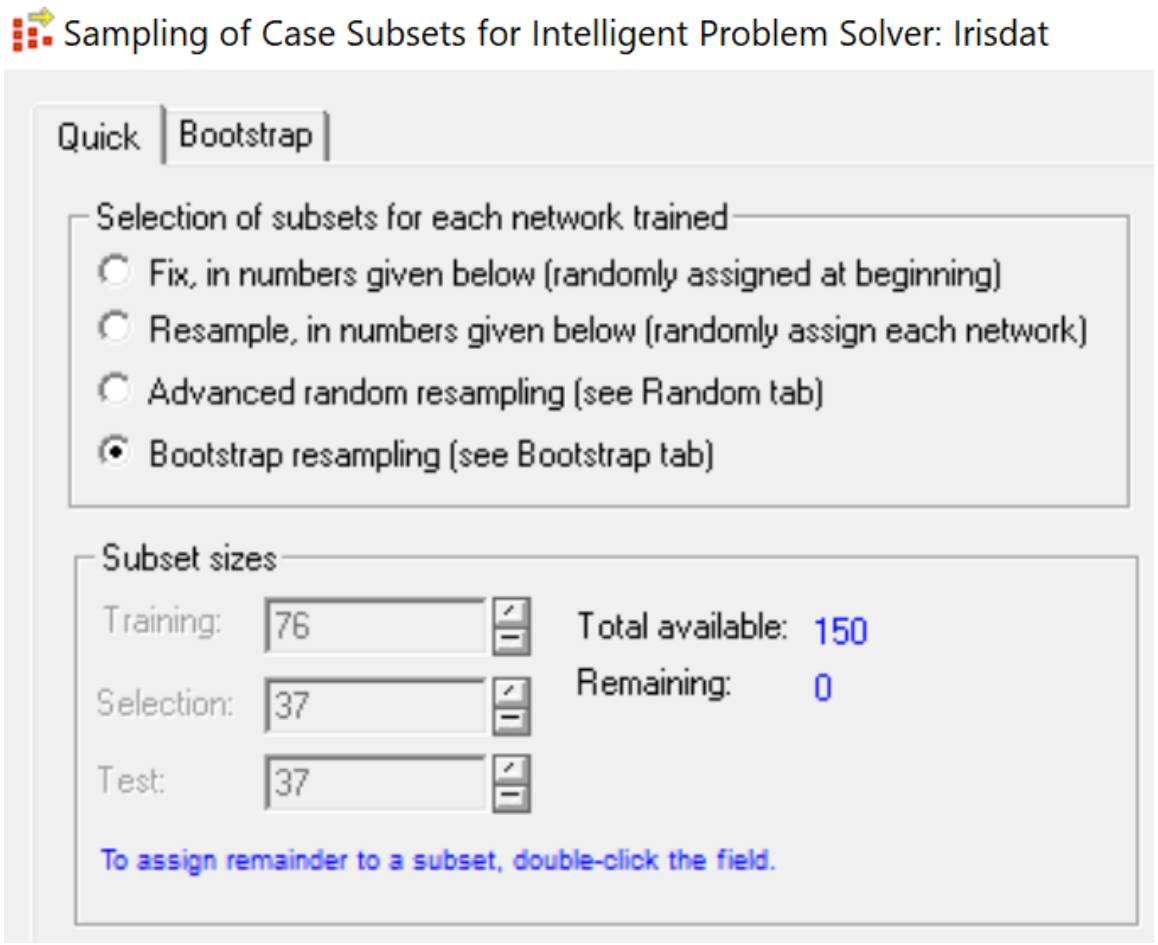




Index	Profile	Train Perf.	Select Perf.	Test Perf.	Train Error	S
1	Linear 2:2-3:1	0,789474	0,756757	0,702703	0,314013	0
2	Linear 3:3-3:1	0,881579	0,918919	0,756757	0,283660	0
3	MLP 4:4-10-3:1	1,000000	0,972973	0,945946	0,114981	0
4	RBF 4:4-6-3:1	1,000000	0,972973	0,972973	0,158091	0
5	RBF 4:4-12-3:1	0,973684	0,972973	0,945946	0,123857	0
6	Output 4:[5]:1	0,928947	0,918919	0,864865	0,198920	0

The resulting spreadsheet contains the predictions of the member networks, and the ensemble, together for comparison. A portion of that spreadsheet is shown below. Remember that your results may vary slightly.

