An Exploratory Study of Artificial Neural Networks for Function Approximation

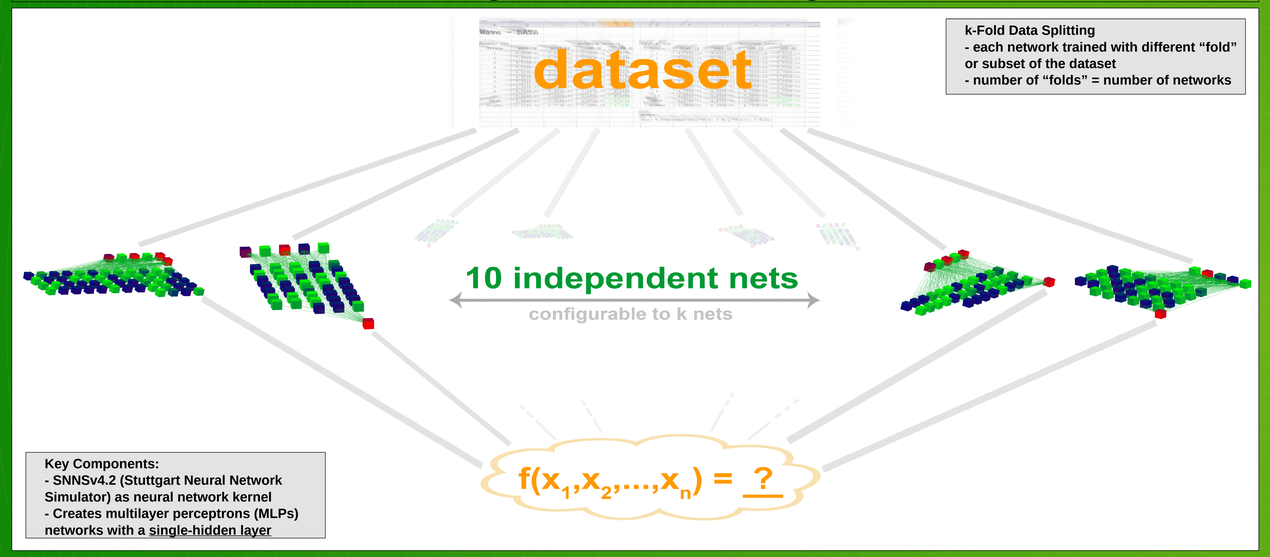
Christopher K. Chan, Princeton University
CASC, Charles H. Tong, Lawrence Livermore National Laboratory

In evaluating the ability of artificial neural networks (ANN) to approximate functions of arbitrary dimensions, we developed a robust and parameter-free implementation of the Stuttgart Neural Network Simulator (SNNS) package that employs k-fold data splitting and aggregated prediction to accurately model data.

Introduction

Function approximation is useful for many engineering and scientific analyses such as sensitivity studies, design optimization and visualization. Popular techniques include regression, splines, kriging and artificial neural networks (ANN). The purpose of this study is to compare the ANN method with the popular Multivariate Adaptive Regression Splines (MARS) developed by J. Friedman.

Diagram of ANN Package



Improvements to Standard ANN Regression Schemes

- To eliminate the need for model selection:
  - uses the Cascade-Correlation algorithm (with Sibling-Descent modification) to automatically set number of hidden units and the "Quickprop" algorithm to train them
- To reduce the variance of prediction:
  - 10-fold data splitting (similar to 10-fold cross-validation)
  - Aggregated output from 10 independently trained networks (similar to "bagging")
  - RE-Train a network if its MSE > 1.25 stddevs of the average MSE of all networks
- To prevent over-fitting: automatically halt training if either stopping measure is satisfied:
  - Criteria 1: If test MSE increases in two successive cycles
  - Criteria 2: If progress of a network is less than 0.01 over six epochs
- To increase prediction accuracy:
  - all input data scaled between [-1.5, 1.5]

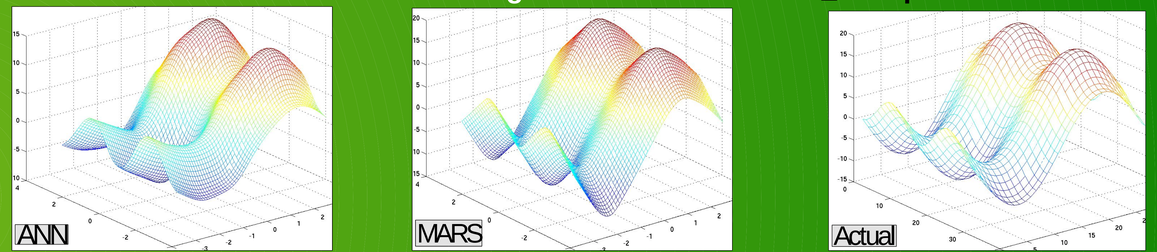
Methods

- Eight datasets were used for testing: five simulated, two from the UCI Repository and one from the Univ. of Toronto. Data ranged from 3 to 32 inputs, but all had a single output.
- MARS and ANN were trained with the same 100 randomly selected points and tested against another 100 randomly selected points. Mean square error (MSE) was averaged over ten runs per data set.
- The procedure was repeated using 200 points for training and 200 points for testing.

Results / Conclusion

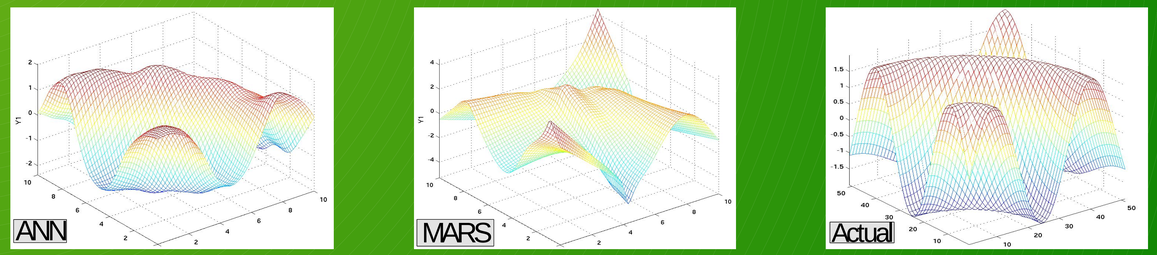
Based on the eight test problems, our ANN package appears to be a competitive approach to function approximation.

- MSE for ANN and MARS showed no significant difference for 6 of 8 problems.



The above surfaces (3-D Ishigami function) illustrate the comparable approximation ability of ANN and MARS on most datasets.

- However, for 2 low-dimensional problems, ANN appeared to give higher approximation accuracy than MARS (the MSE for ANN was half that of MARS)



The above surfaces (2-D Wave function) illustrates the difference in approximation between ANN and MARS on one of the datasets.

- No conclusive trends in MSE variance between ANN and MARS.

Discussion

- In terms of training time required, MARS significantly outperforms ANN.
  - The Quickprop algorithm trains far too slowly on large datasets. Using a conjugate gradient method might reduce training time.
- We used data with relatively low-noise. Would ANN and MARS have performed differently with high-noise data?
- Further optimization of our default settings is definitely possible.

ANN package now integrated into LLNL's PSUADE uncertainty quantification and sensitivity analysis software.

This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48. Research was funded by LLNL's ASC V&V Program.

August 11, 2005