New strategies for OVO Feature Selection on Multiclass Problems

Javier Izetta, Guillermo L. Grinblat and Pablo M. Granitto

CIFASIS
French Argentine International Center for Information and Systems Sciences
UPCAM (France) / UNR–CONICET (Argentina)
Bv 27 de Febrero 210 Bis, 2000 Rosario, República Argentina
{izetta,grinblat,granitto}@cifasis-conicet.gov.ar

Abstract. Feature selection is a useful machine learning technique aimed at reducing the dimensionality of the input space, discarding useless or redundant variables, in order to increase the performance and interpretability of models. The well-known Recursive Feature Elimination (RFE) algorithm provides good performance with moderate computational efforts, in particular for wide datasets. When using Support Vector Machines (SVM) for multiclass classification problems, the most typical strategy is to apply a simple One–Vs–One (OVO) strategy to produce a multiclass classifier starting from binary ones. In this work we introduce improved methods to produce the final ranking of features on multiclass problems with OVO–SVM, based on different combinations of the set of rankings produced by the diverse binary problems. We evaluated our new strategies using wide datasets from mass–spectrometry analysis and standard datasets from the UCI repository. In particular, we compared the new methods with the traditional average strategy. Our results suggest that one of our new methods outperforms the traditional scheme in most situations.

1 Introduction

In-silico chemistry [1] and biology, "high-throughput" technologies [2] or text processing are current problems of high technological importance in machine learning, which share the characteristic of presenting much more features than measured samples [3] (wide datasets). Usually, most of these variables have a relatively low importance for the problem at hand. Furthermore, in some cases they interfere with the learning process instead of helping it, a problem usually known as "the curse of dimensionality".

Feature selection is a useful pre–processing technique aimed at the solution of this problem[4]. Its main goal is to find a small subset of the measured variables that improve, or at least do not degrade, the performance of the modeling method applied to the dataset. But feature selection methods do not only avoid the curse of dimensionality, they also allow a considerable reduction in model complexity, an easier visualization and, in particular, a better interpretation of the data under analysis and the developed models [5].

The well-known Recursive Feature Elimination (RFE) algorithm provides good performance with moderate computational efforts [6] on wide datasets. The original and most popular version of this method uses a linear Support Vector Machine (SVM) [7] to select the features to be eliminated, which is widely used in Bioinformatics [6, 8]. Alternative methods were introduced by Granitto et al. [9, 10] and Izetta et al. [11], which basically replace SVM with Random Forest or ANN ensembles into the core of the RFE method.

Typical feature selection algorithms are designed for binary classification problems. Multiclass problems have received much less attention, because of their increased difficulty and also because some classifiers (needed for the selection) are designed to solve binary problems. Most methods available for feature selection on multiclass problems are simple extensions of base methods. For example, RFE can be associated to a multiclass classifier like Random Forest [12].

Although SVM was originally developed to deal only with binary problems, it was extended to solve multiclass problems in different ways [13, 14], but with low success. On the other hand, in the last years several methods were developed to solve a multiclass problem using an appropriate combination of binary classifiers [15, 14]. The most used strategy for multiclass SVM is known as "One–vs–One" (OVO). In this case a problem with c classes is replaced with $c(c-1)/2$ reduced problems, each one consisting in discriminating a pair of classes. Thus, the most common way of implementing a multiclass SVM-RFE method is to use directly the RFE algorithm over an OVO-SVM.

Interestingly, this common solution to the multiclass RFE-SVM problem involves some decisions about the feature selection process that are usually neglected. In particular, the algorithm uses a very simple solution to the problem of selecting candidate features from multiple lists [16]. As the original problem is decomposed into several binary sub-problems, the algorithm produces a list of relevant features for each one of these sub-problems. In this work we review the base strategy used in this case and propose two new strategies aimed at an efficient selection of features from multiple candidates coming from the several OVO binary problems.

The rest of this article is organized as follows: in Section 2, we describe the typical OVO–RFE feature selection scheme and discuss our two new strategies. In Section 3 we evaluate the three methods using wide and normal datasets. Finally, we draw some conclusions in Section 4.

