Genetic Algorithms for Feature Selection in Data Mining

Magnus Erik Hvass Pedersen (971055)
Daimi, University of Aarhus, November 2003

1 Introduction

The purpose of this document is to verify attendance of the author to the *Data Mining* course at DAIMI, University of Aarhus.

The document is an addendum to [1] of which most terminology is reused, so that *data mining* designates the step building the predictive model, and *knowledge discovery* includes not only this step, but also any pre- and post-processing, data gathering, output simplification, etc.

It was briefly noted in [1] how a perspective on a data-set is necessary to avoid the discovery of incorrect rules. As an example, the relationship between names and ages of people was described, in which names are discovered for the *antecedent* of a rule, where the age should have been used instead.

It was then speculated that a good enough model, accompanied by enough data in terms of both number of samples and attributes for each sample, would be able to predict when such things as names would become fashionable again, and how this would influence already discovered rules.

2 Feature Selection

Even though the development of thoughts and trends seems largely predictable, and such a model is therefore plausible, it was of course a rather philosophical discussion. For solving real-world problems with the models available, a perspective on the data-set *is* needed, not just to aid the search for valid[1] rules, but also to make the search tractable at all. Furthermore this naturally leads to simpler models, and hence better generalizations.

A perspective is better known as *feature selection*, in that features or attributes of the samples are selected if they seem important for the kind of rules that are to be discovered. It can be considered as a means to avoid feature-induced *overfitting*.

For instance, bank customer records may contain name, date of birth, occupation, marital status, last year's income, and account balance. If the knowledge discovery is supervised, the supervising person may prior to the data mining

[1] In the sense of longevity of their applicability.

task, select which features are relevant, e.g. the name and marital status could be excluded manually.

If unsupervised learning is desired though, we must devise a scheme that finds the feature subset that optimizes the quality of the actual data mining output. A number of methods are commonly used [2], depending on the complexity of the data mining task:

- **Embedded** feature selection is a part of the data mining process, in that features are added or removed while building the model, depending on the change in accuracy. It is not described further in this document.
- The **wrapper** approach uses data mining as a sub-routine, evaluating the quality of the feature subset by the accuracy of the model discovered in the data mining sub-process. For example, in classification as in [1], the function $Fitness(c)$ can be used directly as the measure that must be optimized by changing the composition of the feature subset.
- **Filtering** is a non-recurring preprocessing step used when the data mining task is too expensive for iterative use as in the wrapper method, and a low-cost approximation is unavailable or unviable. Instead a measure independent of the ensuing data mining task is used, e.g. based on the theory of information or statistics.

In [3] a filter approach for classification is described that generates a number of random feature subsets, evaluating the quality of each by a measure of *consistency* - essentially two samples are inconsistent if they match except for their target attribute or classification.

Although the wrapper method seems expensive at first, corners can sometimes be cut. For instance, when using a genetic algorithm (GA) for the discovery of rules, the features that are removed in successive steps of selection, may simply also be removed from the chromosomes representing rules. That way the optimal solutions found for one feature subset, are reused in the next step, hopefully shortening the time needed to compute optimal solutions for the new feature subset. Diversity and exploration may be favoured by introducing higher mutation rates early in each run, and the termination criterion can then be a moderate stagnation estimate, because the solutions are continuously refined anyway, only over slightly different feature subsets.

Instead of discretely including a feature or not, its presence may be gradually altered, leading to the concept of *feature weighting*, which lends itself to an embedded implementation.

3 Example Selection

Just as features may *confuse* the data mining algorithm, so may some of the *examples* or samples of the data-set. It is interesting to note, that the more accurate the model of the data-set becomes, the more it will correctly describe

the same examples over and over again. Again, if a decent database implementation is used, the database may automatically *cache* these sub-results for quick retrieval in successive queries.

In general, the placement of this *example selection* process may be similar to that of feature selection: Embedded, wrapper or filter. Simply ignoring traning examples already consistent with the model, is naturally suited for embedded implementation. Note that this form may also be used in the training of neural networks.

Using the wrapper approach is again a safer bet than filtering, but computational complexity may still prohibit its use. A solution however, is to consider a small subset or *window* of examples [2], iteratively choosing new subsets to augment the existing and thus refining the model.

An idea for a filtering method in data mining of time series, is to condense the series so that a statistical measure replaces a subset of points. This measure could be the average, standard deviation, etc., allowing large series to be piecewise *summarized* into a new time series, allowing the data mining algorithm to expand on these measures as required.

4 GA For Feature Selection

Whether the filter or wrapper method is used for feature selection, they are both searching the space of available features and have a fitness function associated. So they are both optimization problems. Since a feature is either included or not, the search-space may naturally be binary coded so that a GA as outlined in [1], may be used for finding the optimal feature subset. This way the classification task of [1] may use two separate GAs - one for feature selection, and one for rule discovery.

References

[1] Genetic Algorithms for Rule Discovery in Data Mining
Magnus Pedersen (971055)
Daimi, University of Aarhus, October 2003
`http://www.daimi.au.dk/~u971055/`

[2] Selection of relevant features and examples in
Machine Learning
Avrim L. Blu, Pat Langley
Artificial Intelligence 97 (1997) pp. 245-271

[3] A Probabilistic Approach to Feature Selection
- A Filter Solution
Huan Liu, Rudy Setiono
National University of Singapore, Kent Ridge 119260
`http://citeseer.nj.nec.com/`