

What is the foundation of genetic algorithms in machine learning?

Charles Darwin's idea of natural evolution served as an inspiration for genetic programming in machine learning. It showcases how natural selection works in humans, where the fittest individuals are chosen for reproducing offspring to take the present generation ahead.

To put it simply, it refers to the survival of the fittest for solving a problem amidst the individuals of consecutive generations.

How genetic algorithm works?

In order to understand the simple genetic algorithm and how it works, there are some basic terminologies that will help you understand it better. We have described them below.

- Genetic operators: In genetic algorithms, genetic operators are used when you wish to change the genetic composition of the next generation.
- Chromosome/Individual: It refers to the collection of genes that can be represented with a string of each bit as a gene.
- Population: Each chromosome represents an individual and a collection of chromosomes/individual make up the population.
- Fitness function: This function in genetic algorithms produces an improved output for a specific input.

Now let's move on to understand the working of genetic algorithms in machine learning.

There are five phases that illustrate the entire process of how this algorithm works.

How Genetic Algorithm Works:



1. Initialization

The working of genetic algorithms starts with the process of initialization where a set of individuals is generated that we refer to as population. It contains a set of parameters called genes that are combined into a string and generate chromosomes. These chromosomes are the solution to the problem that are derived by the technique of random binary strings.

2. Fitness assignment

The fitness function of the genetic algorithm determines an individual's ability to compete with other individuals. It provides the score to each individual that determines its probability of being selected for the process of reproduction.

Higher the fitness score, greater are the chances of an individual getting selected for reproduction.

3. Selection

In this phase, out of all the steps of genetic algorithm, individuals are selected to produce offspring by arranging them in a pair of two. Three types of selection methods are leveraged in this process that are listed below.

- Rank based selection
- Tournament selection
- Roulette wheel selection

4. Reproduction

In the reproduction step, the genetic algorithm leverages two variation operators. These are applied to the parent population. These two operators are:

Crossover

In this process, a crossover point is selected within the genes randomly. This operator then swaps the genetic information of the two selected parents or say individuals from the current generation to produce an offspring.

There are various styles for crossover among the parents namely one-point crossover, two-point crossover, livery crossover, and inheritable algorithms crossover.

The genes of the chosen fittest parents are exchanged until the crossover point is met. When the process ends, an offspring is produced that includes genes from both the parents.

Mutation

In this process, a random gene is inserted in the offspring in order to maintain the diversity in the population. There are three types of mutation styles available namely flip bit mutation, gaussian mutation, and exchange or swap mutation.

This operator helps in resolving the premature convergence issue and enhances diversification in the population.

5. Termination

Now that the offspring is produced, the reproduction face is terminated by applying a stopping criterion. The algorithm reaches the end of the process after the threshold fitness solution is reached. Further, it identifies the final yet best solution in the population.



Advantages of the genetic algorithm

The genetic algorithms put a plethora of advantages on the table. Get acquainted with them in the next segment.

Genetic algorithms provide solutions or answers that improve over a period.

It does not need any derivative information and has excellent parallel capabilities.

It optimizes several problems, such as continuous functions, discrete functions, and multi-objective problems.

