

Stochastic optimization algorithms

Lecture 5, 20180912

Evolutionary algorithms:
Components of EAs

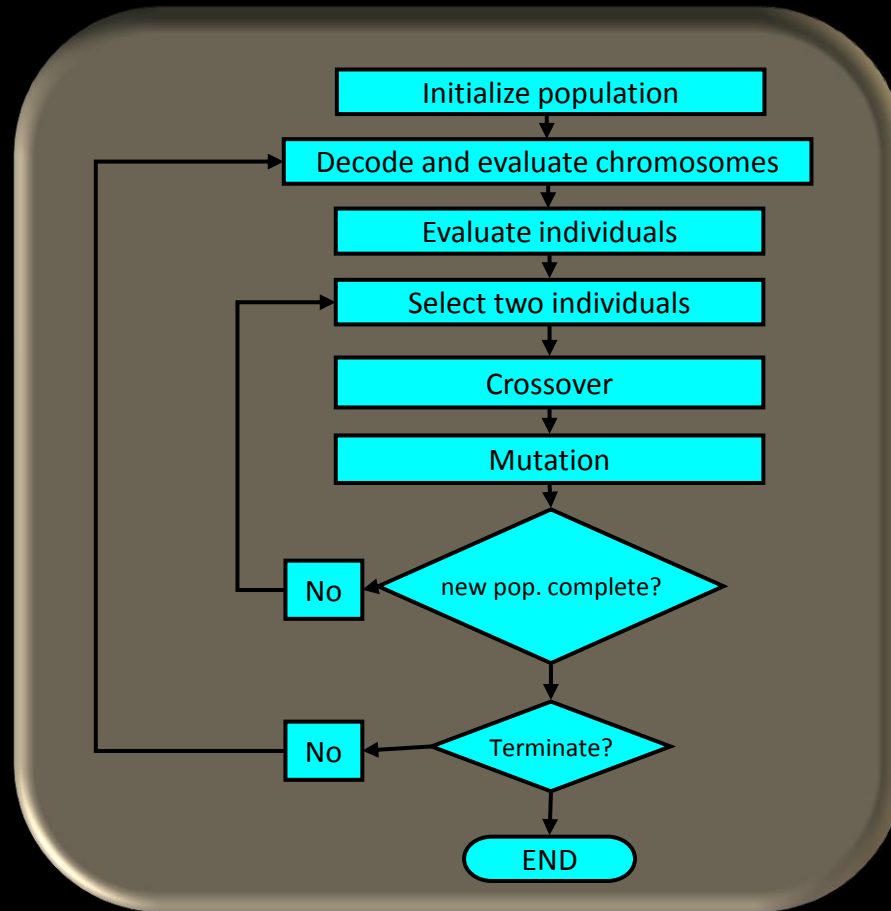
Information

- You do not need to hand in any files from yesterday's Matlab exercise.
- You must, however, hand in solutions to the following:
 - Introductory programming problem (deadline: 20180914, this Friday!)
 - Home problem 1 (deadline: 20180925)
- All relevant documents (including instructions – read carefully!) are available on the course web page.
- Note: Check the FAQ (bottom of the course web page) frequently.

Today's learning goals

- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm

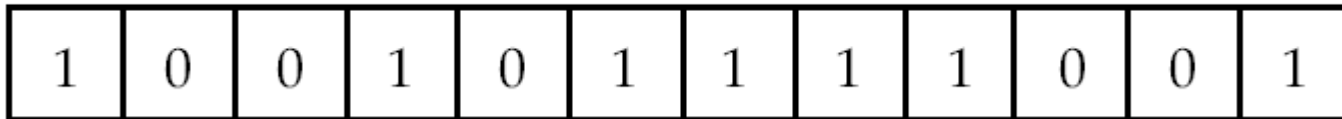
From last time...



