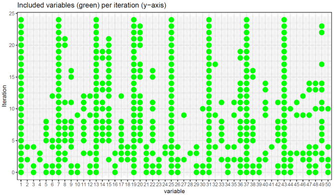
Optimization in Machine Learning

Evolutionary Algorithms - GA / Bit Strings



Learning goals

- Recombination
- Mutation
- A few simple examples

BINARY ENCODING

In theory, all problems can be encoded binary (binary system / binary code), but not always best representation (e.g., if values are numeric or trees or programs).

We typically encode problems with **binary decision variables** in binary representation. Examples:

- Scheduling problems
- Integer / binary linear programming
- Feature selection
- ...

RECOMBINATION FOR BIT STRINGS

Two individuals $\mathbf{x}, \tilde{\mathbf{x}} \in \{0, 1\}^d$ encoded as bit strings can be recombined as follows:

- **1-point crossover**: select crossover $k \in \{1, ..., d-1\}$ randomly and the first k bits from 1st and the last $d-k$ bits from 2nd parent.

$$
\begin{array}{cccc}
1 & 1 & & 1 \\
0 & 0 & & 0 \\
\hline
0 & 1 & \Rightarrow & 1 \\
1 & 1 & & 1 \\
1 & 0 & & 0
\end{array}
$$

- **Uniform crossover**: select bit j with probability p of 1st parent and $1-p$ of 2nd parent.

$$
\begin{array}{cccc}
1 & 0 & & 1 \\
0 & 0 & & 0 \\
0 & 1 & \Rightarrow & 1 \\
0 & 1 & & 1 \\
1 & 0 & & 1
\end{array}
$$

MUTATION FOR BIT STRINGS

An individual $\boldsymbol{x} \in \{0, 1\}^d$ encoded as a bit string can be mutated as follows:

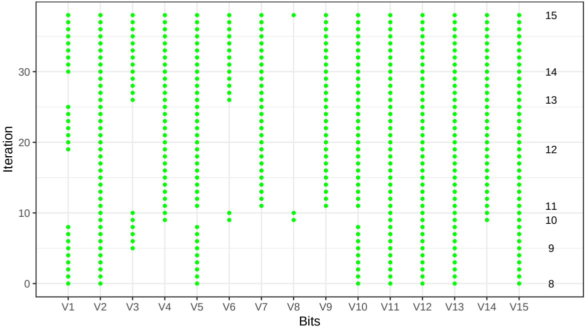
- **Bitflip**: for each index $j \in \{1, ..., d\}$: bit j is flipped with probability $p \in (0, 1)$.

$$\begin{array}{ccc} 1 & & \color{red}{0} \\ 0 & & 0 \\ 0 & \Rightarrow & 0 \\ 0 & & \color{red}{1} \\ 1 & & 1 \end{array}$$

EXAMPLE 1: ONE-MAX EXAMPLE

$\mathbf{x} \in \{0, 1\}^d$, $d = 15$ bit vector representation. Goal: Find the vector with the maximum number of 1's.

- Fitness: $f(\mathbf{x}) = \sum_{i=1}^{d} \mathbf{x}_i$
- $\mu = 15$, $\lambda = 5$, $(\mu + \lambda)$-strategy, bitflip mutation, no recombination



Green: Representation of best individual per iteration. Right scale shows fitness value of individual.

EXAMPLE 2: FEATURE SELECTION

We consider the following simulation setting:

- First, we generate a $(n \times p)$ design matrix $\mathbf{X}$ by drawing $n = 1000$ samples of $p = 50$ independent normally distributed features with $\mu_j = 0$ and σ_j varying between 1 and 5 for $j = 1, \ldots, p$.
- Then, we assume the following linear regression problem with the target variable $\mathbf{y}$ being generated as follows:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \epsilon$$

with $\epsilon \sim \mathcal{N}(0, 1)$
and $\boldsymbol{\theta}$ being defined as follows

$$\begin{aligned} \theta_0 &= -1.2 \\ \theta j &= 1 \qquad \text{for } j \in \{1, 7, 13, 19, 25, 31, 37, 43\} \\ \theta j &= 0 \qquad \text{else} \end{aligned}$$

Hence, there are 8 out of 50 equally influential features.

EXAMPLE 2: FEATURE SELECTION

Aim: Use a $(\mu + \lambda)$ selection strategy for feature selection.

Our iterative algorithm with 100 iterations is as follows:

1. Initialize the population and evaluate it. Therefore, encode a chromosome of an individual as a bit string of length p, i.e. $\mathbf{z} \in \{0, 1\}^p$. Where $z_j = 1$ means that variable j is included in the model.
2. Apply the variation and evaluate the fitness function. As fitness function, select BIC of the model belonging to the corresponding variable configuration $\mathbf{z} \in \{0, 1\}^p$.
3. Finally, use $(\mu + \lambda)$-selection strategy as the survival selection with population size of $\mu = 100$ and $\lambda = 50$ offspring.

In addition:

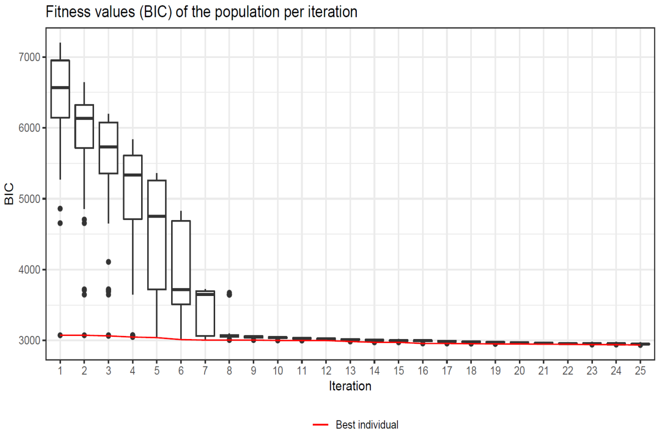
- for the mutation, use bit flip with $p = 0.3$
- for the recombination, use Uniform crossover with $p = 0.5$

EXAMPLE 2: FEATURE SELECTION

By exploiting **Greedy** as a selection strategy, ensure that you always choose individuals with the best fitness.

```
## [1] "After 10 iterations:"
##  [1]  1  7 11 13 14 15 19 20 22 25 30 31 36 37 40 43 44 48
## [19] 49 50
## [1] "After 20 iterations:"
##  [1]  1  7  8 13 15 19 20 25 31 37 43
## [1] "Included variables after 24 iterations:"
## [1]  1  7 13 19 25 31 37 43
```

EXAMPLE 2: FEATURE SELECTION



EXAMPLE 2: FEATURE SELECTION