2 Multiclass RFE

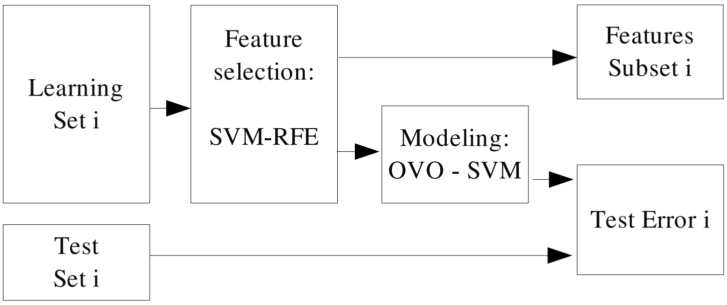
Granitto et al. [17] explain that the RFE selection method [6] is a recursive process that ranks variables according to a given measure of their importance. At each iteration the importance of each feature is measured and the less relevant one is removed. Another possibility, which is the most commonly used, is to remove a group of features each time, in order to speed up the process. Usually, 10% of the variables are removed at each step until the number of variables reaches a lower limit, and from that point on the variables are removed one at a time [18]. The recursion is needed because for some measures the relative importance of each feature can change substantially when evaluated over a different subset of features during the stepwise elimination process (in particular for highly correlated features). The (inverse) order in which features are eliminated is used to construct a final ranking. The feature selection process itself consists only in taking the first n features from this ranking. In the original binary version of SVM–RFE [6], the projection of W (the normal vector to SVM's decision hyperplane) in the direction of each feature is used as the measure of importance.

As we discussed in the previous section, the OVO strategy is the most typical way to produce multiclass feature selections based on SVM classifiers. In this case, a problem with c classes is decomposed into $p = c(c-1)/2$ binary problems discriminating between all possible pairs of classes. At each step of the RFE algorithm an OVO–SVM is adjusted using the training data over all the remaining features and a single ranking of variables is produced, from which some variables are eliminated. To produce a single ranking from the set of binary problems, the method simply averages the (unsigned) components of W from all the binary problems. In the following, we will call this base method the Average SVM–RFE. The drawback of this strategy is that a feature that is crucial for a given class but useless in other cases is usually ranked bellow other features that have a moderate relevance for all the classes (because of the simple averaging over the components of W).

On the other side, this problem can be viewed as the problem of selecting candidates from multiple ranked lists [16]. In fact, we can follow an "a posteriori" strategy, letting each OVO problem produce a complete ranking of features for each individual problem using binary SVM–RFE, and then using voting schemes to produce combined rankings valid for the complete multiclass problem. Our goal here is to give more relevance to features that are the most discriminant for some binary problems and on the other hand to reduce the ranking of "average" features that never reach top positions in the individual rankings.

With this in mind we implemented two different schemes. The first strategy is called Best Ranking. For each feature we keep only the best ranking among the p binary problems and we produce the final ranking based on that information. When we have a tie (as many different features could reach the same best position), we order those features according to the average position over the p rankings. The second strategy is called K-First. The general idea in this case is to consider not only the best position but a group of features ranked at top positions. To this end, for each binary problem we selected the top K features and assigned each one a weight that is proportional to its position in that reduced ranking (i.e., we give a weight of 1 to the top feature, 0 to the $K+1$ one, and a linear scale in between). After that, we produce the final ranking averaging these weights over the p lists. The K parameter completely regulates this strategy. Changing its value, the K-First strategy could be more similar to the Average strategy (for high K values) or to the Best Ranking strategy (for

Fig. 1. The computational setup used for the feature selection process



very low K values). In this work we used a fixed value for K equal to 10% of the total number of features in the problem.

3 Experiments

3.1 Experimental setup

It is well-known that a feature selection method that uses (in any way) information about the targets may lead to overfitting, in particular with wide datasets. Thus, an appropriate experimental setup is needed for these experiments [19].

As in previous works [11], we use a computational setup consisting of two nested processes. The outer loop performs n times a random split of the dataset in a training set (used to develop the models – including the feature selection step), and in a test set, used to estimate the accuracy of the models. The inner process (Figure 1) supports the selection of nested subsets of features and the development of classifiers over these subsets (using only the learning subset provided by the outer loop). The results of the n replicated experiments are then aggregated to obtain error rates estimation.

