Feature Selection for Multi-Purpose Predictive Models: a Many-Objective Task

Alan P. Reynolds, David W. Corne and Michael J. Chantler

School of Mathematical and Computer Sciences,
Heriot-Watt University, Edinburgh, Scotland.

Abstract. The target of machine learning is a predictive model that performs well on unseen data. Often, such a model has multiple intended uses, related to different points in the tradeoff between (e.g.) sensitivity and specificity. Moreover, when feature selection is required, different feature subsets will suit different target performance characteristics. Given a feature selection task with such multiple distinct requirements, one is in fact faced with a very-many-objective optimization task, whose target is a Pareto surface of feature subsets, each specialized for (e.g.) a different sensitivity/specificity tradeoff profile. We argue that this view has many advantages. We motivate, develop and test such an approach. We show that it can be achieved successfully using a dominance-based multiobjective algorithm, despite an arbitrarily large number of objectives.

1 Introduction

One of our motivating applications concerns images of textures (e.g. images of sections of wallpaper, fabric, carpet, etc.). Determining computationally whether two textures are similar to human eyes is a challenging and unsolved problem. However, experimental data are available that, for a varied set of textures, indicate which pairs users considered to be similar; we also have ∼5000 computational features for each texture. To support applications in texture search and browsing, we need to predict whether two textures are perceptually similar, using only the computational features. We also wish to reduce, via feature selection (FS), the number of features that need to be computed.

The selected features need to serve multiple purposes. Consider a search engine that, when given a 'query' texture, searches a database for other textures that would be perceived as being similar. Some users will be interested in as many 'matching' textures as possible and not be troubled by false positives. Others may require only a few textures but may insist that those provided be similar to the query case. Similar considerations apply in any domain where, for different predictive tasks involving the same data, the relative costs of false positives and false negatives vary significantly.

For such scenarios, in which FS is needed but the required performance profiles of the reduced feature set are complex and varied, we introduce a multiobjective (MO) approach that aims to find multiple subsets of features, each specialized for distinct required performance characteristics. In general, FS is easily

phrased as a MO problem, e.g. one may maximize accuracy while minimizing a measure of feature subset complexity [7–9]. However, based on the many ways of measuring accuracy, we argue that this may be considered a problem with an infinite set of objectives. We explain this in sections 2 and 3, showing how the choice of sensitivity and specificity as measures of classifier performance leads naturally to a problem with an infinite set of objectives. In sections 4 and 5 we then describe an algorithm capable of handling such a problem. Sections 6 and 7 describe an investigation of this algorithm on three datasets. The effectiveness of the approach is discussed in section 8, along with ideas for further work.

2 Feature Subset Evaluation in the Wrapper Approach

When selecting features for a particular target application, the quality of the feature set is determined by the resulting performance of the application. Here we consider two-class problems, with the 'class of interest' considered 'positive' and the other 'negative'. Performance is calculated using the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). We make particular use of the following measures: *sensitivity* ($|TP|/(|TP|+|FN|)$); *specificity* ($|TN|/(|TN|+|FP|)$); and *confidence* ($|TP|/(|TP|+|FP|)$).

In the 'wrapper' approach to FS, feature set quality is estimated by applying a simple classification algorithm. So if the balanced error rate is to be minimized in the target application, the evaluation of a feature set should be an attempt to minimize the balanced error rate using a suitable classifier. If, as in the case of the texture search engine, the target application's performance is judged according to multiple, perhaps unknown accuracy measures, then feature subset evaluation should be an attempt to optimize each of these measures. In this case, use of a classification algorithm such as basic k-Nearest Neighbour (k-NN, as used by Emanouilidis [4], in an effort to optimize specificity, sensitivity and feature set size) is not appropriate, since k-NN generates only a single sensitivity-specificity pair that cannot simultaneously optimize each of the competing objectives. On the other hand, a model-based algorithm such as naive Bayes (NB) produces a probability that each record belongs to the class of interest. Concrete predictions are obtained by assigning a record to the class of interest if the probability is above a threshold. By varying the threshold, NB classifiers produce a range of different sensitivity-specificity trade-offs.