pp. 46-48

Encoding schemes

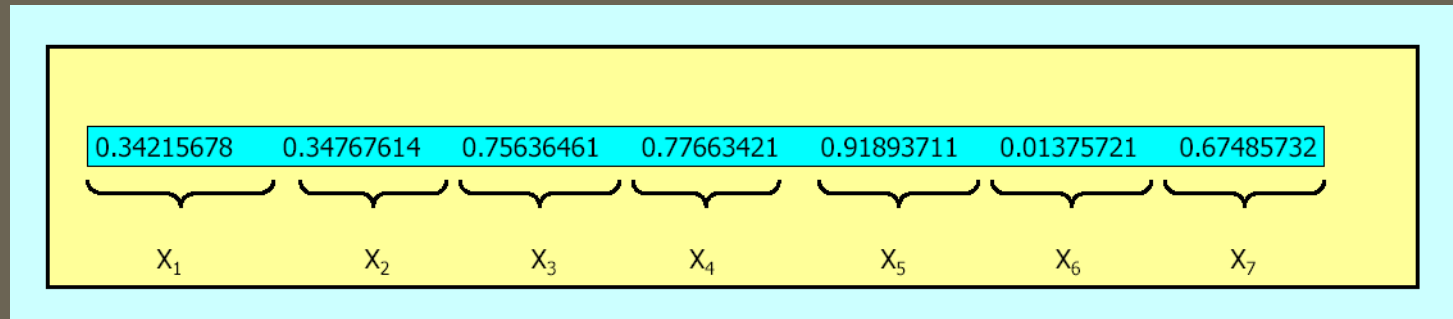
- Binary encoding



$$x = -d + \frac{2d}{1 - 2^{-k}} (2^{-1}g_1 + \dots + 2^{-k}g_k),$$

Encoding schemes

- Real-number encoding



$$x = -d + 2dg$$

Encoding schemes

- Other encoding schemes exist ...
 - Gray coding:
 - Small change in genotype => small change in phenotype
 - Standard binary coding : (00) \Leftrightarrow 0, (01) \Leftrightarrow 1, (10) \Leftrightarrow 2, (11) \Leftrightarrow 3
 - Gray coding: (00) \Leftrightarrow 0, (01) \Leftrightarrow 1, (11) \Leftrightarrow 2, (10) \Leftrightarrow 3
 - Messy encoding: Less position-dependent
 - Encodes both position and allele
 - Problem: avoiding missing genes and duplicates

