ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-989

Exploiting the Accumulated Evidence for Gene Selection in Microarray Gene Expression Data

Gabriel Prat-Masramon and **Lluís A. Belanche-Muñoz** [1]

Abstract. Feature subset selection (FSS) methods play an important role for cancer classification using microarray gene expression data. In this scenario, it is extremely important to select genes by taking into account the possible interactions with other gene subsets. This paper shows that, by accumulating the evidence in favour (or against) each gene along a search process, the obtained gene subsets may constitute better solutions, either in terms of size or in predictive accuracy, or in both, at a negligible overhead in computational cost.

1 INTRODUCTION

FSS methods can play an important role in these tasks, since they are characterized by a large number of features (the *genes*) and a few observations, making the modeling a non-trivial undertaking. The selection of a new feature (to be removed/added from/to the current set) involves the evaluation of many models. Only the *best* such evaluation is considered for selecting which feature should be removed/added. Yet there is valuable information in the *discarded* evaluations: when an inducer builds a model using a given feature subset, no indication is given on which feature is the most recent addition (or deletion). Since the most difficult part is to evaluate the *interactions* between features, the *accumulated* evaluation of a feature in diverse *contexts* should account for many of these interactions and ultimately provide with a more informed estimation of usefulness for the chosen inducer. The different contexts of a feature x are given by all those subsets which are being evaluated along the search process, either containing or *not* containing x. The idea can be applied to any sequential search algorithm and any inducer, at a negligible extra cost. We present preliminary experimental results showing good performance in a suite of benchmark *microarray* problems.

2 ACCUMULATED EVIDENCE IN FSS

It is common to see FSS in a set Y of size n as a *search problem* where the search space is $\mathcal{P}(Y)$ [1]. In this setting, the problem is to find an optimal subset $X^* \in \mathcal{P}(Y)$ which maximizes a given objective function $J : \mathcal{P}(Y) \to [0, 1]$. In the literature, several suboptimal algorithms have been proposed for doing this, by combining forward and backward steps. The objective function J may be inducer-independent (as in filter methods) or may be the same inducer being used to solve the task (as in wrapper methods). In either case, we will refer to $J_{\mathcal{L}}(X)$ as the *usefulness* of $X \subseteq Y$ estimated using $\mathcal{L}$ (either filter or wrapper). Since the evaluation of $\mathcal{L}(X)$ in a sample varies depending on the resampling method used, we prefer to use the notation $J_{\mathcal{L}}(X)$. Let $Y_x = \{X \in \mathcal{P}(Y) | x \in X\}$ be the set of all feature subsets of the initial set that contain a certain feature x. Given $\mathcal{L}$ define, for a given feature $x \in Y$, the *relevance* of x as:

$$R_{\mathcal{L}}(x) = \frac{1}{2^{n-1}} \left(\sum_{X \notin Y_x} J_{\mathcal{L}}(X \cup \{x\}) - J_{\mathcal{L}}(X) \right) = \mathcal{L}_x^+ - \mathcal{L}_x^-,$$

where $\mathcal{L}_x^+ = \frac{1}{2^{n-1}} \sum_{X \in Y_x} J_{\mathcal{L}}(X), \qquad \mathcal{L}_x^- = \frac{1}{2^{n-1}} \sum_{X \notin Y_x} J_{\mathcal{L}}(X).$

Let X_k denote the current set, where $|X_k| = k$, for notational simplicity (thus $X_0 = \emptyset$ and $X_n = Y$); let X_{n-k} be the set of features not in X_k, i.e. $X_{n-k} = Y \setminus X_k$. In a classical *backward step* the search algorithm evaluates a feature x for possible exclusion from X_{n-k} in such a way that the set $\left\{ J_{\mathcal{L}}(X_{n-k} \setminus \{x\}) \,|\, x \in X_{n-k} \right\}$ is computed and the feature $x' = \arg\max_{x \in X_{n-k}} J_{\mathcal{L}}(X_{n-k} \setminus \{x\})$ is selected for removal, but the information $\left\{ J_{\mathcal{L}}(X_{n-k} \setminus \{x\}) | x \in X_{n-k}, x \neq x' \right\}$ is readily discarded. Yet sometime in the future these individual features x (and eventually x' itself) will be considered again for inclusion/exclusion in/from the current set in other forward or backward steps, respectively. Now let $P_{\mathcal{L}}$ denote the set of feature subsets that the search algorithm has evaluated so far (implying a call to $J_{\mathcal{L}}$). Let $P_{\mathcal{L}|x} = \{X \in P_{\mathcal{L}} \mid x \in X\}$. For every $x \in Y$, define the accumulated evaluations (or simply *accumulators*) as:

$$\hat{\mathcal{L}}_x^+ = \frac{1}{|P_{\mathcal{L}|x}|} \sum_{X \in P_{\mathcal{L}|x}} J_{\mathcal{L}}(X); \;\; \hat{\mathcal{L}}_x^- = \frac{1}{|P_{\mathcal{L}} \setminus P_{\mathcal{L}|x}|} \sum_{X \notin P_{\mathcal{L}|x}} J_{\mathcal{L}}(X)$$

which are approximations to $\mathcal{L}_x^+$ and $\mathcal{L}_x^-$, respectively. These two values depend on the search algorithm, which determines the strategy to traverse the search space. Consider $\hat{R}_{\mathcal{L}}(x) = \lambda/2(\hat{\mathcal{L}}_x^+ - \hat{\mathcal{L}}_x^- + 1) + (1 - \lambda)\hat{J}_{\mathcal{L}}(x), \lambda \in [0, 1]$, where $\hat{J}_{\mathcal{L}}(x) = J_{\mathcal{L}}(X \setminus \{x\})$ in a backward step (effect of removing x) and $\hat{J}_{\mathcal{L}}(x) = J_{\mathcal{L}}(X \cup \{x\})$ in a forward step (effect of adding x) and λ is a free parameter. By setting $\lambda = 0$, conventional forward and backward steps are recovered. Otherwise, the *search history* makes an influence on the search itself, conditioning the selection of features. In this case, only a $1 - \lambda$ fraction of the importance is assigned to the current subset evaluation.

