CSCI 447 — Machine Learning: Soft Computing

Project #2

Assigned: September 18, 2017
Design Document Due: October 2, 2017
Project Due: October 16, 2017

This assignment requires you to implement several neural network training algorithms to perform function approximation and compare the results of applying these algorithms to approximate several versions of a function (specified below). In addition to implementing and testing the algorithms, you are required to write a research paper describing the results of your experiments.

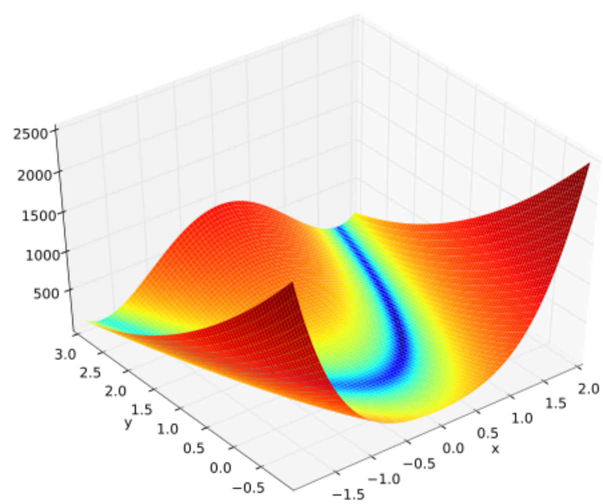
The function approximation problem consists of constructing a model (in this case, learning a neural network) that implements a function

$$f : \mathbf{X} \rightarrow \Re.$$

The trick is that the actual function is never really learned exactly but is approximated (hopefully to within some arbitrary ϵ) by the model. For this assignment, you will approximate five versions of the Rosenbrock function, defined as follows:

$$f(\mathbf{x}) = f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n-1} [(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2].$$

We illustrate this function for $f(x_1, x_2) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2$.



Your assignment consists of the following steps:

1. Prepare a design document addressing the design of two different neural networks for performing function approximation (see below). Be sure to include an explanation of your experimental design as well.

2. Implement a multi-layer feedforward network with backpropagation learning capable of training a network with an arbitrary number of inputs, an arbitrary number of hidden layers, an arbitrary number of hidden nodes by layer, and an arbitrary number of outputs. In other words, the number of inputs, hidden layers, hidden units by layer, and outputs should be furnished as inputs to your program. Be able to specify whether a node uses a linear activation function or a sigmoidal activation function (you may choose between logistic or hyperbolic tangent). Implement learning such that momentum may or may not be used.

3. Implement a radial basis function neural network with an arbitrary number of inputs, an arbitrary number of Gaussian basis functions, and an arbitrary number of outputs. As with the feedforward network, your program should accept the number of inputs, Gaussians, and outputs. It is your choice which output activation function is used.

4. Develop a hypothesis focusing on convergence rate and final performance of each of the chosen algorithms for each of the various problems.

5. Test the MLP algorithm with 0, 1, and 2 hidden layers and the RBF network with at least three different numbers of basis functions on the Rosenbrock function for $n = 2, 3, 4, 5, 6$.

6. Write a paper that incorporates the following elements, summarizing the results of your experiments:

   (a) Title and author name

   (b) A brief, one paragraph abstract summarizing the results of the experiments

   (c) Problem statement, including hypothesis

   (d) Description of algorithms implemented

   (e) Description of your experimental approach

   (f) Presentation of the results of your experiments

   (g) A discussion of the behavior of your algorithms, combined with any conclusions you can draw

   (h) Summary

   (i) References (you should have at least one reference related to each of the algorithms implemented and any other references you consider to be relevant)

7. Submit your design document, fully documented code, sample runs of each algorithm, and your paper.