

# Feature Selection using a Genetic Algorithm applied on an Air Quality Forecasting Problem.

Elias Kalapanidas<sup>1</sup> and Nikos Avouris<sup>2</sup>

**Abstract.** Feature selection is a process followed in order to improve the generalization and the performance of several classification and/or regression algorithms. Feature selection processes are divided in two categories, the filter and the wrapper approach. The former is performed independently of the learning algorithm while the latter makes use of the algorithm in an iterative way.

This paper focuses on the exploitation of a genetic algorithm used to extract an optimal feature subset of a large database containing pollutant concentration measurements, following the wrapper approach. The feature subset feeds a nearest neighbor algorithm in order to predict the daily maximum concentration for two pollutants. The encoding problem of the complexity of representation of the features in the genomes is tackled. Results of the experimentation on an air quality forecasting problem will be presented, as well as slight alterations on the standard simple genetic algorithm paradigm that guided the algorithm to a mature convergence and gave good solutions.

## 1. INTRODUCTION

The main problem is to provide with adequate predictions for the daily maximum NO<sub>2</sub> and O<sub>3</sub> concentrations for the city of Athens. Since every calendar day is thus considered as a case, the gathered raw data that are based on an hour basis are transformed in datasets where every row represents a calendar day. It is known from the field experts that at the first stage of the morning a chemical reaction called photochemical is taking place between NO, NO<sub>2</sub> and O<sub>3</sub>. High concentrations of NO and an adequate presence of sunlight can lead to high production of NO<sub>2</sub> which is a source of danger for the human health. Consequently high morning NO<sub>2</sub> concentrations have serious impact on the O<sub>3</sub> dispersion around the suburbs of the city during the afternoon hours. Our aim, besides the accurate NO<sub>2</sub> and O<sub>3</sub> daily prediction, is to evaluate the theory of the experts behind the daily lifecycle of the two pollutants and to identify further attributes that contribute to this cycle.

Prediction algorithms have failed one too many times when presented with difficult real-time problems. A few steps that could lead to better efficiency of the learning algorithm are:

- the selection of optimized algorithm parameters,
- dealing with the learning overfitting,
- dealing with noise in the input dataset,
- applying post-processing (eg. exploiting ensembles of classifiers, or boosting the classifier), or
- selecting a suitable subset of significant features.

The problem of feature selection can be seen as a case of feature weighting, where the numerical weights for each of the features have been replaced by binary values. A value of 1 could mean the inclusion of the corresponding feature into the subset, while a value of 0 could mean its absence.

As [1] describe, the feature weighting algorithms are divided into two categories: the filtering methods and the wrapper methods. The former is a no-feedback, pre-selection approach where the selection of the feature subset is performed independently of the learning algorithm. The latter is an iterative method that encapsulates the learning algorithm in the feature selection process.

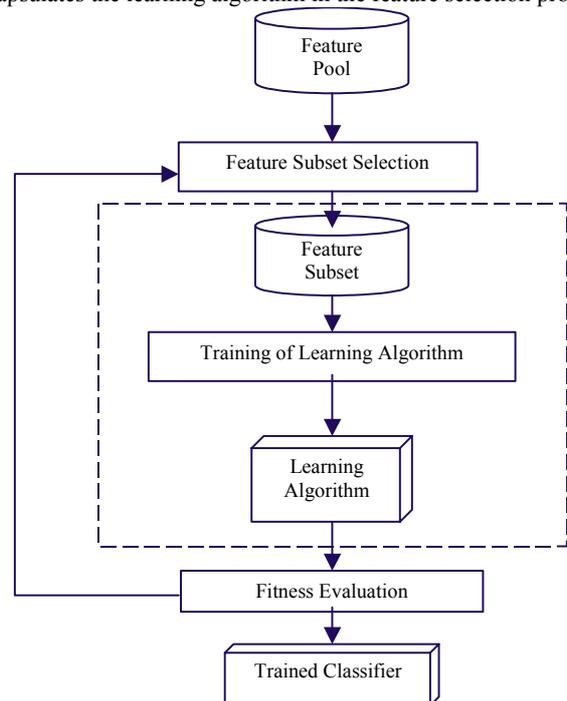


Figure 1. Flowchart of the wrapper approach

In this paper we investigate the applicability of feature selection driven by a genetic algorithm at the input of a nearest neighbor classifier. The wrapper approach followed is presented in figure 1. The problem under consideration is the prediction of the daily maximum concentration for two important pollutants (nitrogen dioxide and ozone) in the urban area of Athens. The database where the input datasets are coming from stores concentrations for five pollutants, for every hour of the day, for every of the eleven measurement stations that are dispersed in the Athens basin. Previous applications of a case-based reasoning approach, a neural network and a decision tree predictor in [2],[3], a bayesian classifier in [4] and a LVQ neural network in [5], have proved the efficiency of machine learning algorithms for various short term air quality forecasting problems.