3.2 Datasets

We used five real world datasets, listed in Table 1. The first three problems are wide datasets, while the last two are traditional "tall" problems that were included to check the new methods also in this more typical context. The first two refer to cultivar characterization of berry fruits (Strawberries [20] and Raspberries) and the third one to typicality assessment of Grana cheeses [21]. All products come from Trento Province, North Italy, or other places in the same area. In all cases the headspace composition of the samples has been measured by direct injection in a PTRMS apparatus (experimental details can be found in previous papers [20]). Each sample was then associated to its PTR-MS spectrum normalized to unit total area. The last two datasets were obtained from the UCI repository [22] and were selected for being real world data with several classes.

For all cases we replicate the feature selection process on $n = 100$ runs. For each run, we split the dataset at random into train/test sets with a 75%/25% proportion, stratifying on class frequencies. Each train set is used for the three SVM-RFE strategies to select features and to develop models, which are then evaluated on the corresponding test set.

3.3 Classification Errors

In Figures 2, 3 and 4 we compare the three selection methods on the PTR-MS datasets. In all cases we show mean classification errors (± the standard error of the mean) as a function of the number of features selected at each step for the three methods. In the last two cases the K-First strategy clearly produces the lowest error rates, while in the Fragola case there are not clear differences in this aspect. On the other hand, it is always interesting to analyze the behavior of the methods when selecting only a few features. In this case the results are opposite, in two out of the three problems K-First shows the worst results.

In Figures 5 and 6 we show the corresponding results for the two tall datasets. Qualitatively, the results are similar to the wide datasets. In both problems the K-First strategy shows the minimum error rate and also its performance degrades when considering just a few features.

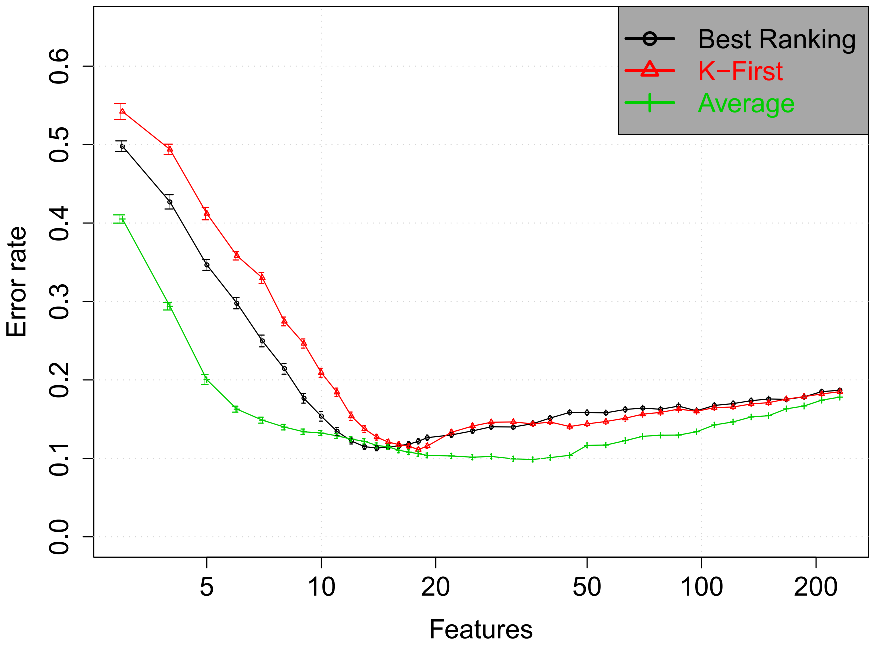
4 Analysis and Conclusions

In this exploratory work we have introduced two new strategies for feature selection on multiclass problems, all based on the well-known SVM-RFE method. We discussed the traditional OVO strategy (Average) and its limitations and

Table 1. Details on the five datasets used in this work.

Dataset	Variables	Samples	Classes
Fragola	232	233	9
Lampone	232	92	5
Grana	235	60	4
Satimage	36	500	6
Pendigits	16	500	10

Fig. 2. Error rates as a function of number of variables selected by the three RFE methods for the Fragola dataset.