3 Uncountably Many-Objective Feature Selection

Users of a texture search engine will have varying preferences for the balance to be struck between sensitivity and specificity. Figure 1 shows the results of applying a classifier like NB to a single feature subset and the preferences of three users. User 1 is happy with just a few (12%) similar textures being returned, but is irritated by false positives. User 3 requires almost all (94%) similar textures to be returned, and will tolerate a large number of false positives. User 2 takes the middle ground, being satisfied with a sensitivity of 48%. Each user has set

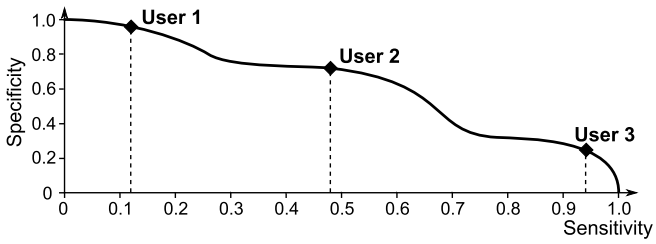

Fig. 1. User 1 is happy with few correct matches, but is easily irritated by false positives. User 3 requires most truly similar textures to be provided and can tolerate many false positives. User 2 strikes a balance between these extremes.

a threshold on sensitivity and wishes specificity to be maximized subject to this constraint. Users may state their requirements in different ways, e.g. via estimates of the relative costs of false positives and false negatives, yet clearly we wish the graph to be as high as possible at each value of sensitivity.

Conceptually, this results in uncountably many objectives: to please each potential user of our search engine we should maximize specificity for each value of sensitivity. In practice, however, the graph of specificity against sensitivity is piecewise horizontal, with the number of pieces bounded by the number of records in the class of interest. This reduces the number of objectives to 'very many'. The height of neighbouring points on the graph are also highly correlated, which increases the chance that any pair of feature sets are comparable, i.e. that one dominates the other. Finally, section 4 introduces modified dominance relations that further increase the probability that a pair of solutions are comparable, enabling an effective dominance-based approach to this problem. Meanwhile, note that the methods developed here can handle the conceptually infinite-objectives case — the resulting dominance relations and crowding measures are suitable for the comparison of graphs, rather than vectors of objectives.

There are many approaches to evaluating feature subsets; in this paper we examine two. In each case, an objective is the value of some measure of quality at a fixed value of some other quality measure or parameter. The first approach plots specificity against sensitivity (equivalent to optimizing ROC curves [5]); the second plots confidence against sensitivity. In either case, if a threshold value produces a sensitivity-specificity pair or a sensitivity-confidence pair that is dominated by some other pair, it is not considered part of the curve produced.

4 Dominance and Crowding

The basic dominance relation is familiar: one solution dominates another if it is at least as good on all objectives and better on at least one. In the case of specificity vs. sensitivity curves, this translates to the curve for the first solution being at least as high as the curve for the second in all places, and higher in

some. While this seems reasonable, there is a concern that the large number of objectives will result in a weak dominance relation. (Here, a dominance relation is considered 'strong' if, given a random pair of solutions, the probability that they are comparable is high.) Solution B need only beat solution A over a tiny portion of the curve in order to avoid being dominated by A. This may result in a lack of selection pressure in dominance based algorithms such as NSGA II [3] and a potentially unmanageable number of non-dominated solutions. Hence we apply a simple modification to the dominance relation. In Fig. 2 the area in dark

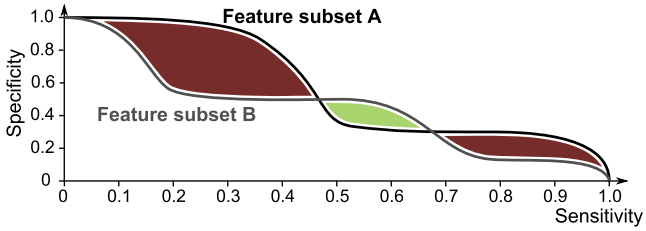

Fig. 2. Dividing the area of dark red by the area of pale green gives a value used by the modified dominance relation.

red is dominated by feature set A but not by feature set B, while the pale green area is dominated by B but not A. The modified dominance relation states that A dominates B if there is a red area but no green area, or if the result of dividing the red area by the green area exceeds a given *dominance factor*. A dominance factor of 1 makes almost every pair of solutions comparable, reducing to the problem of maximizing the area under the curve — a commonly used measure of the performance of a machine learning algorithm (e.g. [2]). At the other extreme, an infinite dominance factor produces the basic dominance relation.

