An Introduction to Soft Computing:

Neural Networks, Evolutionary Computations and Fuzzy Logic

(15 — 17 December 2003)

Akira Imada (Prof. Ph.D.)
Brest State Technical University
e-mail: akira@bstu.by

Still under construction though, this material ws last modified on:

December 17, 2003

Abstract

In real world, we have many problems which we have had no way to solve analytically, or problems which could be solved theoretically but actually impossible due to its necessity of huge resources and/or enormous time required for computation. For these problems, *methods inspired by nature* sometimes work very efficiently and effectively. Although the solutions obtained by these methods do not always equal to the mathematically strict solutions, a near optimal solution is sometimes enough in most practical purposes. These biologically inspired methods are called *Soft Computing*, and here in this course, we study *(1) Neural Networks*, *(2) Evolutionary Computations*, and *(3) Fuzzy Logics* as three representative methods of Soft Computing.

Bibliography

To obtain further overview, the folloings web-sites might be a good start.

- An Introduction to Neural Networks
  - ⋆ http://www.geatbx.com/docu/algindex.html
- Evolutionary Algorithms: Principles, Methods and Algorithms
  - ⋆ http://vv.carleton.ca/˜neil/neural/neuron.html
- Introduction on Evolutionary Algorithms
  - ⋆ http://neo.lcc.uma.es/opticomm/introea.html
- Fuzzy Logic Tutorial - An Introduction
  - ⋆ http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html
- Introduction to Fuzzy Logic
  - ⋆ http://www-ugrad.cs.colorado.edu/˜cs3202/papers/Brigette_Krantz.html

Contents

The 1st day:

Monday, 15 December 2003
(19:20 – 22:10)

Today's Keywords:

neuron-and-synapse, sigmoid-transfer-function, neural-network-as-a-black-box, input-output-relation-as-a-sesory-mortor-system, brain-reaction-according-to-input-from-retina, pattern-recognition-by-pixel-input feature-instead-of-pixel, learning, AND-OR-EXOR-circuit, regression dimensionality-easiness

1 Neural Networks

1.1 What are NNs and What for?

A part of the goal of studying Neural Networks is to learn the mechanism of our brain. Neural Network is made up of *neurons* and *synapses.* We have many variants of Neural Networks, based on how neurons are connected. In this course, however, we employ Neural Networks as a black box which has a number of inputs and outputs. The task is to classify objects. For example,

- We can recognize handwritten characters by giving pixel values as inputs
- We can classify coins inserted into Coke-machine by giving some *features* like diameter and weight of the coin as inputs.
- We can identify a jet fighter as enemie's by a set of data from radar image.

All what we have to do is to determine the strength of connection of every synapses called *synaptic weight*. For the purpose, we adjust each of the weight values starting with a set of random values by giving a number of example inputs. This is called a *learning* of Neural Networks, and most popular learning algorithm is called *back propagation.*

Here, in this course, we study (1) "What is Neural Networks?" That is, "What does inside of the above mentioned black-box look like?" (2) "What for?" That is, "To what applications we can apply them?" And (3) "How it learns?" That is, the mechanism of ' 'How they adjust their synaptic weight values?"

1.2 How a NN works?

Output Y of the neuron which receives weighted-sum of the signals X_i from other N neurons is:

$$Y = \mathrm{sgn}(\sum_{i=1}^{N} w_i X_i - \theta)$$

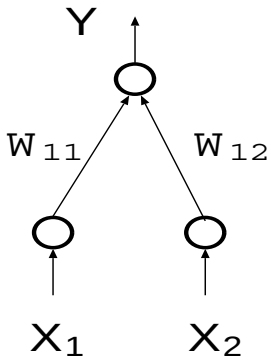
where $\mathrm{sgn}(x) = 1$ if $x \geq 0$ and 0 otherwise, and w_i and θ are called *weight* and *threshold*, respectively.

1.3 Simplest examples

1.3.1 NN to solve AND logics

AND

X_1	X_2	Y
0	0	0
0	1	0
1	0	0
1	1	1



Exercise 1 *Determin two weights w_{11} and w_{12} so that the NN function as AND logic.*

Exercise 2 *Construct OR logic in the same way.*

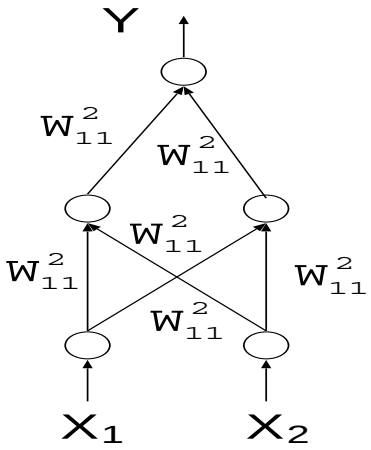
Exercise 3 *Then what about XOR?*

1.3.2 NN to solve XOR logics

To realize XOR, we need one additional layer called *hidden layer*.