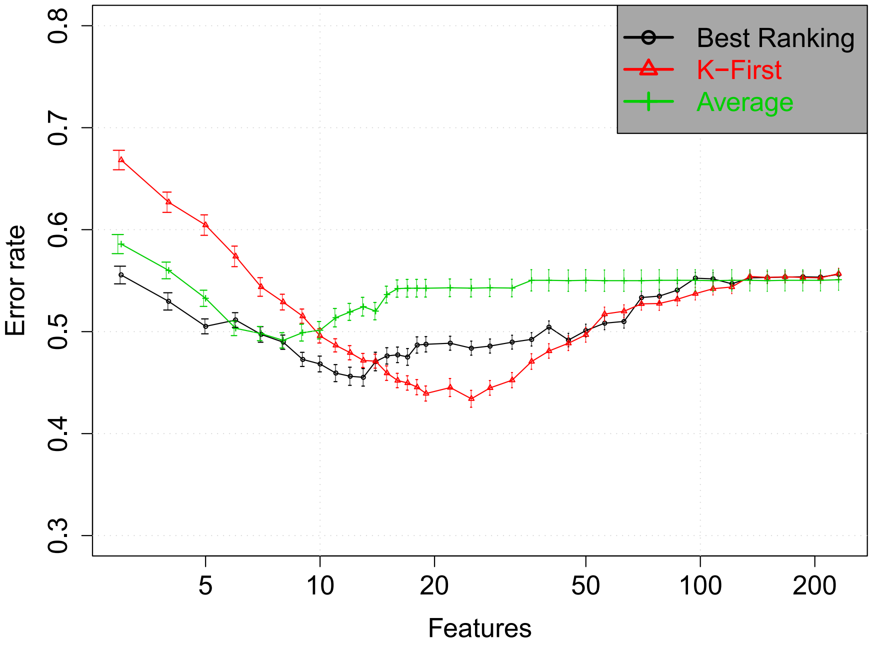


described the Best Ranking and K-First strategies. Both methods are based on simple strategies to produce ranked list from multiples sources in voting theory.

The Best Ranking strategy works by giving high rankings to features that are very relevant at least for one problem. Our results suggest that this strategy is not superior to the traditional method. It is probable that this method is discarding relevant features (that are not ranked strictly at the first position) at early stages of the process.

The K-First strategy, on the other hand, considers a subset of very relevant features for each binary problem. Our preliminary results suggest that this strategy has the potential to discard redundant or irrelevant features at the first stages of the method, leading in almost all cases to the lowest error rates. However, the method clearly loss performance when working with a few features. We believe that this is related to our use of a fixed K value. It is evident in all figures that K-First's results deteriorates when working with less than K features. More work is needed in this direction in order to evaluate diverse alternatives to our current simple threshold strategy.

Fig. 3. Error rates as a function of number of variables selected by the three RFE methods for the Lampone dataset.



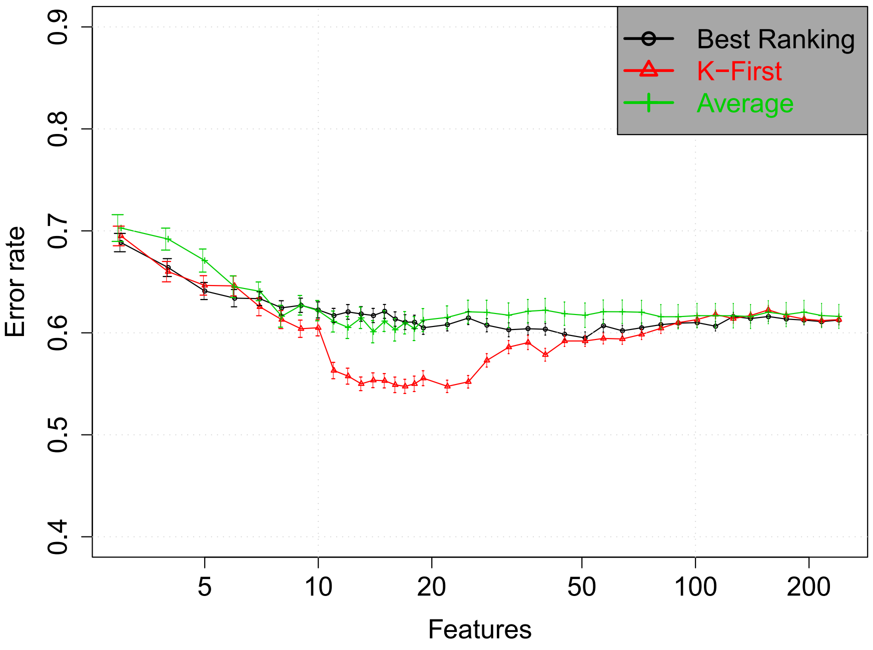
Acknowledgments

We acknowledge partial support for this project from ANPCyT (grant PICT 237).