## 2. RELATED WORK

Feature selection as an optimization process can be solved using a number of different search techniques. Exhaustive search has been used by [6] by exploiting a breadth-first algorithm in order to find a feature subset of minimal length consistent with the training set cases. Hillclimbing and best-first search have been studied by [7].

<sup>1</sup> Department of Electrical Engineers, University of Patras, email: ekalap@ee.upatras.gr

<sup>2</sup> Department of Electrical Engineers, University of Patras, email: n.avouris@ee.upatras.gr

Branch and bound search for feature selection of a pre-determined size has been reported by [8]. A filtering method for feature weighting called Relief using random sampling is introduced in [9] and further enhanced to handle noisy features in [10].

Genetic algorithms as tools in feature selection have been studied by several researchers, particularly in [11] and in [12]. Feature selection using genetic algorithm as a pre-processing step for neural network classification has been exploited in [13]. [14] empirically prove for a set of problem datasets that the genetically generated feature subsets that are fed to a neural network are more efficient than using the full feature sets. [15] compare the use of a filtering method (F-Relief) to a wrapper method of a genetic algorithm, showing that a C4.5 and a neural network classifiers are improved using the later.

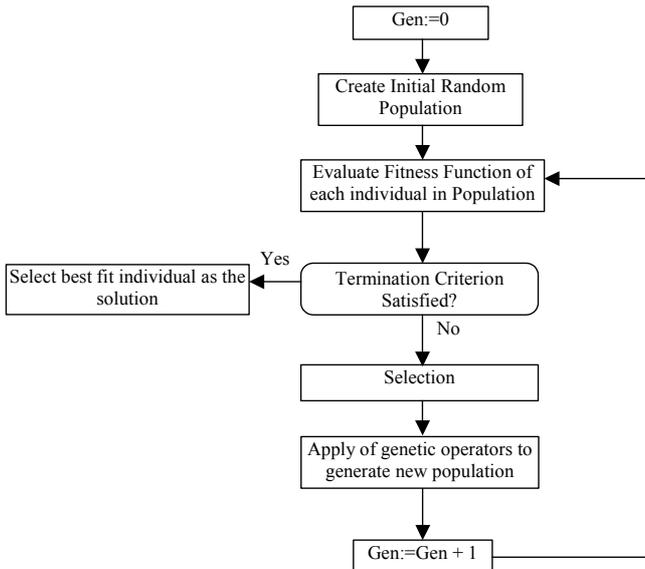


Figure 2. Flowchart for the applied genetic algorithm

### 3. IMPLEMENTATION DETAILS

The difficulty with the specific problem and consequently with every time-series and/or spatial problem is the large feature space.

Since we investigate the power of feature selection to provide with an effective subset of features, the input data are not passed through any transformation. It is examined whether the feature selection scheme has the power to improve the learning algorithm by itself, without any subsequent pre-processing.

Considering that every row in the datasets corresponds to a calendar day (let's say day-0), then for every day-0 in the data sets, two types of features are considered:

1.  $Q(P,S,H,O)$  which is a query of pollutant P concentration at measurement station S at hour H, O days before day-0.
2.  $Q(P,S,F,H1,H2,O)$  which is a query of an aggregate function F of pollutant P concentrations at station S between the time period  $[H1,H2]$  O days before day-0.

