Intelligent Finance
Information System

Subject 4: Hybrid System
Hybrid intelligent systems

- Evolutionary Neural Networks
- Fuzzy Evolutionary Systems
- Neural Fuzzy Systems
Evolutionary neural networks

Limitation of Neural Network

- Although neural networks are used for solving a variety of problems, they still have some limitations. One of the most common is associated with neural network training. The back-propagation learning algorithm cannot guarantee an optimal solution as the back-propagation algorithm might converge to a set of sub-optimal weights from which it cannot escape. As a result, the neural network is often unable to find a desirable solution to a problem.
Evolutionary neural networks

Limitation of Neural Network

- Another difficulty is related to selecting an optimal topology for the neural network. The network architecture for a particular problem is often chosen by means of heuristics, and designing a neural network topology is still more art than engineering.

- Genetic algorithms are an effective optimisation technique that can guide both weight optimisation and topology.
Evolutionary neural networks

- The architecture of the network (i.e. the number of neurons and interconnections) often determines the success or failure of the application.

- Usually the network architecture is decided by trial and error; there is a great need for a method of automatically designing the architecture for a particular application. Genetic algorithms may well be suited for this.
Evolutionary neural networks

- To apply the GA, firstly require to define a fitness function for evaluating the chromosome’s performance. This function must be able to estimate the performance of a given neural network.

- Then we can apply here a simple function defined by the sum of squared errors. The training set of examples is presented to the network, and the sum of squared errors is calculated. The smaller the sum, the fitter the chromosome.
Evolutionary neural networks

- The genetic algorithm attempts to find a set of weights that minimise the sum of squared errors.

- The next step is to choose the genetic operators such as crossover and mutation.
Evolutionary neural networks

- A crossover operator takes two parent chromosomes and creates a single child with genetic material from both parents. Each gene in the child’s chromosome is represented by the corresponding gene of the randomly selected parent.

- Crossover produces new individuals that have some parts of both parent’s genetic material. The simplest form of crossover is that of single-point crossover.
Evolutionary neural networks

- mutation is a random process where one allele of a gene is replaced by another to produce a new genetic structure. In GAs, mutation is randomly applied with low probability, typically in the range 0.001 and 0.01, and modifies elements in the chromosomes.

- A mutation operator selects a gene in a chromosome and adds a small random value between -1 and 1 to each weight in this gene.
Fuzzy evolutionary systems

- Evolutionary computation is also used in the design of fuzzy systems, particularly for generating fuzzy rules and adjusting membership functions of fuzzy sets.

- The problem of selecting fuzzy IF-THEN rules can be seen as a combinatorial optimisation problem with two objectives. One objective is to maximise the number of correctly classified patterns and the other objective is to minimise the number of rules.
Fuzzy evolutionary systems

- Genetic algorithms can be applied to this problem. A basic genetic algorithm for selecting fuzzy IF-THEN rules includes the following steps:

- Step 1: Randomly generate an initial population of chromosomes. The population size may be relatively small, say 10 or 20 chromosomes. Each gene in a chromosome corresponds to a particular fuzzy IF-THEN rule in the rule set.
Step 2: Calculate the performance, or fitness, of each individual chromosome in the current population. The problem of selecting fuzzy rules has two objectives: to maximise the accuracy of the pattern classification and to minimise the size of a rule set.

The fitness function has to accommodate both these objectives.
Fuzzy evolutionary systems

- **Step 3:** Select a pair of chromosomes for mating. Parent chromosomes are selected with probability associated with their fitness; a better fit chromosome has a higher probability of being selected.

- **Step 4:** Create a pair of offspring chromosomes by applying a standard crossover operator. Parent chromosomes are crossed at the randomly selected crossover point.
Fuzzy evolutionary systems

- Step 5: Perform mutation on each gene of the created offspring. The mutation probability is normally kept quite low, say 0.01. The mutation is done by multiplying the gene value by \( -1 \).

- Step 6: Place the created offspring chromosomes in the new population.
Fuzzy evolutionary systems

- Step 7: Repeat *Step 3* until the size of the new population becomes equal to the size of the initial population, and then replace the initial (parent) population with the new (offspring) population.

- Step 8: Go to *Step 2*, and repeat the process until a specified number of generations (typically several hundreds) is considered.

- The number of rules can be cut down to less than 2% of the initially generated set of rules.