Any well-behaved dominance relation should be anti-symmetric and transitive. If the dominance factor is at least 1, then the relation is clearly anti-symmetric — if the red area is bigger than the green area then the green area cannot be bigger than the red area. Transitivity can also be shown, though this requires a little more work. A brief proof is available in supplementary material at `http://www.macs.hw.ac.uk/~ar136`, along with our code and datasets.

Finally, to underpin maintenance of a limited-size archive of non-dominated solutions, we consider the choice of 'crowding' measure. It is possible to generalize the standard crowding measure of NSGA II [3], but we elect to use a crowding measure based on distances between pairs of solutions. We define this as the area between the two curves, i.e. the sum of the red and green areas in Fig. 2.

5 Implementation

Two classification algorithms, logistic regression (LR) and naive Bayes (with Laplace correction) (NB) [10], are used to evaluate the feature subsets. LR re-

quires numeric data, so categorical data were converted. For example, a categorical field with three categories, cyan, magenta and yellow, is converted into two numeric fields, taking the values zero and one. A one in the first field translates to 'cyan', while a one in the second field translates to 'magenta'. Two zeroes imply that the record is 'yellow'. In contrast, our implementation of NB requires categorical data. Any numeric field is discretized by partitioning its range into a number of bins. To avoid problems that may arise from a highly non-uniform distribution over the bins, we aim for an equal-frequency discretization. (Details may be found in the supplementary material.)

Feature subsets were optimized using NSGA II [3], with the dominance relation and crowding measure replaced by those described above. Solutions were encoded as bitstrings, with bits indicating the presence or absence of the corresponding feature. A limit on the number of features was imposed and enforced after crossover by removing random features as necessary. Three types of mutation were used at equal rates: addition of a feature (if permitted), removal of a feature or swapping a feature in the subset for one currently absent.

It has been suggested that dominance based algorithms such as NSGA II perform poorly for problems with more than 4 objectives [6], most likely due to the resulting weak dominance relation. Here we illustrate that despite the large number of objectives, good results can be obtained if the dominance relation is strong enough.

Finally, we note that, after performing the optimization, the presence of so many objectives raises issues with the presentation of the results. Figure 3 shows quality plots for four non-dominated feature subsets. Their quality may

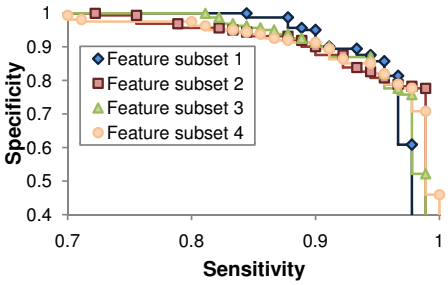

Fig. 3. Quality plots for four feature subsets, generated from the ionosphere data with dominance factor set to 2 and a limit of 4 features per subset.

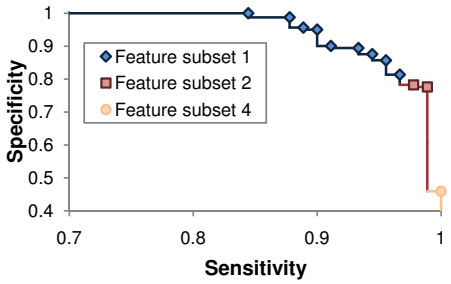

Fig. 4. Plotting the envelope of the quality plots for four feature subsets. Notice that feature subset 3 does not contribute, despite being non-dominated.

be compared without too much difficulty using the figure. However, this task becomes much more difficult given twenty or thirty non-dominated solutions.

One possibility is to plot only those points that are non-dominated with regard to sensitivity and specificity, producing Fig. 4. Notice that feature subset

3 does not contribute to the sensitivity-specificity front. However, this subset may be the best choice, since it performs well over all values of sensitivity. Moreover, this method of presenting results may result in an overoptimistic view of certain feature subsets on unseen data. An alternative approach is to present two types of graph. The main graph plots solutions according to, e.g., the specificity at two different values of sensitivity. Selecting a solution produces a second graph of specificity against sensitivity for the solution including two markers, in this case vertical lines, that indicate the objectives used on the main graph. With a suitable user interface, dragging these markers changes the objectives used on the main graph, allowing the user to fully explore the solution set.