	Prediction (6) (Irisdat)						
	IRISTYPE	IRISTYPE.1	IRISTYPE.2	IRISTYPE.3	IRISTYPE.4	IRISTYPE.5	IRISTYPE.6
1	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA
2	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
3	VERSICOL	VIRGINIC	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL
4	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
5	VIRGINIC	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL
6	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA
7	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
8	VERSICOL	VIRGINIC	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL
9	VERSICOL	VIRGINIC	VIRGINIC	VERSICOL	VERSICOL	VERSICOL	VERSICOL
10	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA
11	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL
12	VERSICOL	VERSICOL	VERSICOL	VIRGINIC	VERSICOL	VIRGINIC	VERSICOL
13	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
14	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL
15	VIRGINIC	VERSICOL	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
16	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
17	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC
18	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA	SETOSA
19	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL	VERSICOL
20	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC	VIRGINIC



This displays the Sampling of Case Subsets for Intelligent Problem Solver dialog.

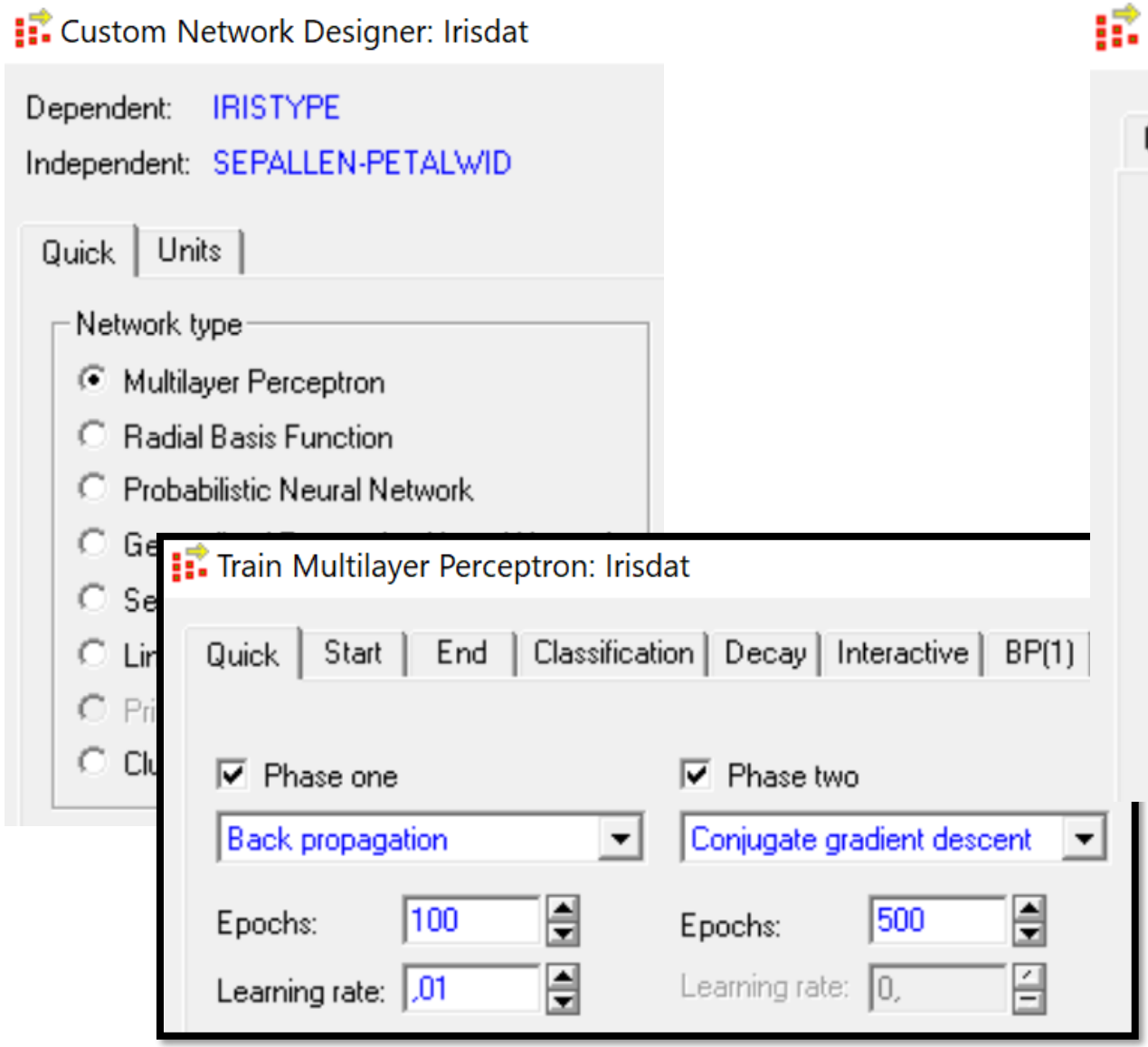
By default, the Selection of subsets for each network trained option is set to Fix in numbers given below (randomly assigned at beginning). This implies that the case subsets are sampled once when the Intelligent Problem Solver starts, and are not altered thereafter. You can change this so that the cases are resampled for each network. For example, select the Bootstrap resampling (see Bootstrap tab) option, to train the network.