References

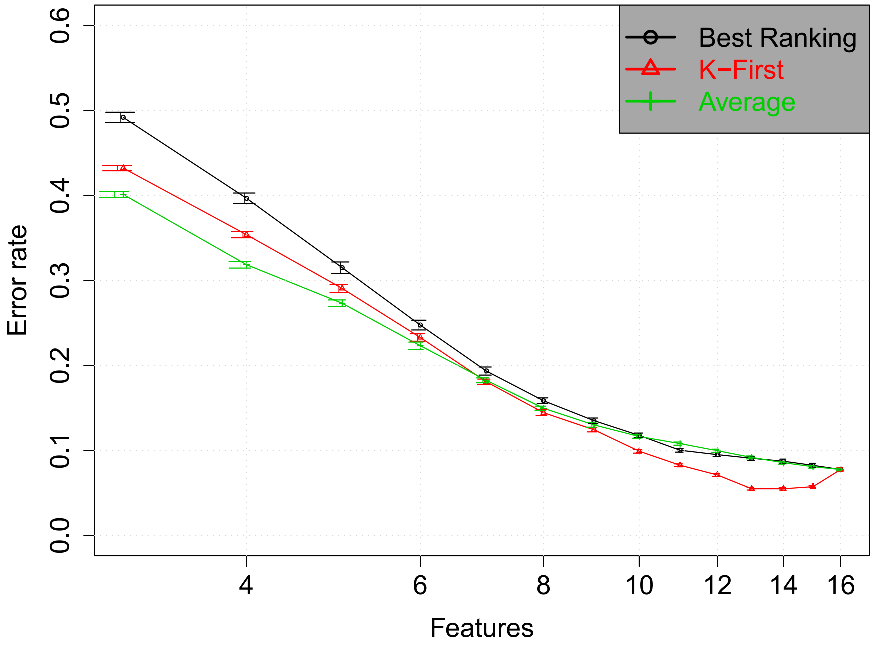
1. Li, H., Ung, C. Y., Yap, C. W., Xue, Y., Li, Z. R., Cao, Z. W., Chen Y. Z.: Prediction of Genotoxicity of Chemical Compounds by Statistical Learning Methods. Chem. Res. Toxicol. 18, 1071–1080 (2005)
2. T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression, Science, 286:531-537, 1999.
3. Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. J. Mach. Learn. Res. 3, 1157–1182 (2003)
4. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artif. Intell. 97, 273–324 (1996)

Fig. 4. Error rates as a function of number of variables selected by the three RFE methods for the Grana dataset.



5. Huan Liu, Edward R. Dougherty, Jennifer G. Dy, Kari Torkkola, Eugene Tuv, Hanchuan Peng, Chris Ding, Fuhui Long, Michael Berens, Lance Parsons, Zheng Zhao, Lei Yu, George Forman: Evolving Feature Selection, IEEE Intelligent Systems, v.20 n.6, p.64-76, November 2005.
6. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene Selection for Cancer Classification using Support Vector Machines. Mach. Learn. 46, 389–422 (2002)
7. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
8. Alexander Statnikov, Constantin F. Aliferis, Ioannis Tsamardinos, Douglas Hardin and Shawn Levy. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics, 21:5, 631–643, (2005).
9. Granitto, P.M., Biasioli, F., Gasperi, F., Furlanello, C.: Modeling Sensory Analysis datasets: the case of Italian Cheeses. Proceedings of JAIIO 2005 - The 34th International Conference of the Argentine Computer Science and Operational Research Society, Rosario, Argentina (2005).
10. Granitto, P.M., Furlanello, C., Biasioli, F., Gasperi, F.: Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products. Chemometr. Intell. Lab. 83, 83–90 (2006)
11. Izetta J., Granitto P. M.: Feature selection with simple ANN ensembles, Proceedings of CACIC 2009 - XV Argentine Congress on Computer Science, Jujuy,