6 Experimentation

We explore this approach to generating multiple feature subsets by testing our algorithm on three datasets. In each case the result is a Pareto front of feature subsets, each of which has its own characteristic tradeoff curve (e.g. specificity vs. sensitivity). In evaluating the technique, we are constrained by the fact that there are as yet no suitable alternative algorithms in the literature that address the same problem. The closest is the approach of Emanouilidis [4]. However, we maximize specificity for each possible value of sensitivity. Emanouilidis's use of 1-NN as the core classifier means that only a single sensitivity-specificity pair is obtained for each feature set and it is these single values of sensitivity and specificity that are maximized. Hence the algorithms optimize different measures of feature set quality. (Note that we can compare the best sensitivity-specificity values obtained by the two approaches, where we might expect Emanouilidis's use of 1-NN to restrict the spread of solutions across the sensitivity-specificity front.) Evaluation of our approach is therefore restricted mainly to illustrating that it achieves apparently effective results on the three datasets studied, in each case yielding a set of feature subsets with varied performance characteristics. Beyond this, we also report on aspects of performance that vary according to the dominance factor, and according to constraints on the size of feature subsets.

In all experiments, we used a crossover rate of 0.8, a mutation rate (the chance that a solution is mutated) of 0.2, a population size of 100, 500 generations, and 10-bin discretization when NB was used as the core classifier. Each dataset is split into training and test sets, used during optimization and final evaluation respectively. Whenever a feature subset is evaluated on the training or test set, cross-validation (CV) is used — leave-one-out-CV for the ionosphere data, and 5-fold CV in the other two cases. Dataset details are as follows:

Ionosphere: Available from [1] and used previously for MOFS [4], the ionosphere data comprises 351 records and 35 fields. The class field is either "good" (g) or "bad" (b), where "bad" is the class of interest. Non-class fields are numeric. The dataset was split into training and test data, with the test set containing 100 records, 36 in the class of interest.

Breast Cancer Wisconsin (Diagnostic): Again from [1], this dataset has a class field that takes the value 'M' (malignant) or 'B' (benign) and has

30 numeric input fields. The class of interest is the malignant class. The 569 records, 212 in the class of interest, are divided randomly into training and test sets, with the test set containing 169 records, including 63 in the class of interest.

Texture: In the texture data, each record corresponds with a pair of textures. 5376 numerical features were extracted computationally from a set of textures, using a range of methods including spectrum analysis, radon transforms, auto-correlation etc. This was reduced to 283 features using simple correlation-based methods. The input fields were obtained by calculating feature differences for each texture pair. The class field was obtained by asking 30 people to group similar textures, given either the full set of textures or a subset. Two textures were considered similar if at least a third of people grouped the pair together. The training set involved 19900 texture pairs (200 textures), 333 of which were considered similar. The test set was produced using another 100 textures, generating 4950 records, 85 in the class of interest.

The texture data is much larger than the other datasets, providing a more challenging test. The class of interest forms only a small part of the dataset, making it difficult to make true positive predictions without introducing many false positives, i.e. it is difficult to achieve high confidence values.

7 Results

First, to examine the effect of the dominance factor and to determine a suitable value for this parameter, experiments were performed on the ionosphere data using an upper limit of 4 features per feature subset. NB was used to evaluate feature sets, with sensitivity-specificity curves used to determine feature set quality. The time taken by the algorithm and the numbers of non-dominated solutions obtained, averaged over 30 runs, are outlined in Table 1. The last column corresponds with the use of the basic dominance relation.

Table 1. The effect of modifying the dominance factor.

Dominance factor	1	2	5	10	20	50	∞
No. non-dominated	1.00	3.77	16.0	41.9	85.7	187	397
Time (s)	20.7	23.0	23.9	25.5	26.9	28.4	20.3

Note that the number of non-dominated solutions is small compared with the number of solutions examined, even when the basic dominance relation is used. This implies that the dominance relation is stronger than one might expect for a problem with so many objectives. However, given the difficulty in comparing 397 feature subsets, modified dominance is used in the following experiments.

On the Ionosphere data, the algorithm was applied using NB, sensitivity-specificity curves and a dominance factor of 5. Runs were performed with different limits on the number of features. Table 2 shows the number of non-dominated

Table 2. Number of non-dominated solutions for different feature set size limits.