The valid range for the P,S,H,F,H1,H2 and O variables is as follows:

1. Pollutant P: one of the five measured pollutants {NO2, O3, CO, NO, SO2}
2. Station S: one of the nine measurement stations in Athens that gave consistent data, from the coded range [101,...,109]
3. Hour H: one in the range [100,...,2400]
4. Function F: one of {"Average", "Maximum", "Minimum", "Variance"}

5. Hour H1: one in the range [100,...,2300]
6. Hour H2: one in the range (H1,...,2400]
7. Day offset O: one in range [0,...,31]

The standard procedure during the encoding of the feature subset is to generate a bitstring containing a 0 (for disabled) /1 (for enabled) bit for every possible feature in the initial pool of features. This approach works well for problems having a reasonably reduced set of features [14]. In our problem the 7 variables P,S,H,F,H1,H2 and O define a feature space containing 1,723,680 possible features, corresponding to a considerable amount of memory for a computer system to store all genotypes for the whole population. Thus we need an alternative representation scheme: a free size array of bitstrings containing only the enabled features. The representation along with a sample crossover operation of two genotypes is presented in figure 3.

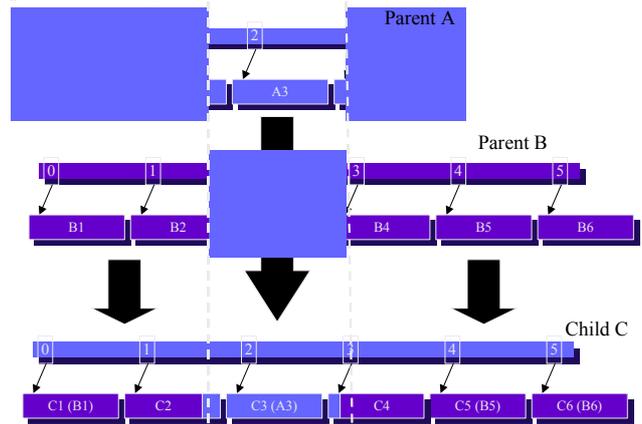


Figure 3

Two disadvantages arise from this approach:

1. The algorithm does not "know" the whole feature space, since the majority of it is absent from the genotype encoding.
2. The possibility of redundancy should be dealt with. Duplicate features may be showed up in any genotype during the application of the genetic operators.

The anticipation measures for each of these drawbacks include a big mutation rate and a feature duplication detection and elimination after each genetic operation cycle. Experiments with different mutation rates show that big rates favor the population expansion in the exploration of the feature space.

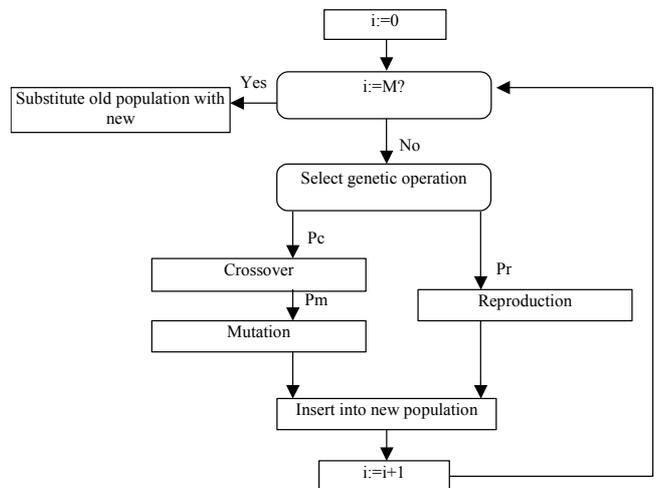


Figure 4. Flowchart for the applied genetic operations

The genetic algorithm that is used follows the steps of figure 2. For the implementation of the genetic algorithm, the GA library SimBioSys was used. This is a set of C++ classes where the basic GA functionality is available for exploitation. The GA operations are applied in a slightly different way from the simple genetic algorithm. In this implementation the possibility of reproduction is 1 for a defined number of the elite genotypes of the previous generation, while all of the remainder genotypes are passed through crossover and then through mutation with a variable mutation probability, as it is presented in figure 4.

The fitness function selected after preliminary experiments is composed of two parts: one corresponding to the minimization of the classification error and one corresponding to minimizing the length of the solution, which is the number of features in the genotype:

$$F(E(i), A(i)) = \frac{1}{a \cdot \frac{E(i)}{E \max|n} + b \cdot \frac{A(i)}{A \max|n}} \quad (1)$$