Today's learning goals

- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm

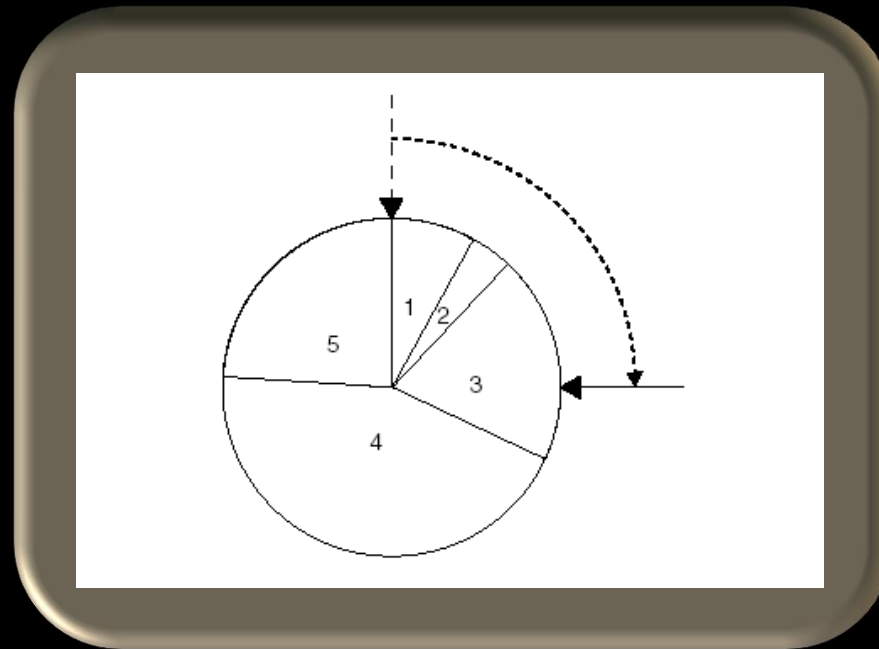


Selection

- Two main methods
 - Roulette-wheel selection
 - Tournament selection

Roulette-wheel selection

- Equivalent to spinning a roulette-wheel, on which the size of the sector occupied by each individual is proportional to its fitness:



Roulette-wheel selection

- In equation form: Select the individual with the smallest j that satisfies

$$\phi_j \equiv \frac{\sum_{i=1}^j F_i}{\sum_{i=1}^N F_i} > r,$$

...where r is a random number in $[0,1[$.

Roulette-wheel selection

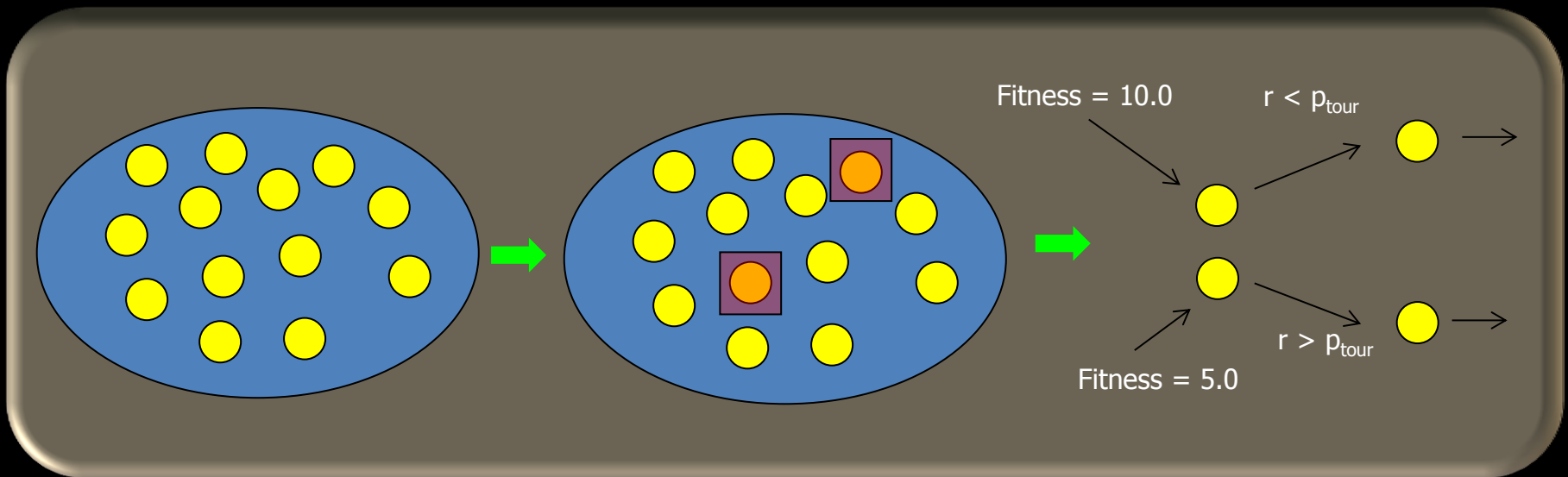
- Example 3.3 (p. 49)
- Fitness sum = 25.0 in this example.
 - Individual 1: $F_1 = 2.0 \Rightarrow \phi_1 = \frac{2}{25} = 0.08$
 - Individual 2: $F_2 = 1.0 \Rightarrow \phi_2 = \frac{1+2}{25} = 0.12$ etc.
 - Individual 3: $F_3 = 5.0 \Rightarrow \phi_3 = \frac{1+2+5}{25} = 0.32$ etc.
 - Individual 4: $F_4 = 11.0$
 - Individual 5: $F_5 = 6.0$
- If $r = 0.25$ (say), then $j = 3$ (smallest j such that $\phi_j > r$) etc.