Max. features	1	2	3	4	5	6	7	8
No. non-dominated	7.00	19.0	16.0	16.0	36.0	32.4	70.0	173
Time (s)	11.0	19.6	20.1	23.6	26.1	27.5	30.7	34.2

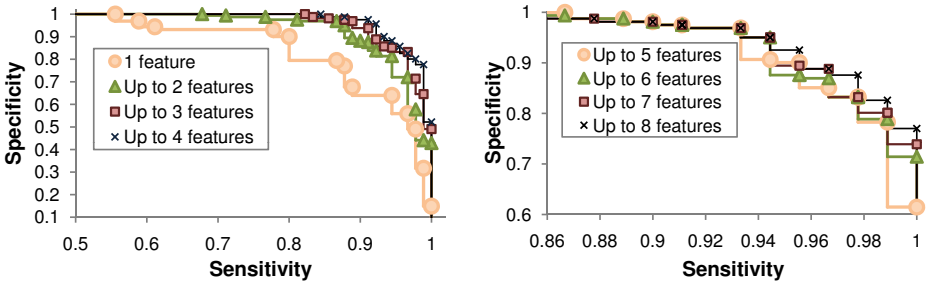

Fig. 5. Performance on Ionosphere training data.

solutions obtained and time requirements, averaged over 30 runs. Time taken was sufficient to find all Pareto-optimal solutions for subsets of up to 5 features, in all runs. Results on training and test data are shown in Figs. 5 and 6 respectively. (For clarity, Figs. 5–9 show the envelope of the sensitivity-specificity curves.)

Comparing with [4], the most notable difference is that the results presented in Figs. 5 and 6 cover a much broader range of sensitivity-specificity values, since the classification algorithm used is capable of effectively evaluating feature sets that perform well at either high sensitivity or high specificity values.

On the breast cancer data, the algorithm was applied using LR, sensitivity-confidence curves and a dominance factor of 5. Typical time requirements were 23 min. for four features and 32 min. for eight. Results are shown in Figs. 7 and 8. Finally, on the texture data the algorithm was applied using NB, sensitivity-

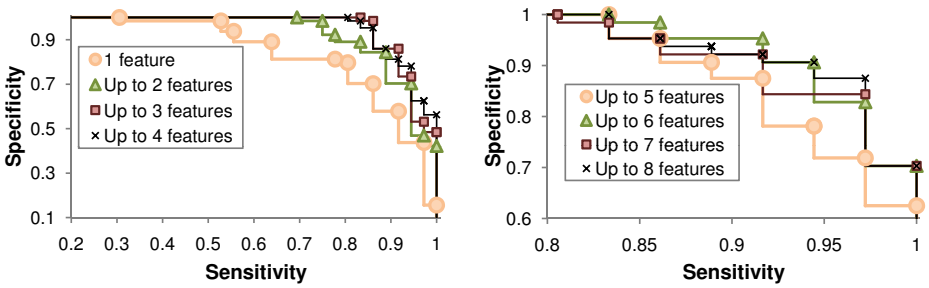

Fig. 6. Performance on Ionosphere test data.

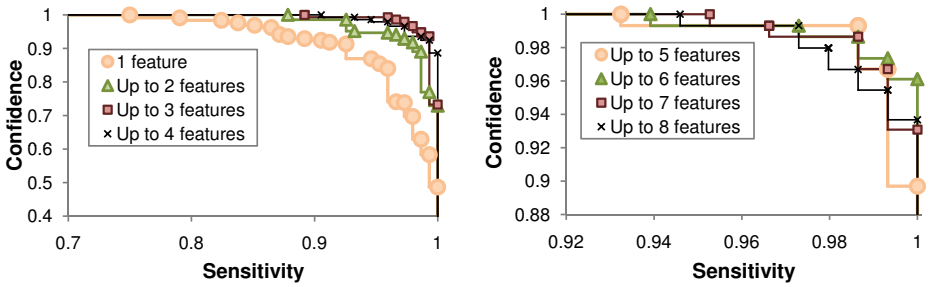

Fig. 7. Performance on Breast Cancer Wisconsin (diagnostic) training data.

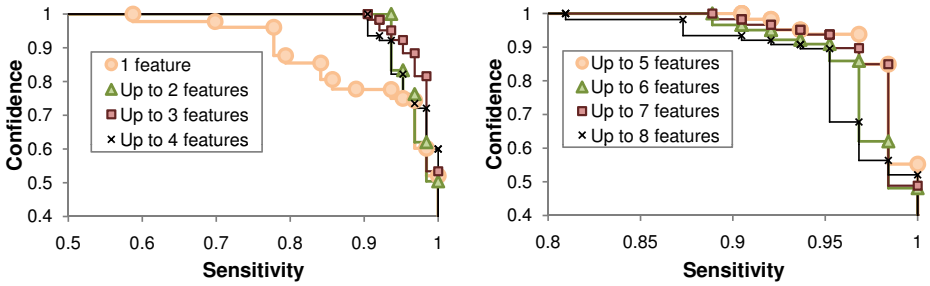

Fig. 8. Performance on Breast Cancer Wisconsin (diagnostic) test data.

confidence curves and a dominance factor of 2. Typical time requirements were 23 min. for four features and 29 min. for eight. Results are in Fig. 9.

