CT5141 Lab Week 7

James McDermott

Feature Selection with LAHC and GA

Feature selection is the problem, in machine learning, of choosing which features we should use, out of a set of n features.

1. What is the search space? What is the objective function? (No need to write code here.)
2. Get `feature_selection.py` from Blackboard. It contains an implementation of the objective for an example problem, the Boston Housing regression problem. Notice that we use `train_test_split(X, y, random_state=0)` so that we get the same train-test split every time. (This doesn't affect the randomness elsewhere in the algorithm.) There are 13 features in this problem. How large is the search space? Could we use enumerative search?
3. Run both LAHC and GA on this problem. Check whether your code is maximising or minimising! Use a fixed evaluation budget of 400 (this is tiny, for illustration only). For LAHC, investigate the `L` hyperparameter. For GA, investigate the `popsize` hyperparameter (bear in mind that `popsize * ngens = maxits`).
4. Observe that in LAHC, the objective never disimproves. But we know that LAHC allows disimproving moves. Explain this by looking at the LAHC code.

CMA

Covariance Matrix Adaptation (CMA) uses a multivariate normal to model the locations of good individuals in the population, then draws from that distribution to make new individuals, then discards bad ones to leave only good ones again. It also does some more clever stuff that we will not study. We are going to implement a simple version of CMA!

5. See the code provided in `eda_start.py`. Test out the individual functions provided by running them in IPython or Jupyter Notebook with some small inputs.
6. Use those function to build a complete CMA-like algorithm and test it on a `sphere` problem and `rosenbrock`.