Decision Trees in Machine Learning

A tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal, its also widely used in machine learning, which will be the main focus of this article

How can an algorithm be represented as a tree?

For this let's consider a very basic example that uses titanic data set for predicting whether a passenger will survive or not. Below model uses 3 features/attributes/columns from the data set, namely sex, age and sibsp (number of spouses or children along).

Although, a real dataset will have a lot more features and this will just be a branch in a much bigger tree, but you can't ignore the simplicity of this algorithm. The feature importance is clear and relations can be viewed easily. This methodology is more commonly known as learning decision tree from data and above tree is called Classification tree as the target is to classify passenger as survived or died. Regression trees are represented in the same manner, just they predict continuous values like price of a house. In general, Decision Tree algorithms are referred to as CART or Classification and Regression Trees.
So, what is actually going on in the background? Growing a tree involves deciding on which features to choose and what conditions to use for splitting, along with knowing when to stop. As a tree generally grows arbitrarily, you will need to trim it down for it to look beautiful. Lets start with a common technique used for

splitting.

the approches of decesion tree

2.2 GA-DT Algorithm
In this step, the GA algorithm was modified using the DT approach for generating a better initial population. DT was used to choose meaningful descriptors that will increase model fitness. As shown in Figure 1, the initiation of GA uses DT. The DT algorithm checks if the randomly selected descriptors increase the R2 value of the populations. In this way, each member of the population represents a potential model. The first descriptor is a random selection. From the second descriptor and onwards, DT tries to include descriptors such that the R2 value keeps getting better. This ensures not only a good initial population but also a good final model using the fewest number of generations. After obtaining a good initial population, the population goes through Roulette's wheel population substitution.