8 Discussion

This paper has shown that FS can be effectively treated as a MO optimization problem with an infinite set of objectives. The approach has advantages over other FS methods, in that each feature subset generated is evaluated across a range of values of sensitivity. There are many possible avenues of further research. For example, if one is only interested in the feature sets that contribute to the overall sensitivity-specificity (or sensitivity-confidence) front, an alternative approach to dominance is indicated. Meanwhile, the class field in the texture data originally indicated the proportion of people that considered a pair of textures to be similar. This can be considered as the 'probability of class membership'. Here we used a threshold to convert this into a binary field. However, research should be performed into adapting the approach of this paper to probabilistic class membership. Finally, an obvious avenue of further work is to generalize the method to multi-class problems. For example, given a three class problem we may evaluate feature subsets according to the accuracy on each class. So rather than dealing with curves and the area between curves, the problem becomes one of surfaces and the volume between surfaces.

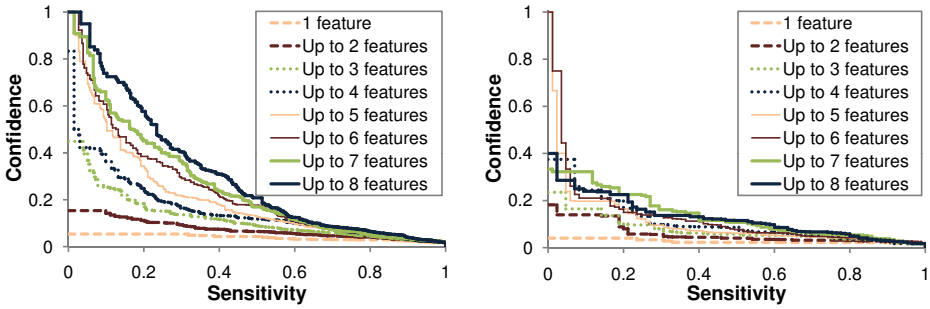

Fig. 9. Performance on texture data, for one to eight features; results on training (left) and test (right) data.

References

1. Asuncion, A., Newman, D.: UCI machine learning repository (2007), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
2. Bradley, A.P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition 30(7), 1145–1159 (1997)
3. Deb, K., Agrawal, S., Pratab, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J., Schwefel, H.P. (eds.) PPSN VI. LNCS, vol. 1917, pp. 849–858. Springer (2000)
4. Emmanouilidis, C.: Evolutionary multi-objective feature selection and ROC analysis with application to industrial machinery fault diagnosis. In: Giannakoglou, K., Tsahalis, D., Periaux, J., Papailiou, K., Fogarty, T. (eds.) Evolutionary Methods for Design, Optimisation and Control. CIMNE (2002)
5. Fawcett, T.: An introduction to ROC analysis. Pattern Recognition Letters 27, 861–874 (2006)
6. Hughes, E.J.: Evolutionary many-objective optimisation: Many once or one many? In: Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC 2005). vol. 1, pp. 222–227. IEEE Service Center (2005)
7. Oliveira, L.S., Sabourin, R., Bortolozzi, F., Suen, C.Y.: Feature selection using multi-objective genetic algorithms for handwritten digit recognition. In: Proceedings of the 16th International Conference on Pattern Recognition (ICPR-02). vol. 1, pp. 568–571. IEEE Computer Society (2002)
8. Oliveira, L.S., Sabourin, R., Bortolozzi, F., Suen, C.Y.: A methodology for feature selection using multi-objective genetic algorithms for handwritten digit string recognition. International Journal of Pattern Recognition and Artificial Intelligence 17(6), 903–929 (2003)
9. Pappa, G.L., Freitas, A.A., Kaestner, C.A.A.: A multiobjective genetic algorithm for attribute selection. In: Proceedings of the 4th International Conference on Recent Advances in Soft Computing (RASC-2002). pp. 116–121 (2002)
10. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, second edn. (2005)