**Genetic Algorithm**

Genetic Algorithms the individuals are coded as integers. The selection is done by selecting parents proportional to their fitness. So individuals must be evaluated before the first selection is made. Genetic operators work on the bit level (e.g. cutting a bit string into multiple pieces and interchanging them with the details of the other parent or switching single bits). GA has efficient parallel capabilities and does not need derivative information. For applications, it can optimize various problems which are discrete/ continuous functions and multi-objective problems. There are some limitations of GA which are the quality of the final solution is not guaranteed and repetitive calculation of fitness function creates some excessive computational challenges.

**Evolutionary Strategies**

The algorithm of ES, as an innovative method, was first introduced by Rochenberg. In Evolution Strategies the individuals are coded as vectors of real numbers. On reproduction, parents are selected randomly and the fittest offspring are selected and inserted in the next generation. ES individuals are self-adapting. The step size or "mutation strength" is encoded in the individual, so good parameters get to the next generation by selecting good individuals. ES is effectively used in large search space when the best solution is not necessarily required. ES is easier to implement and scale in a distributed computational environment and can create more diverse solutions.

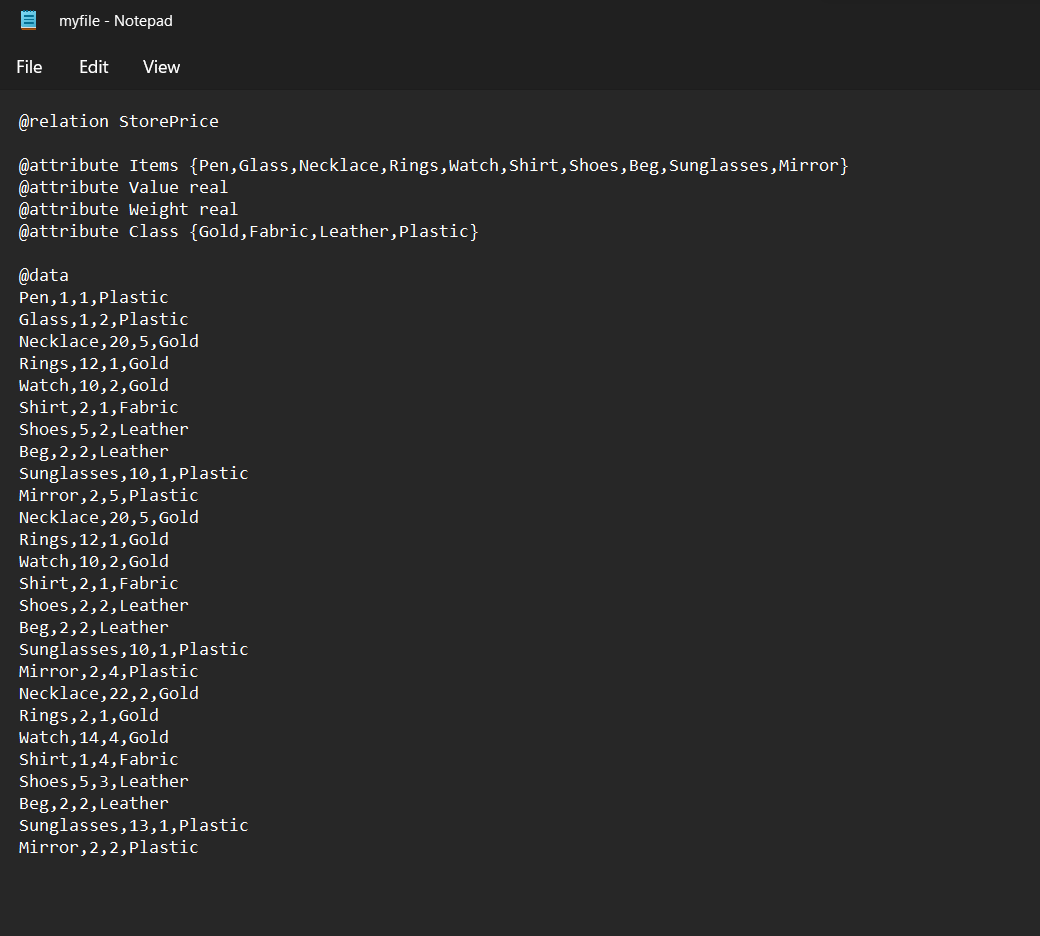
**Differential Evolution**

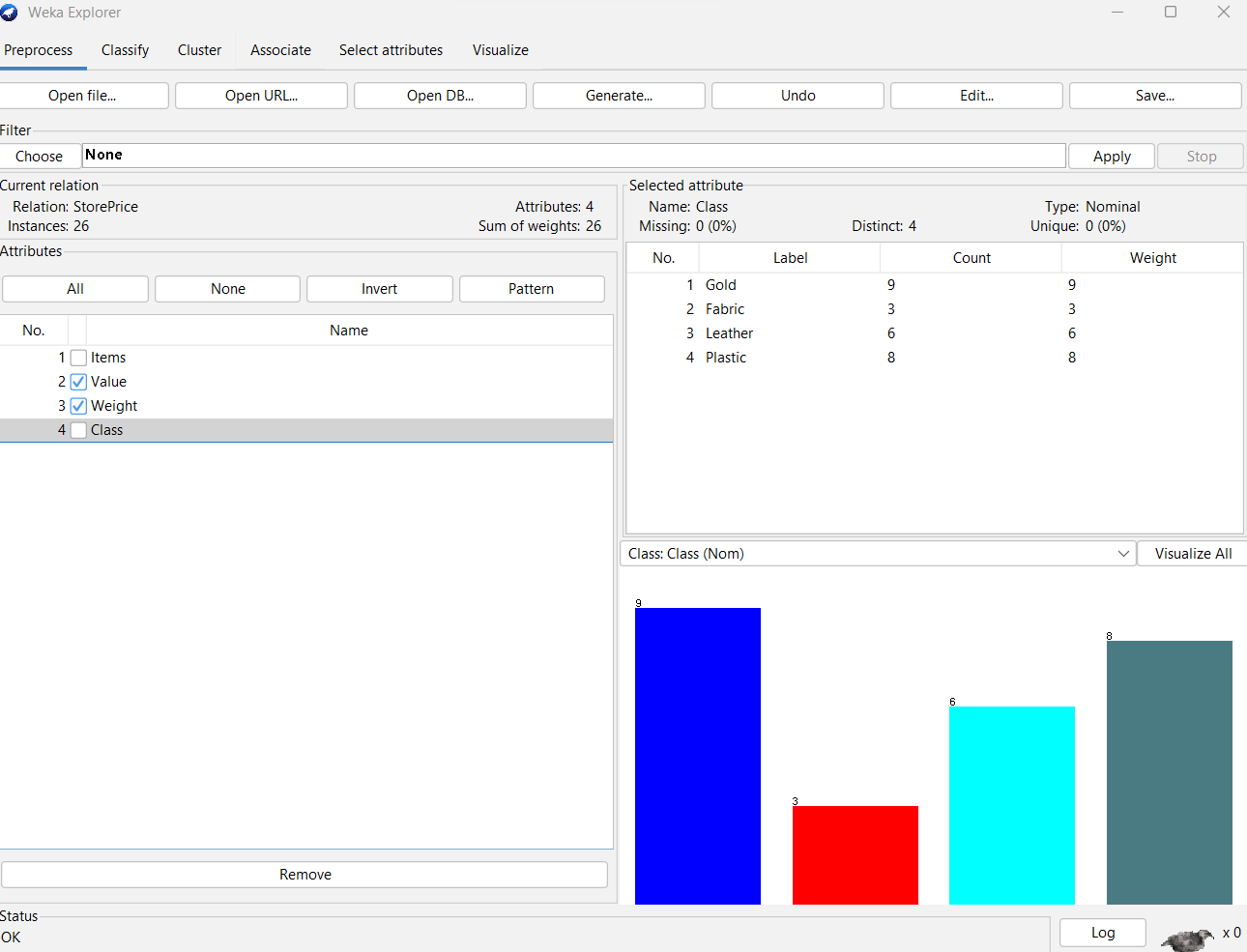
Differential evolution (DE) is a population-based metaheuristic search algorithm that optimizes a problem by iteratively improving a candidate solution based on an evolutionary process. Such algorithms make few or no assumptions about the underlying optimization problem and can quickly explore very large design spaces. Differential Evolution is similar to evolution strategies because it is also coded as vectors of real numbers. The mutation/crossover operations make use of the difference between two or more vectors in the population to create a new vector. DE is usually stuck in local minima just like GA. GA is particularly good at escaping local minima. If the space is very complicated, the mutation operator can be increased to force the GA to explore more aggressively and thus escape these local minima. However, it needs computation power.