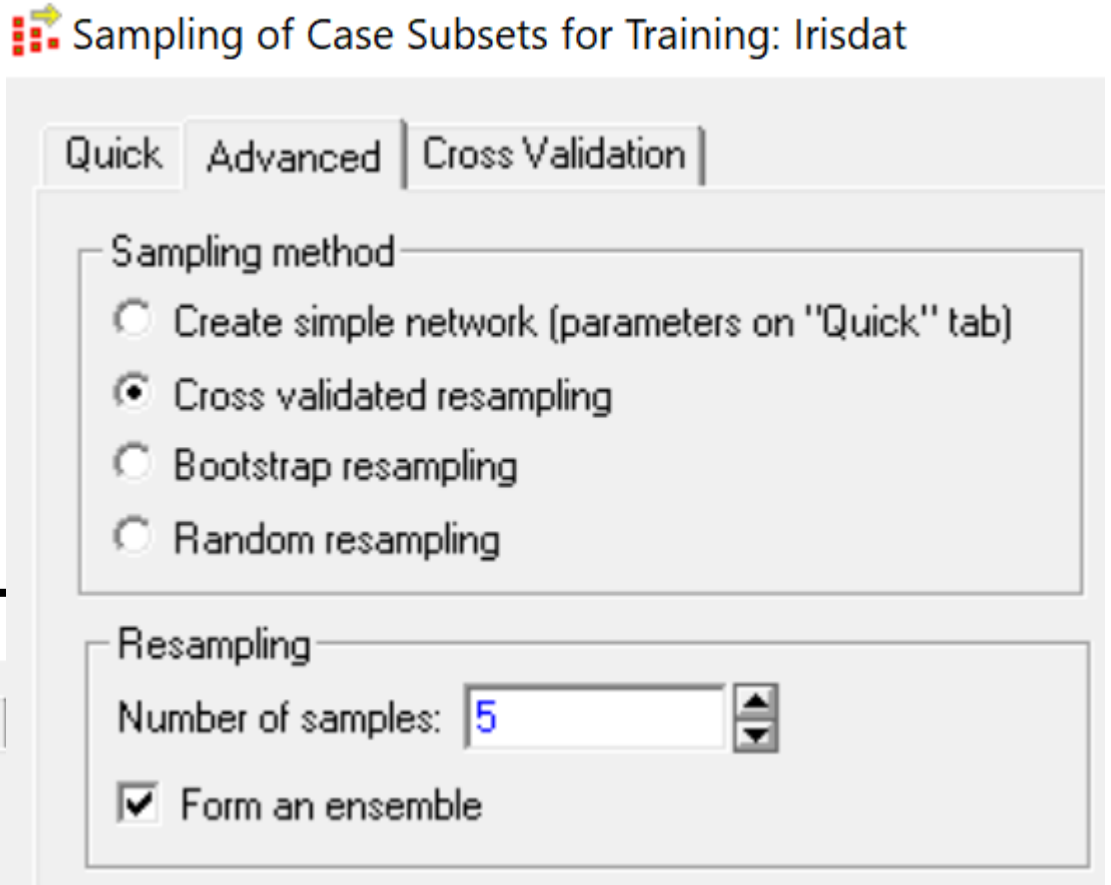
Bootstrap is an effective resampling technique for variance reduction. Each time a network is trained, the training set is produced by sampling with replacement from the original data (i.e. some cases may not be selected, and some may be selected more than once). This process acts to reduce the variance caused by the particular data set available. The ensemble averages across the resulting networks to produce the final prediction. (Remove outliers to train)