where

**F(i)** is the fitness function,

**n** is the number of phenotypes in the population,

**E(i)** is the weighted classification error WSE of the  $i_{th}$  classifier,

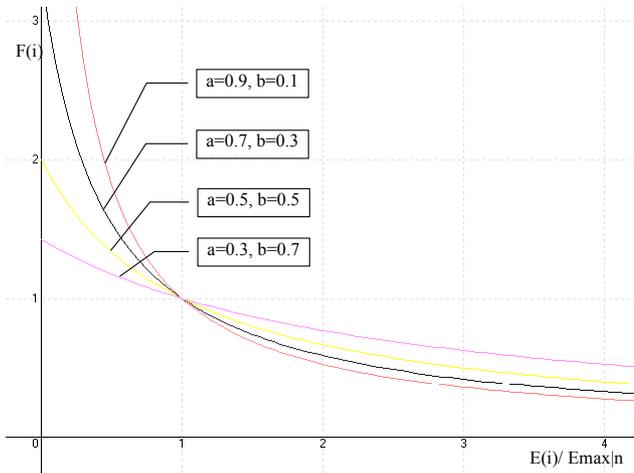
**E<sub>max|n</sub>** is the maximum WSE value in the population per generation,

**A(i)** is the number of features of the  $i_{th}$  classifier,

**A<sub>max|n</sub>** is the maximum number of features among the phenotype population per generation,

a and b are numerical values in the range [0,...,1], with the following three constraints:

$$a \leq 1, b \leq 1, a + b = 1. \quad (2)$$



**Figure 5.** Fitness plot against different (a,b) pair values

The coefficient ‘a’ controls the importance of the classification error, while the coefficient ‘b’ corresponds to the importance of the length of the feature subset. Since our interest was primarily towards low classification error, we set ‘a’ to be 0.9 and ‘b’ to be 0.1. Several runs of the algorithm have proved the above assumption.

The fitness function plot for different pair values of a,b, for constant  $A(i)/A_{max|n}$ , is reported in figure 5. The sensitivity of  $F(i)$  in the cases of a,b pairs with a greater a value and a lesser b value increases for low  $E(i)/E_{max|n}$  ratios. Such low ratios imply

exceptional solutions among the population that should be appropriately rewarded.

The weighted standard error (WSE) is a modification of the root mean square error (RMSE). If  $e_t$  is the forecast error for the case  $i$ ,  $X_t$  is the actual value,  $F_t$  is the learning algorithm forecast value, then the simple RMSE is:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n e_t^2}{n}} \quad (3)$$

$$e_t = X_t - F_t \quad (4)$$

The WSE is computed as follows:

$$WSE = \sqrt{\frac{\sum_{t=1}^n (e_t \cdot X_t)^2}{\sum_{t=1}^n X_t^2}} \quad (5)$$

The modification made to RMSE aims at attributing more penalty at cases with large  $X_t$  than at cases with low  $X_t$ , having the same forecast error  $e_t$ . This is justified because errors at large pollutant concentration values have much more serious health and social impact than errors concerning low concentration values.

The k-NN algorithm used to evaluate each feature subset is a 3-NN with a Euclidean distance metric. The selection of the learning algorithm to make the feature subsets evaluation is based on the time needed for training and evaluation. The K-NN algorithm does not need any training, and the overall time spent during the training and evaluation phase is reasonably better than a decision tree or a neural network classifiers.

## 4. EXPERIMENTS

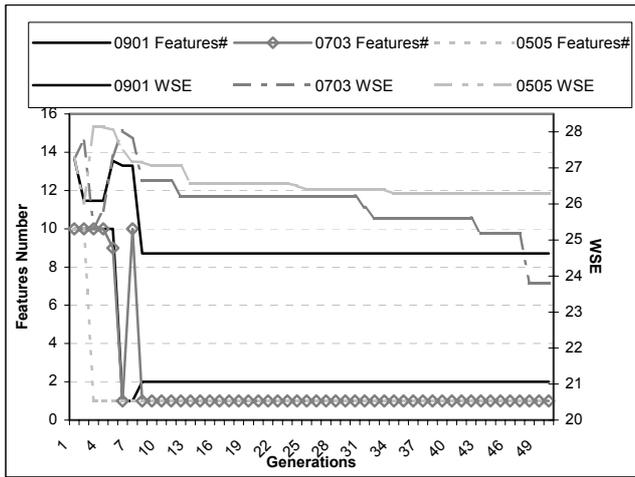
We run the genetic algorithm with the following parameters, for 3 different (a,b) pairs, aiming at predicting the day-0 maximum ozone concentration after 10:00 hour at the Patisson station (code 101):

- 30 phenotypes in the population
- 50 generations
- 3 elite phenotypes to be reproduced to next generation
- Mutation rate of 0.8

In figure 6, the WSE and the number of features per best solution per generation are presented. As seen in the legend, the first four digits in the names of the measured attributes correspond to one of the three (a, b) pairs:

- (0.9, 0.1),
- (0.7, 0.3) and
- (0.5, 0.5).