**Estimation of Distribution Algorithms**

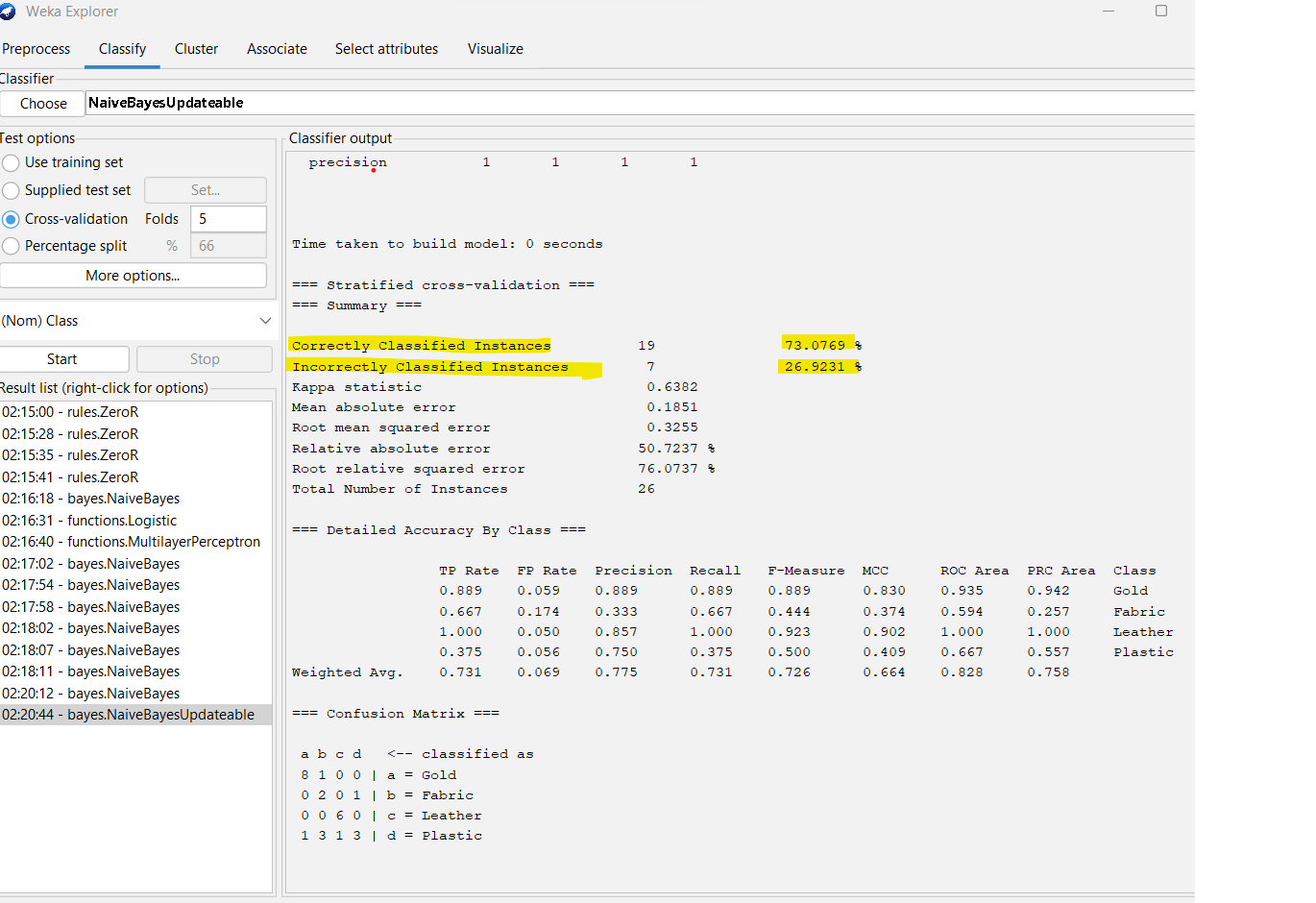
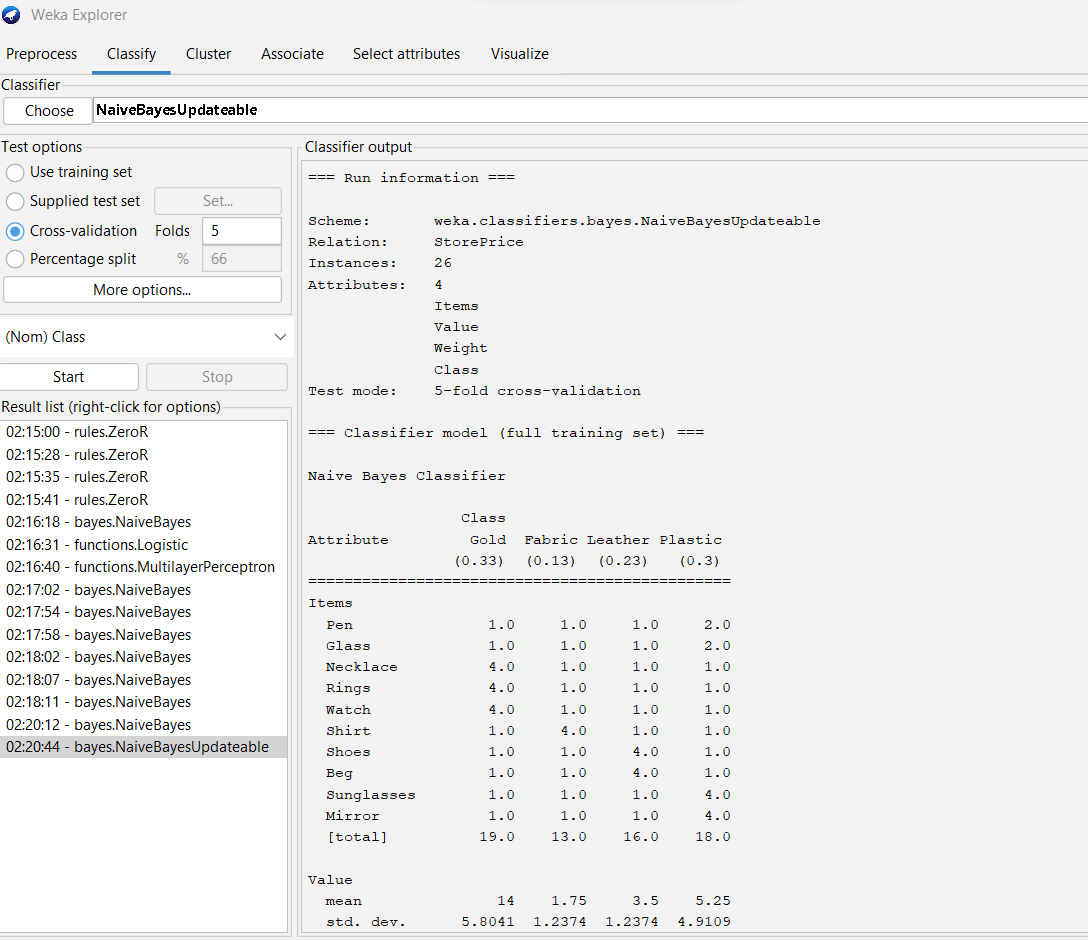
Estimation of distribution algorithms (EDAs) are stochastic optimization techniques that explore the space of potential solutions by building and sampling explicit probabilistic models of promising candidate solutions. This explicit use of probabilistic models in optimization offers some significant advantages over other types of metaheuristics. The model construction of EDA captures the probability distribution of the promising solutions which differs from other metaheuristics approach. In EDA, once the model is constructed, new solutions are generated by sampling the distribution encoded by this model. The new solutions are then incorporated back into the old population or take place entirely. The process is repeated until some termination criteria are met. The main advantage of EDA is that it uses adaptive operators whereas other algorithms use fixed operators. EDA reduced memory requirements by replacing the population of candidate solutions with a probabilistic model.

