

**CS 410/510  
Machine Learning  
Winter, 2006**

**Homework 7:  
Genetic Algorithms**

Due Thursday, March 9.

In this homework assignment, you will modify the “Simple GA in C” code to do some experiments on feature selection with the GA, as described in class (and below).

You can download the Simple GA in C from the class web page:

<http://www.cs.pdx.edu/~mm/MachineLearning2006/simple-ga-in-c.tar.gz>

Un-gzip the tar file, extract the files, and follow the instructions in README.pdf to run the program.

You should run this code on the Linux network to do the experiments described below.

If you have any questions about this code (or if you are not familiar with C), feel free to e-mail me or come see me.

## **Genetic algorithm for feature selection**

You will be writing a fitness function for the genetic algorithm to perform feature selection for a naive Bayes classifier. The individuals are bit strings of length 57. Each position in the string is an off/on switch for the corresponding attribute in the UCI spam database. Suppose you have a training set `spam.train`, a validation set `spam.val`, and a test set `spam.test`. The fitness of individual  $c$  is calculated as follows:

1. Train a naive Bayes classifier using `spam.train`, but only using the attributes whose corresponding value in  $c$  is 1.
2. Run the resulting naive Bayes classifier on `spam.val`. The fitness of  $c$  is the accuracy of the naive Bayes classifier on the validation set (i.e., a fraction between 0 and 1).

## **What you need to do**

1. Starting with the Simple GA in C code, modify the `fitness.c` file to implement the fitness function described above. You can adapt your own naive Bayes code if you wish, or adapt the C version provided on the class website.

2. Modify the “params” file to use the following parameters:

```
NUM_RUNS 10
FITNESS_FUNCTION_NAME "feature selection"
MAX_NUM_FUNCTION_EVALS 1000          # Maximum of 50 generations
POP_SIZE 20
CHROM_LENGTH 57
ELITISM TRUE
FITNESS_FUNCTION_OPTIMUM_KNOWN TRUE
FITNESS_FUNCTION_OPTIMUM 1.0
RUN_NUM_DIR "<fill in pathname here>"
OUTPUT_DIR  "<fill in pathname here>"
LONG_PRINT TRUE
```

All other parameters can stay the same as in the original params file.

3. Using the UCI spam data, create a training set, validation set and test set. The validation set and test set should be smaller than the training set (e.g., 800 examples in each). Make sure each data set contains about 40% spam and 60% ham.

4. Run the GA (it will do 10 independent runs, as indicated in the params file). For each run, record the best fitness found in the run, and the individual that produced that fitness. (If there is more than one best individual, chose one of them arbitrarily to record.)

5. For each of your 10 best individuals (from the 10 different runs), run the corresponding naive Bayes classifier on the test set, spam.test. Record the number of attributes used and the classification accuracy of each on the test set. Compare with the classification accuracy on the test set of the naive Bayes classifier using all of the 57 attributes.

Give the recorded information from steps 4 and 5 in table form in your writeup, and write a one-paragraph discussion.

6. Which GA parameters do you think could be modified to improve your results? Design and run an experiment (10 runs of the GA) with these modified parameters. Describe your experiment, why you thought modification of those parameters might help, and the results.

7. Do one run of your GA using

```
SELECTION_METHOD "fitness proportionate"
```

in the params file. Once your run is done, choose one generation  $G$  and choose a schema  $s$ . Using information given in the .long file, record the number of instances of  $s$  at generation  $G$  and at generation  $G + 1$ . along with their fitnesses. Now use the Schema Theorem to compute the predicted change in frequency of  $s$  over one generation. In your writeup, give the schema you chose, show your work in the calculation, and describe how the predicted result compares with the observed result.

Here is what you need to turn in: **Hard copy:** Hand in a hard-copy of your (computer formatted, not handwritten) writeup as described above. **Electronic:** E-mail me your (well-commented) fitness.c file.