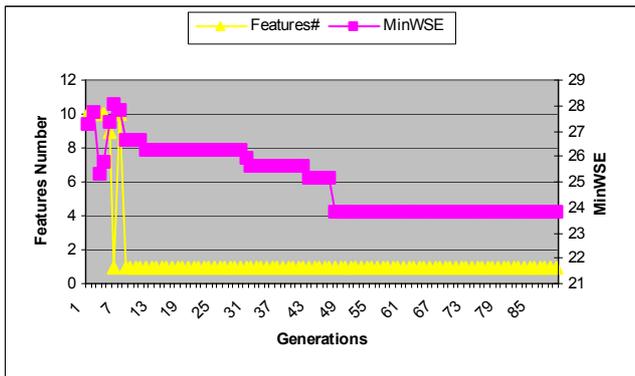
It can be observed that the convergence rate is much greater for large ‘a’ values and small ‘b’ values, not followed though by optimal solutions. The later can be found for intermediate values of the (a, b) pair of fitness parameters.



**Figure 6.** (a, b) pairs comparison for best solution per generation

In figure 6 an example run is solving the problem of predicting the day-0 maximum nitrogen dioxide concentration after 10:00 hour again at the Patissson station, with the following parameters:

- 30 phenotypes in the population
- 50 generations
- 1 elite phenotypes to be reproduced to next generation
- Mutation rate of 0.6
- $a=0.7, b=0.3$



**Figure 7.** Convergence of the best solution per generation

For every generation of the genetic algorithm a best proposed solution stands out. Figure 7 depicts the evolution of the best solution through each generation. The solutions that have been acquired after the experiments are recorded, are discussed in the next section.

## 5. SUMMARY AND DISCUSSION

What it was expected from the feature selection through the use of the genetic algorithm was apart from the efficiency improvement, the time and space relations between the input features and the output classes to be indicated by the best selected solutions. This should contribute to the evaluation of the basic knowledge that the experts have about the phenomenon, as well as to the apocalypses of the hidden physical time-space interactions.

For the problem of ozone prediction the following table summarizes the 5 best solutions over all runs sorted by WSE, followed by the features number for each solution.

**Table 1.** Best features selected for the ozone problem

|    | WSE     | Features# | Features                                  |
|----|---------|-----------|---|
| 1. | 23.81   | 1         | O3At106At1800Day-4                        |
| 2. | 25.685  | 1         | O3At108_Avg(900To2200)Day-8               |
| 3. | 24.4868 | 1         | O3At106At2000Day-2                        |
| 4. | 24.6261 | 2         | SO2At108At300Day-0,<br>SO2At106At400Day-0 |
| 5. | 24.9811 | 2         | SO2At108At300Day-0,<br>SO2At106At300Day-0 |

The results indicate a clear relation between station 101 (Patissson station) and stations 106 and 108. Past ozone readings for the hours between 18:00 and 20:00 and early SO<sub>2</sub> values have an important effect on the day-0 ozone maximum. The results indicate that the high O<sub>3</sub> concentrations at the central Patissson (code 101) station are originated from N-NW directions, where the Liosia (code 106) and the Peristeri (code 108) stations are situated.

In table 2 the corresponding solutions for the nitrogen dioxide problem are displayed:

**Table 2.** Best features selected for the nitrogen dioxide problem

|    | WSE     | Features# | Features  |
|----|---------|-----------|---|
| 1. | 76.2627 | 4         | NO2At105At1400Day-20,<br>NOAt101At600Day-0,<br>O3At105At1700Day-13,<br>NO2At101At900Day-0   |
| 2. | 77.3852 | 4         | NOAt105At1400Day-20,<br>NOAt101At600Day-0,<br>O3At105At1700Day-13,<br>NO2At101At900Day-0  |
| 3. | 77.9934 | 4         | NO2At105At1400Day-20,<br>NOAt101At500Day-0,<br>O3At105At1700Day-13,<br>NO2At101At900Day-0   |
| 4. | 80.0881 | 4         | O3At106At600Day-0,<br>NOAt101At600Day-0,<br>O3At105At1700Day-13,<br>NO2At101At900Day-0  |
| 5. | 80.7089 | 6         | SO2At101At600Day-14,<br>NOAt101At400Day-0,<br>O3At105At1700Day-13,<br>NO2At101At900Day-0,<br>NO2At106At100Day-0,<br>COAt103At900Day-0 |