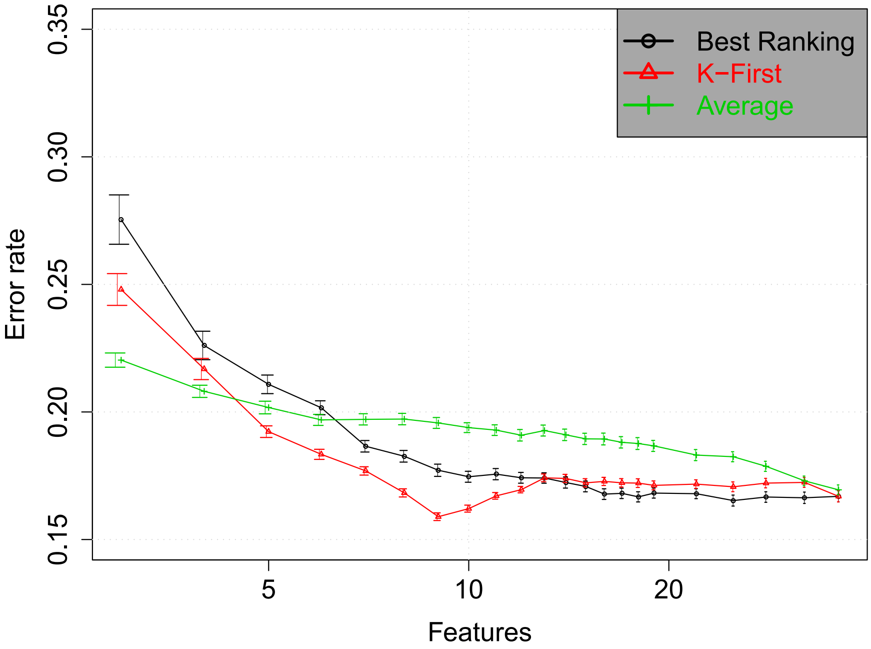
Fig. 5. Error rates as a function of number of variables selected by the three RFE methods for the Pendigits dataset.



Argentina, (2009).

12. Breiman, L.: Random Forests. Mach. Learn. 45, 5–32 (2001)
13. Crammer, K., Singer, Y., On the Learnability and Design of Output Codes for Multiclass Problems, Machine Learning, 47, 201233, 2002.
14. Hsu, C.-W., Lin C.-J.: A comparison of methods for multi-class support vector machines , IEEE T. Neural Networ., 13 415-425 (2002)
15. Allwein, E., Schapire, R., Singer, Y.: Reducing Multiclass to Binary: A unified Approach for Margin Classifiers. J. Mach. Learn. Res. 1, 113-141 (2000)
16. G. Jurman, S. Merler, A. Barla, S. Paoli, A. Galea, C. Furlanello: Algebraic stability indicators for ranked lists in molecular profiling. Bioinformatics, 24(2), 258-264 (2008)
17. Granitto, P. M., Biasioli, F., Furlanello C., Gasperi, F., Efficient Feature Selection for PTR-MS Fingerprinting of Agroindustrial Products. Proceedings of ICANN08, 18th International Conference on Artificial Neural Network, Prague, Czech Republic, (2008).
18. Furlanello, C., Serafini, M., Merler, S., Jurman, G.: Entropy-Based Gene Ranking without Selection Bias for the Predictive Classification of Microarray Data. BMC Bioinformatics 4, 54 (2003)
19. Ambroise, C., McLachlan G.: Selection bias in gene extraction on the basis of microarray gene-expression data. P. Natl. Acad. Sci. USA 99, 6562–6566 (2002)

Fig. 6. Error rates as a function of number of variables selected by the three RFE methods for the Satimage dataset.



20. F. Biasioli, F. Gasperi, E. Aprea, D. Mott, E. Boscaini, D. Mayr and T.D. Märk, J. Agr. Food Chem. 51, 7227 (2003).

21. F. Biasioli, F. Gasperi, E. Aprea, I. Endrizzi, V. Framondino, F. Marini, D. Mott and T.D. Märk, Food Qual. Prefer. 173, 63 (2006).

22. Asuncion, A., Newman, D.: UCI machine learning repository (2007), http://www.ics.uci.edu/ mlearn/MLRepository.html