Machine Learning Worksheet 9

Evolution

1 Biological Evolution

In evolutionary algorithms the terms used in biology and the principles observed in nature are taken over to optimization algorithms.

Problem 1. Explain the biological Meaning of the terms population, individual, selection, mutation, recombination, reproduction and fitness. Give examples on how these principles are realized in evolutionary algorithms.

2 Local Search - Hill Climbing

Local search is the category of evolutionary algorithms that uses only one individual for transmission from generation to generation. The simplest local search algorithm is the Hill Climber. Here we have one individual that reproduces - if the new individual has a higher fitness the individual is taken over in the new generation if not the old individual is kept. This algorithm is very fast and simple, but it can very easily get stuck in local optima or end up in a suboptimal part of the parameter space.

Problem 2. The fitness function $f(x) = (x - 23)^2$ is given as a minimization problem. X is a 6 bit integer, X is initialized to 000000. The bit flip (mutation) probability is p=0.5. Use a coin to decide if a bit is flipped or not. Write down the value, bit representation and fitness of X in every generation in a table for 15 generations. Do the same in gray code representation. Explain the properties of the different representations and explain the term *Hamming Wall*.

Problem 3. Assume in the above example that all but one of the bits are correct. What is the probability that the next individual is the optimal one while keeping p=0.5? What bit flip probability would be the optimal one?

Problem 4. In the last exercise the optimal mutation probability is correlated to the number of bits the genome is represented by. Show that the optimal mutation probability is always the reciprocal of the number of bits representing the genome, if only one bit is wrong. (We will see in the Genetic Algorithms section that this mathematical relationship is important)

3 Genetic Algorithms

The main cycle of evolution as defined by Darwin 1859 in *On the Origin of Species*, includes selection, mutation and reproduction. For a long time the power of recombination was underestimated. In Genetic Algorithms the focus is on recombination. Mutation is only a secondary operator. How recombination works is best explained with a binary genome and the schema theorem.

Schema: A schema is a template made up of a string of 1s, 0s, and *s, where * is used as a wild card that can be either 1 or 0. For example H = 1**0*0 is a schema; it has eight instances, one of which is 101010. The number of non-*, or *defined* bits in a schema is called its *order* and denoted o(H). In the example, H has order 3. The greatest distance between two defined bits is the defining length $\delta(H)$. In the example, H has a defining length of 3. In the discussion that follows we use the term *schema* to refer both to the template and to the set of instances it defines within a population. The fitness of a schema is the average fitness of all strings matching the schema.

Schema Theorem (Holland): Short, low-order, schemata with above-average fitness increase exponentially in successive generations.

$$m(H, t+1) \geq \frac{m(H,t)f(H)}{a_t}[1-p]$$

Here m(H,t) is the number of strings belonging to schema H at generation t. f(H) is the Fitness of H and a_t is the average fitness of generation t. p is the probability that crossover or mutation destroys the schema. It is obvious that schemata with a low order and a short length have a very low probability of being destroyed. Assume p is very close to zero. It is clear that for schemata that have a higher fitness than average the amount of instances of that schema is growing.

Problem 5. Given is a population of binary coded individuals. Also given is the situation that for every position in the genome there is at least one 0 and one 1 available in the population. Therefore every binary string can be created by recombination alone. Explain why it is still crucial to have mutation included in the algorithm. Remembering the results of exercise 4, why is the suggested mutation probability in Genetic Algorithms 1/(number of bits representing the genome)?

Problem 6. Given is a minimization problem with the fitness function $f(X) = (X-5)^2$ for x ϵ 0,..,15. Which schema H of all schemata with o(H)=2 and $\delta(H) = 1$ has the best (smallest) fitness?

Determine the fitness of all schemata with o(H)=2 that fit the solution x=5 for binary and gray code representation and compare them to the overall average fitness. Describe your observations.

4 Evolution Strategies

Evolution Strategies are guided by mutation. Under the assumption that similar parametric solutions have similar fitness (which is not always the case) and in particular that solutions nearer to a local/global optima have in most cases a higher fitness, optimization can be achieved by mutation of the current individuals.

As far as real-valued search spaces are concerned, mutation is normally performed by adding a normally distributed random value to each vector component. The step size or mutation strength (i.e. the standard deviation of the normal distribution) should decrease over time as the population approaches the optimum. In Evolution Strategies this is governed by self-adaptation. Self-adaptation is done by making the mutation strength a part of the genome itself. Because individuals with the right mutation strength (strategy parameter) are more likely to produce offspring with higher fitness, the amount of individuals with suitable strategy parameters will grow. As a result the population will adapt faster/better.

Problem 7. Given is a minimizing problem with the fitness function $f(X) = (X-5)^2$ for x $\epsilon\mathbb{R}$. $X_0 = 8$, $\sigma_0 = 1.0$. From this starting point you may calculate a very simplified version of an Evolution Strategy:

- Mutate step size σ - chose a σ with $\sigma_{t+1} = \sigma_t * 2$ or $/2$ you assume to be suitable
- Mutate parameter X - make a step in a direction you chose with σ_{t+1}
- Evaluate - calculate the Fitness.
- Repeat

Record Generation, σ_{t+1}, X and f(X) for 10 generations in a table like this:

Generation	σ	X	f(X)
0	1.0	12.0	64.0
1	2.0	10.0	25.0
2	4.0	6.0	1.0

Bonus: Give a reason for your choices of σ. Would that σs also be a good choice if the mutation were normal distributed as usual?