pp. 48-52



Tournament selection

- Procedure for selection one individual:
 1. Pick two individuals randomly from the population (equal probability for all individuals in the population)
 2. With probability p_{tour} (usually around 0.7-0.8) pick the better of the two individuals (higher fitness). Otherwise, pick the worse of the two.



Tournament selection

- Can be generalized to larger tournaments (see p. 50).
- Note that the procedures described on the previous slides result in *one* individual being selected.
- Thus, selection must be carried out twice to select a pair of individuals.
- Note also that selected individuals are returned to the population after selection. Thus, the same individual can be selected many times, in both roulette-wheel and tournament selection.

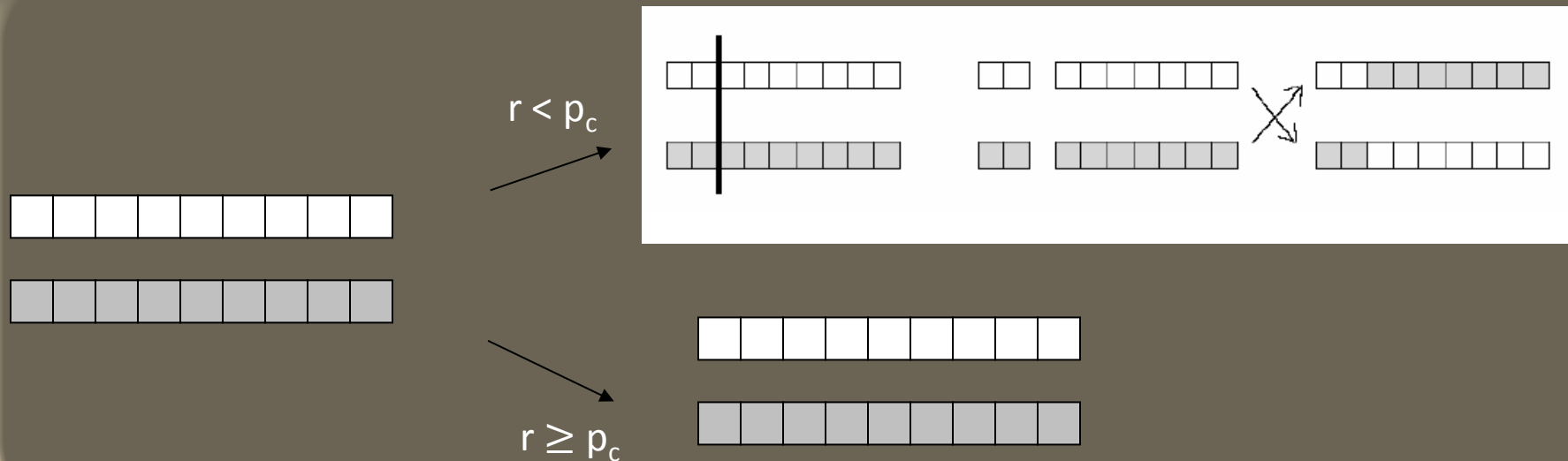
Today's learning goals

- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm



Crossover

- Carried out with a probability p_c . Otherwise: No change in the selected individuals (in this step).