For NO<sub>2</sub> prediction at 101 station, a space relation between stations 101 and 105 is revealed. Morning to mid-day hour measurements for NO and NO<sub>2</sub> during the same day-0 display an important predictive power. Also, measurements two or three weeks before day-0 seem to be important.

The results reported here indicate that the feature selection pre-processing using a genetic algorithm can be a useful tool for environmental problem solving, not only for improving the classification error of the subsequent learning algorithm, but also for revealing hidden time-space relations among the feature space. This paper showed also the modifications made to the genetic algorithm needed to achieve both convergence and accepted k-nn evaluation error.

## REFERENCES

- [1] John, G., Kohavi, R. & Pflieger, K. (1994). Irrelevant features and the subset selection problem, in W. W. Cohen & H. Hirsh (eds), *Machine Learning: Proceedings of the 11<sup>th</sup> International Conference*, Morgan Kaufmann, San Francisco, CA., pp. 121-129

- [2] Kalapanidas, E. And Avouris, N. (2001). Short-term air quality prediction using a case-based classifier, *Journal of Environmental Modeling and Software*, 16(3), 263-272.
- [3] Kalapanidas, E. and Avouris, N. (2002). Air Quality Management using a Multi-Agent System, *Int. J. Computer-Aided Civil and Infrastructure Engineering*, Special Issue in Environmental Applications of Artificial Intelligence, 17(2): 119-130. Blackwell.
- [4] Sucar, L.E., Perez-Brito, J., Ruiz-Suarez, J.C., Morales, E. (1997). Learning Structure from Data and its Application to Ozone Prediction, *Applied Intelligence*, 7(4), 327-338.
- [5] Perantonis, S. J., Vassilas, N., Amanatidis, G. T., Varoufakis, S. J. & Bartzis, J. G. (1994), Neural network techniques for SO<sub>2</sub> episode prediction, In S.-E. Gryng and M. M. Milan (Eds.), *Proceedings of the 20th Int. Tech. Meeting on Air Pollution Modelling and its Applications*, Valencia, Spain, Plenum Publishing Corp., New York, pp. 305–13.
- [6] Almuallim, H. and Dietterich, T. (1994). Learning boolean concepts in the presence of many irrelevant features. *Artificial Intelligence*, 69(1-2):279-305.
- [7] Kohavi, R. (1994). Feature subset selection as search with probabilistic estimates. In *AAAI Fall Symposium on Relevance*.
- [8] Narendra, P. and Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, 26:917-922.
- [9] Kira, K. and Rendell, L. (1992). A practical approach to feature selection. In *Proceedings of the Ninth International Conference on Machine Learning*, pages 249-256. Morgan Kaufmann.
- [10] Kononenko, I. (1994). Estimating attributes: Analysis and extension of relief. In *Proceedings of European Conference on Machine Learning*, pages 171-182.
- [11] Siedlecki, W. and Sklansky, J. (1989). A note on genetic algorithms for large-scale feature selection. *IEEE Transactions on Computers*, 10:335-347.
- [12] Punch, W., Goodman, E., Pei, M., Chia-Shun, L., Hovland, P., and Enbody, R. (1993). Further research on feature selection and classification using genetic algorithms. In *Proceedings of the International Conference on Genetic Algorithms*, pages 557-564. Springer.
- [13] Brill, F., Brown, D., and Martin, W. (1992). Fast genetic selection of features for neural network classifiers. *IEEE Transactions on Neural Networks*, 3(2):324-328.
- [14] Yang, J. and Honavar V. (1998). Feature selection using a genetic algorithm, in Motoda & Liu (eds), *Feature Extraction, Construction and Selection – A Data Mining Perspective*, Kluwer.
- [15] Jarmulak, J., and Craw, S. (1999). Genetic algorithms for feature selection and weighting. In *Proceedings of the IJCAI'99 workshop on Automating the Construction of Case Based Reasoners*. Cambridge, England, 1999.