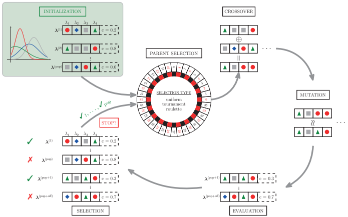
Optimization in Machine Learning

Evolutionary Algorithms - Introduction



Learning goals

- Evolutionary algorithms
- Encoding
- Parent selection, variation, survival selection

EVOLUTIONARY ALGORITHMS

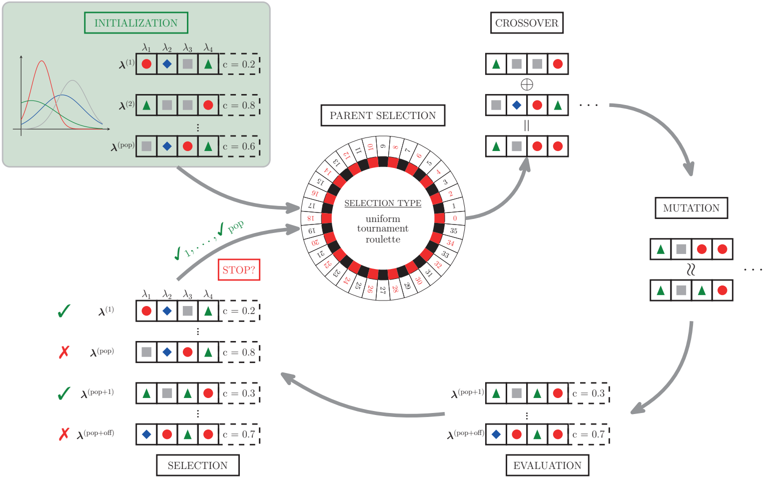
Evolutionary algorithms (EA) are a class of stochastic, metaheuristic optimization techniques whose mode of operation is inspired by the evolution of natural organisms.

History of evolutionary algorithms:

- **Genetic algorithms**: Use binary problem representation, therefore closest to the biological model of evolution.
- **Evolution strategies**: Use direct problem representation, e.g., vector of real numbers.
- **Genetic programming**: Create structures that convert an input into a fixed output (e.g. computer programs); solution candidates are represented as trees.
- **Evolutionary programming**: Similar to genetic programming, but solution candidates are not represented by trees, but by finite state machines.

The boundaries between the terms become increasingly blurred and are often used synonymously.

STRUCTURE OF AN EVOLUTIONARY ALGORITHM



NOTATION AND TERMINOLOGY

- A chromosome is a set of parameters which encodes a proposed solution to the problem that the genetic algorithm is trying to solve. The chromosome is often represented as a binary string, although a wide variety of other data structures are also used.
- The set of all solutions is known as the population.

Symbols	EA Terminology
solution candidate $\boldsymbol{x} \in \mathcal{S}$	Chromosome of an individual
$\boldsymbol{x}_j$	j-th gene of chromosome
Set of candidates P with $\mu = \|P\|$	Population and size
λ	Number of generated offsprings
$f : \mathcal{S} \to \mathbb{R}$	Fitness function

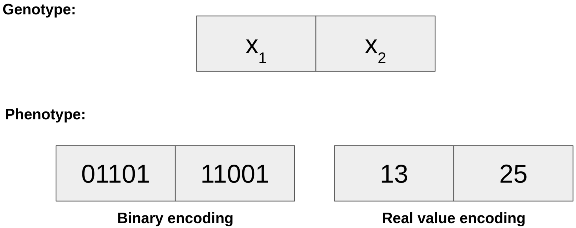
Note: Unintuitively, we are minimizing fitness because we always minimize f by convention.

ENCODING

Encoding of chromosomes is the first step of solving a problem with EAs. Technically: Mapping from **genotype** to **phenotype**. Encoding depends on the problem, and eventually decides performance of problem solving.