Model Summary Report (Irisdat)

Index	Profile	Train Perf.	Select Perf.	Test Perf.	Train Error	Select Error	Test Error	Training/Members	Note	Inputs	Hidden(1)	Hidden(2)
6	Output 4:[5]:1	0,910000	0,896000	0,855759	0,248190	0,263085	0,248789	1-5		4	5	0

Creating an ensemble using the Custom Network Designer. You may also create ensembles using the Custom Network Designer. Besides the usefulness of ensembles for predictions, you can also use the facility to generate estimates of generalization performance, by averaging across repeated sampling experiments.





STATISTICA Neural Networks will now run the training algorithm five times, each time resampling. When training is finished, the five networks are formed into an ensemble, and their average performance statistics are displayed in the summary line in the list box.

In this particular experiment, we have used the **cross validation sampling approach**. The data set is divided into equal parts (in this case, five parts), and a number of experiments run. On each run, one part is held back for use as a test set, and the other parts are used for training and selection. With the five-fold cross validation we used here, 80% of the data set is used for training each network, and 20% for testing. Higher fold testing (even up to the limit of leave-one-out cross validation) can be used, on the basis that the reliability of the individual models will be higher if each uses as much data as possible.

Compared with sampling and ensemble creation in the *Intelligent Problem Solver*, the *Custom Network Designer* is both more flexible and more controlled. You can specify an exact network architecture, then built multiple networks using different case divisions.

	Prediction (6) (Irisdat)	
	IRISTYPE	IRISTYPE.6
1	SETOSA	SETOSA
2	VIRGINIC	VIRGINIC
3	VERSICOL	VERSICOL
4	VIRGINIC	VIRGINIC
5	VIRGINIC	VERSICOL
6	SETOSA	SETOSA
7	VIRGINIC	VIRGINIC
8	VERSICOL	VERSICOL
9	VERSICOL	VERSICOL
10	SETOSA	SETOSA
11	VERSICOL	VERSICOL
12	VERSICOL	VIRGINIC
13	VIRGINIC	VIRGINIC
14	VERSICOL	VERSICOL
15	VIRGINIC	VIRGINIC
16	VIRGINIC	VIRGINIC
17	VIRGINIC	VIRGINIC
18	SETOSA	SETOSA
19	VERSICOL	VERSICOL
20	VIRGINIC	VIRGINIC

Index	Model Summary Report (Irisdat)											
	Profile	Train Perf.	Select Perf.	Test Perf.	Train Error	Select Error	Test Error	Training/Members	Note	Inputs	Hidden(1)	Hidden(2)
6	Output 4:[5]:1	0,975000	0,985000	0,953333	0,205824	0,138710	0,267196	1-5		4	5	0