XOR

X_1	X_2	Y
0	0	0
0	1	1
1	0	1
1	1	0



Exercise 4 *Obtain six weights values so that the above NN function as XOR.*

1.4 Learning of NN

This section is now under construction. Please be patient.

The 2nd day:

Tuesday, 16 December 2003
(19:20 – 22:10)

Today's Keywords:

adenine-thymine-guanine-cytosine,chromosome, gene, selection, fitness, recombination, crossover, mutation, population, generation, Traveling-Salesperson-Problem, Knap-sack-Problem, Search-for-a-needle-in-haystack.

2 Evolutionary Computations

To solve a problem, in most cases, means to search for an appropriate set of parameters. For example, when we want to express a function of $x \approx 0$ with a power series of x, our task is to find 10 appropriate values of a_i of the equation

$$f(x) \approx \sum_{i=1}^{10} a_i x^i.$$

Or when we want to make a neural network classify objects properly, our task is to find out an appropriate configuration of synaptic weights, as we mentioned above.

In Evolutionary Computation, In order for us to be ablet to solve this kind of problems, It is required, first of all, that we can create a set of candidate solutions at random. This set of random candidate solution is called a *population* of the *1st generation*. Typically, a candidate solution is expressed as a single string of parameters. We call this string *chromosome* and each of its entry *gene*, This is the first condition under which we can solve the problem by Evolutionary Computations. That is, It is necessary to be able to express candidate solution with a single string. The second condition is that we should be able to evaluate the degree to which how good is each of these chromosomes, which is called *fitness evaluation*.

Then we select somewhat of a better two parents chromosomes than others, and create one child chromosome using biological analogy of crossing their *genes* (crossover) and occasionary by replacing some of the genes with other random genes (mutation). This procedure of selecting parents and reproducing children is repeated until the number of children reaches the population. Thus we can expect better chromosome to appear from generation to generation, and eventually find an optimum chromosome.

In this course, (1) we study the algorithm more in detail, that is, "What kind of selection, crossover and mutation scheme we have?" And (2) we apply this method to as many different kinds of problems as possible.

2.1 NN for XOR revisite

Exercise 5 *Create Pseudo code for EC to obtain the six weights above.*

2.2 On what condition we need to apply EC?

This section is now under construction. Please be patient.

The 3rdd day:

Wednes, 17 December 2003
(19:20 – 22:10)

Today's Keywords:

crisp-set, fuzzy-set, membership-function, triangle-memebership-function, trapezoid-memebership-function, Cauchy-memebership-function, singleton, AND/OR/NOT-of-fuzzy-sets, relation-of-2-sets, composition-of-relations, membership-function-of-IF-THEN-rules,

3 Fuzzy Logic

The goal of Fuzzy Logic is to design intelligent system based on our human knowledge which can be described by our natural language using so-called IF-THEN rules. A toy example of our knowledge is

- IF the apple is red THEN buy it OTHERWISE do not buy it.

Human knowledge or fact in real world, however, is approximate rather than exact, something like

- IF the apple is red THEN it is sweet, possibly sweet-sour, and unlikely to be sour.

Or what would be an answer for

- Now an apple is more or less red then what does the taste seem to be?

In classic logic when we use set theory (which we now call *crisp set*) an element either belongs to a set or not. The apple in the first statement above must either belong to a set RED-APPLE or not. On the other hand, Fuzzy Logic concerns the *degree of belonging* which is expressed using a *membership function* whose value ranges from 0 (no possibility to belong) to 1 (sure to belong), while in crisp set the value is either 1 (belong) or 0 (not belong).

Here, in this course, we study (1) "What is fuzzy set and its membership function?" (2) "How we express our knowledge using fuzzy set?" (3) "How we combine multiple fuzzy set (using like AND and OR operations in crisp set)?" (4) "How we express our knowledge with Fuzzy Logic?" and (5) "How we control a system with Fuzzy Logic?"

3.1 Crisp-set or Fuzzy-set?

- {all positive integer small than 3.}
- {all real number much small than 10.}
- {all real number close to 12}

3.2 Membership function

possible among many altanatives are:

3.3 AND, OR, Compliment