Today's learning goals

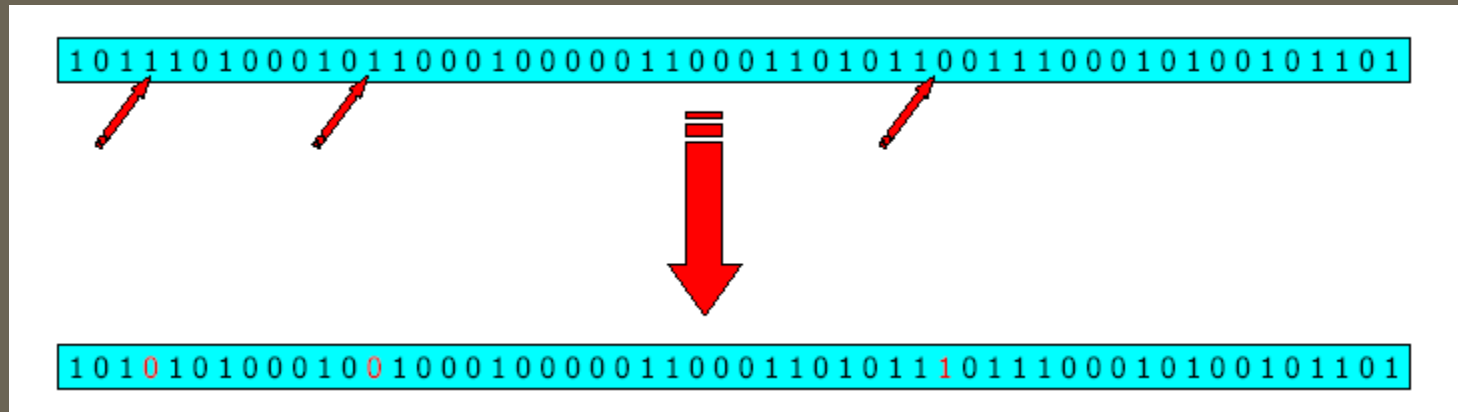
- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm



Mutation

- Small random changes, provide new material for the evolutionary process.
- Mutation rate (per gene): p_{mut} .
- Typical value: $p_{\text{mut}} = \frac{1}{m}$, (m = number of genes).

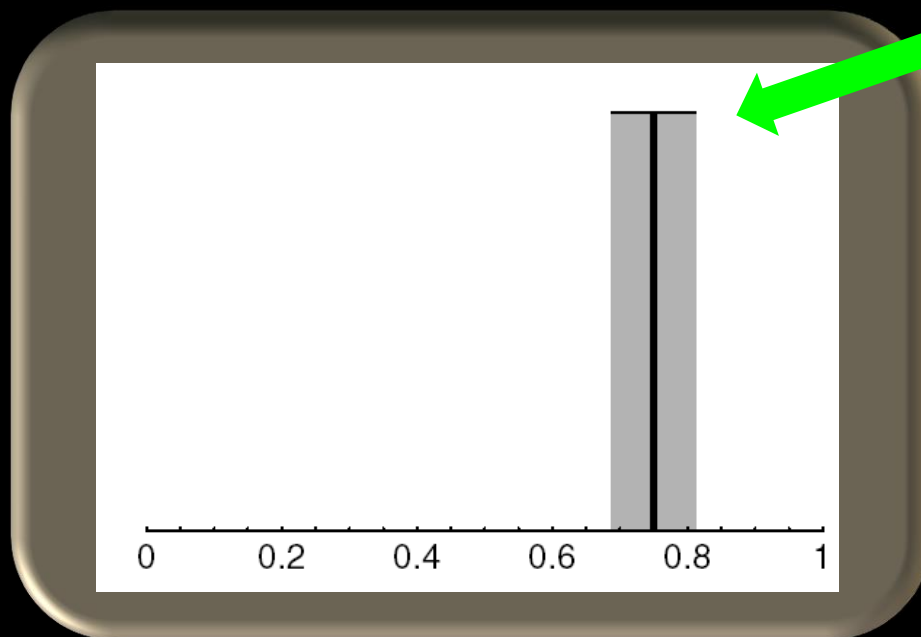
Mutation



- Every gene is checked. If, for a given gene, $r < p_{\text{mut}}$, the gene mutates ($r \in [0,1[$).