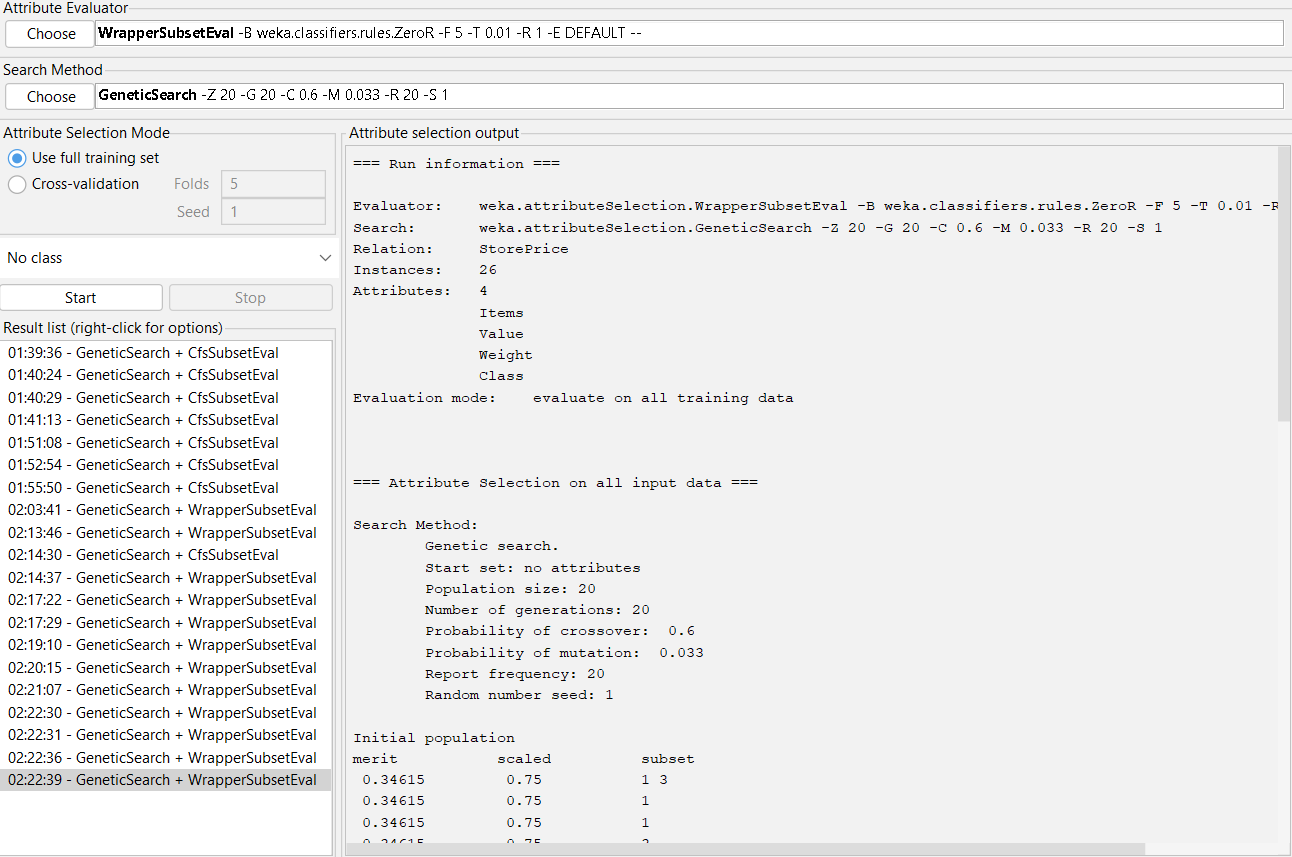
For implementing knapsack problem in the tool. I use WEKA developed at the University of Waikato, New Zealand - an open-source software provides tools for data preprocessing, implementation of several Machine Learning algorithms, and visualization tools. I installed the GeneticsSearch package and perfrom the following steps.