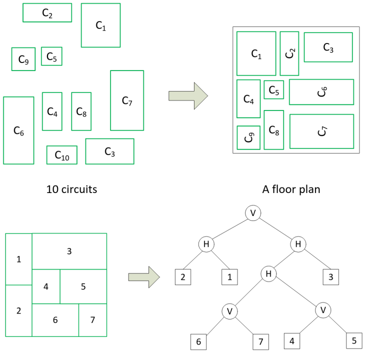
Encoding methods:

- Binary encoding: Strings of 0s and 1s
- Real value encoding: Real values



ENCODING

- Tree encoding: Tree objects



Floor planning problem. Given are *n* circuits of different area requirements. Goal: arrange them into a floor layout so that all circuits are placed in a minimum layout. Each solution candidate can be represented by a tree.
Source: Encoding Techniques in Genetic Algorithms, Debasis Samanta, 2018.

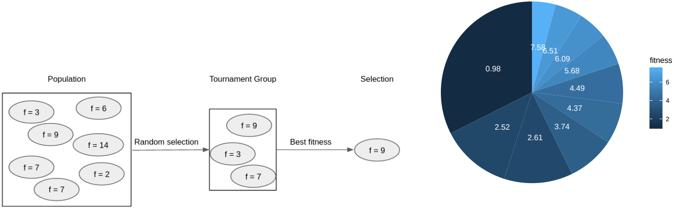
STEP 1: INITIALIZE POPULATION

- An evolutionary algorithm is started by generating an initial population $P = \{\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(\mu)}\}$.
- Usually, we sample this uniformly at random.
- We could introduce problem prior knowledge via a smarter init procedure.
- This population is evaluated, i.e., the objective function is computed for every individual in the initial population.
- The initialization can have a large influence on the quality of the found solution, so many EAs employ *restarts* with new randomly generated populations.

STEP 2: PARENT SELECTION

The first step of an iteration chooses λ parents, that create offspring for the next step.

- **Neutral selection:** choose individual with a probability $1/\mu$.
- **Fitness-proportional** / **Roulette wheel selection:** draw individuals with probability proportional to their fitness.
- **Tournament selection:** randomly select *k* individuals for a "tournament group". Of the drawn individuals, the best one (according to fitness value) is then chosen. In order to draw λ individuals, the procedure must be performed λ-times.



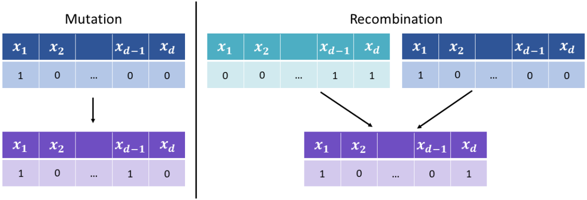
Left: Random selection. Right: Fitness-proportional selection. Fitness values of $\mu = 10$ individuals (labels in pie-chart) are transferred inversely into probabilities (proportions of the pie-chart).

STEP 3: VARIATION

New individuals are now generated from the parent population. This is done by

- Recombination/Crossover: combine two parents into one offspring.
- Mutation: (locally) change an individual.

Recombination and mutation are not always performed both; sometimes only one operation is performed.



Note: The operation depends on the encoding. We will later look into examples for binary and numeric encodings.

STEP 4: SURVIVAL SELECTION

Now individuals are chosen who survive. Two common strategies are:

- (μ, λ)**-selection:** We select from the λ descendants the μ best ($\lambda \geq \mu$ necessary).
  But: best overall individual can get lost!
- $(\mu + \lambda)$**-selection:** μ parents and λ offspring are lumped together and the μ best individuals are chosen.
  Best individual safely survives.

EVOLUTIONARY ALGORITHMS

Advantages

- Conceptually simple, yet powerful enough to solve complex problems (including HPO)
- All parameter types possible in general
- Highly parallelizable (depends on λ)
- Allows customization via specific variation operators

Disadvantages

- Less theory available (for realistic, complex EAs)
- Can be hard to get balance between exploration and exploitation right
- Can have quite a few control parameters, hard to set them correctly
- Customization necessary for complex problems
- Not perfectly suited for expensive problems like HPO, as quite a higher number of evaluations is usually needed for appropriate convergence / progress