Creep mutations

- Used in connection with real-number encoding.
- The new values are drawn randomly from a distribution centered on the old value:



Uniform
creep mutation

Creep mutations: Procedure

- For each gene, check (as usual) whether or not the gene should be mutated (using the mutation rate)
 - If the gene should be mutated, check whether or not creep mutations should be used (using p_{creep})
 - If yes, carry out a creep mutation:

$$g \Rightarrow g' = g - C_r/2 + C_r r$$
 - If no, carry out an ordinary mutation:

$$g \Rightarrow g' = r$$
 - ...where, in both cases, r is another random number in $[0,1[)$.
 - If the gene should not be mutated, do nothing (proceed to the next gene).

Today's learning goals

- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm



Replacement

- **Generational replacement:**
 N new individuals are formed, and they replace the N old individuals (used e.g. in HP1.3).
- **Steady-state replacement:**
New individuals are formed in pairs, and replace the two worst individuals in the population.

Elitism

- A few exact copies (usually one) of the best individual in generation $g - 1$ is copied unchanged to generation g .
- This procedure ensures that
 - ... the maximum fitness will rise monotonously (or at least never fall),
 - ... the best individual is always available when the algorithm is terminated.

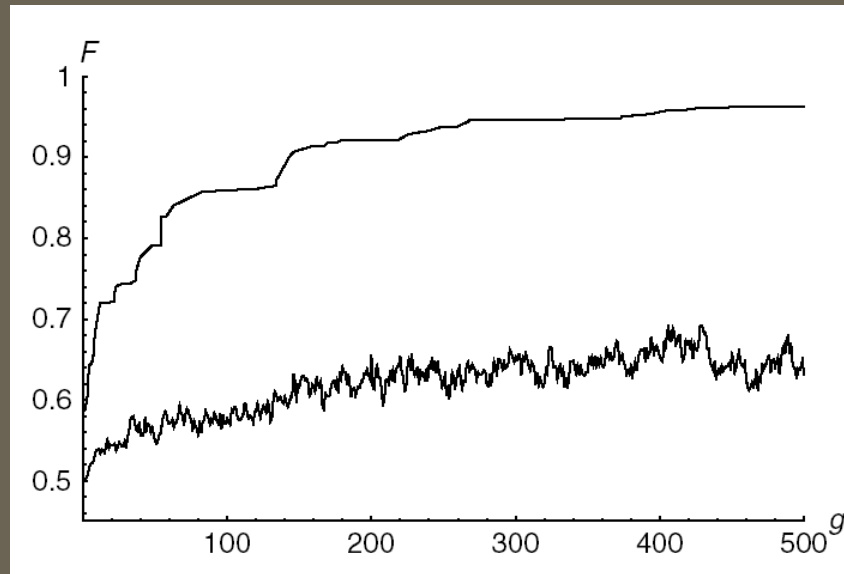
Today's learning goals

- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm



Typical fitness curve

- Top curve: Maximum fitness, Bottom curve: average fitness



Standard GA

- Algorithm 3.2 in the book.
- Study carefully! (Note, however, that elitism can be done more easily by simply copying the best individual into position 1 (in the population) at the end of each generation; see also the Matlab introduction).
- The selection step is usually carried out with tournament selection.

Today's learning goals

- After this lecture you should be able to
 - Describe and compare some different encoding schemes
 - Describe and compare two different selection methods
 - Describe the crossover operator
 - Describe the process of mutation
 - Describe the processes of replacement and elitism
 - Define a standard genetic algorithm



NOTE!

- When solving HP1.3, you can, *to some extent*, make use of the code from the Matlab introduction, but you need to adapt (generalize) the code a bit. Read the instructions carefully!
- We aim to return the results from the introductory programming problem (IPP) on Wednesday (20180919): The results will be handed out (on paper) at the lecture.
- Make sure to study your results for the IPP before handing in HP1!