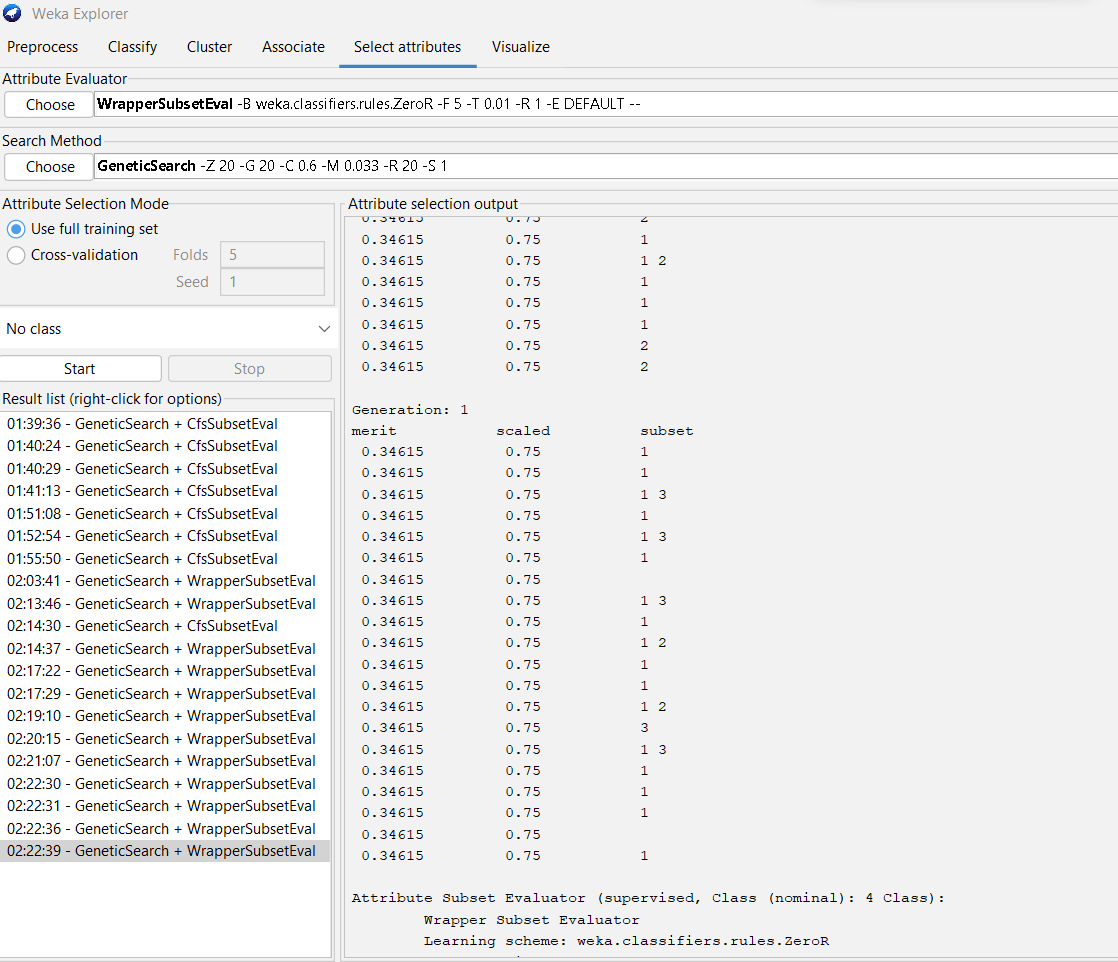
1. First I created a dummy dataset in csv format and change into .rff and modify syntax according to requirements.
2. Imported data select 2 attributes Value and Weight.



1. For classifier I use NaiveBayes.



1. Select GeneticSearch with Wrapper subset evaluator



1. Ploting the result.

