On GP, data cardinality and the computational cost of fitness evaluation

CSCI6506 Sandbox 2

Assignment due date: 29th January, 2015

Task definition

- Your goal is to extend your canonical GP for multi-class classification from sandbox 1 such that it now addresses the issue of training exemplar cardinality.
- Consider the case of the two data sets of Table 1. These consist of between 3,772 and 43,500 exemplars in the training partition, T. Even assuming a relatively small population of 500 and 50 generations, there would have to be between $\approx 94 \times 10^6$ and $\approx 1 \times 10^9$ evaluations *per run*.
  - Note that you might want to revisit the range over which attributes are normalized. Specifically, the unit interval is *not* recommended for GP. Make sure than any normalization is designed relative to the training partition and then reapplied to the test set.
- Clearly we need to be able to decouple the number of exemplars used within any single fitness evaluation. The most direct way of achieving this is to introduce the concept of a *data subset* that defines a 'subset' of exemplars sampled from the larger training partition under a sampling heuristic (Figure 1).
- Let the number of training instances within the data subset be τ where $\tau << T$, say $\tau = 200$.

[1] `http://archive.ics.uci.edu/ml/datasets/Thyroid+Disease`
[2] `http://archive.ics.uci.edu/ml/datasets/Statlog+\%28Shuttle\%29`

Table 1: Source data for sandbox 2. See footnotes for URL.

Name	Num. Training	Num. Test	Num. Classes
Thyroid[1]	3 772	3 428	3
Statlog "Shuttle"[2]	43 500	14 500	7

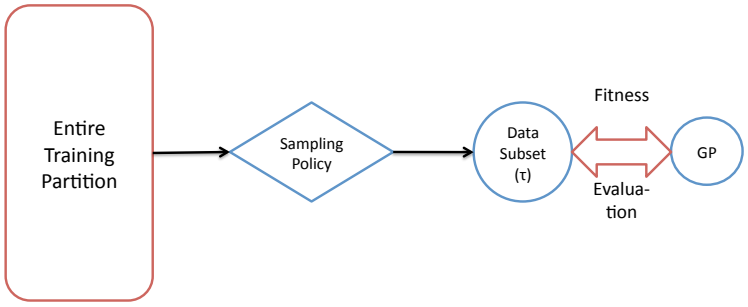

Figure 1: Relation between GP population, original training partition and the data subset. Either all or some fraction of the Data Subset will need replacing periodically. The easiest starting point is to replace all τ exemplars at each GP generation.

- You will consider two simple sampling heuristics:

  1. At each generation sample τ exemplars to appear within the data subset from the original training partition with uniform probability (without replacement[3]).
  2. At each generation sample $\tau/|C|$ exemplars from the original training partition with uniform probability (without replacement); where $|C|$ denotes the number of classes.

- Post training, performance will be evaluated from the perspective of two performance metrics

  1. Accuracy (ACC):

$$ACC = \frac{tp + tn}{tp + tn + fp + fn} \tag{1}$$

     where tp and tn are the counts of true positive and true negatives respectively, and fp and fn are the counts of the corresponding falsely labelled exemplars. For more information see the definition for 'confusion matrix' from wikipedia.[4]

  2. Class-wise detection rate (DR):

$$DR = \frac{1}{|C|} \sum_{i=1...|C|} DR(i); \text{ where } DR(i) = \frac{tp(i)}{tp(i) + fn(i)} \tag{2}$$

     i.e. the $DR(i)$ is estimated independently w.r.t. each class, i and then normalized by the number of classes $|C|$. In effect you are estimating the detection rate of each column of the confusion matrix.

- You will need to consider what metric to use for fitness evaluation *during* training (it need not be one of the above, but it could be).

[3] i.e. the exemplars within the data subset are unique

[4] `http://en.wikipedia.org/wiki/Confusion_matrix`

Reporting

- Write a short 2 page summary answering the following questions:
  - the approach you adopted for designing the fitness function;
  - identify your best classifier, post training, using the *training partition* and *each* of the performance metrics (each run will potentially identify 2 different champion classifiers). Now report the percentage of each class this classifier identifies under the *test partition*.
  - how much sensitivity is there to the value assumed for τ?
  - what other issues are you implicitly addressing aside from the dataset cardinality when you design a sampling policy?
- Your report should be emailed to `mheywood@cs.dal.ca` before midnight on the sandbox due date.