Example. Consider the following feature subset mask for a current feature subset $X \subseteq Y$ where the i-th index is 1 when $x_i \in X$ and 0 otherwise: `10010010001010100101`, signaling the presence of features $1, 4, 7$, etc. An evaluation $J_{\mathcal{L}}(X)$ of this subset is indeed expressing how good is to have the first feature but not the second or the third, also how good is to have the seventh feature but not the one before the last, and so forth. This is the reason why *all* the features in Y have their accumulators (known evaluations) updated every time.

We illustrate our approach on the popular SBG or backward search algorithm and give a practical implementation of the previous ideas

[1] Technical University of Catalonia, Barcelona, Spain, emails: gprat@lsi.upc.edu, belanche@lsi.upc.edu

for it (**Algorithm 1**). The initialization of the accumulated relevances is 0 for all $x \in Y$. Note that the results are first accumulated and then used; for this reason, even in the *first* algorithmic step (first discarded feature) the behavior of both algorithms may start to diverge. At the end of the FSS process, $n_x^+(n_x^-)$ will be the number of times that a feature subset (not) containing x has been evaluated. Note that the computation is done at a negligible overhead in cost; this is due to the fact that the inducer is called *exactly* the same number of times.

Algorithm 1 SBG$^+$ (inducer $\mathcal{L}$, set Y, $\lambda \in [0,1]$)

$X_n \leftarrow Y$; $k \leftarrow 0$; $\forall x \in Y : \hat{L}_x^+ \leftarrow \hat{L}_x^- \leftarrow 0$; $n_x^+ \leftarrow n_x^- \leftarrow 0$
repeat
  Compute the set $\left\{ J_{\mathcal{L}}(X_{n-k} \setminus \{x\}) \mid x \in X_{n-k} \right\}$
  $\forall x \in Y$: **if** $x \in X_{n-k}$
    then $n_x^+ \leftarrow n_x^+ + 1$; $\hat{L}_x^+ \leftarrow \hat{L}_x^+ + \sum_{y \in X_{n-k} \setminus \{x\}} J_{\mathcal{L}}(X_{n-k} \setminus \{y\})$
    else $n_x^- \leftarrow n_x^- + 1$; $\hat{L}_x^- \leftarrow \hat{L}_x^- + J_{\mathcal{L}}(X_{n-k} \setminus \{x\})$
  $x' \leftarrow \arg\max_{x \in X_{n-k}} \left\{ \lambda/2(\hat{L}_x^+/n_x^+ - \hat{L}_x^-/n_x^- + 1) + (1-\lambda) J_{\mathcal{L}}(X_{n-k} \setminus \{x\}) \right\}$
  $k \leftarrow k+1$; $X_{n-k} \leftarrow X_{n-k} \setminus \{x'\}$
until $k = n$
return $\arg\max_{k=1 \div n} J_{\mathcal{L}}(X_k)$ {Selected subset}

3 EXPERIMENTAL WORK

Experimental work is now presented in order to assess the described modifications, comparing the original algorithm (SBG) and its accumulated version (SBG$^+$). Each full experiment consists of an *outer* loop of 5x2-cross-validation (5x2cv) for model selection, as proposed by several authors [2]. This procedure performs 5 repetitions of a 2-fold cross-validation. It keeps half of the examples out of the FSS process and uses them as a *test set* to evaluate the final quality of the selected features. The selected inducers are the *nearest-neighbor* technique (1NN), *linear discriminant analysis* (LDA) and the *Support Vector Machine* with radial kernel (SVM). The evaluation of these inducers is resampled in a second (*inner*) 5x2cv loop for a more informed estimation of usefulness. In all cases, stratification is used to keep the same proportion of class labels across the partitioned sets. After some preliminary experiments, we set $\lambda = \frac{2}{3}$. It is very important to mention that there is no stopping criterion in the algorithms: the two backward methods run until all the features have been removed. Then the *best* subset in the obtained sequence of subsets is returned. This setting avoids the specification of an a priori size for the solution. It also eliminates the possibility that the accumulated algorithm performs differently simply because it merely influences the stopping point. Once the best feature subset is found (a different one in every outer loop), this subset is evaluated in the corresponding test set. The final error is the mean of these 10 values. We work with five public-domain microarray gene expression data sets: *Colon Tumor* (**CT**) [3], *Leukemia* (**LK**) [4], *Lung Cancer* (**LC**) [5], *Prostate Cancer* (**PC**) [6] and *Breast Cancer* (**BC**) [7]. We made a preliminary selection of 200 genes on the basis of the ratio of their between-groups to within-groups sum of squares, to make a wrapper approach computationally feasible [8]. The results are displayed in Table 1: the (cross-validated) average test error and the (cross-validated) average size of the final selected subsets. The accumulated version outperforms the standard version (though in general by a modest margin) in all cases. This is a remarkable result, given the differences among the problems and among the inducers; SBG$^+$ finds in general solutions of lower size than SBG does, sometimes by a substantial amount. Since there is no stopping condition, our explanation is that the standard backward version is *greedier* than the accumulated one. By the (early) inclusion of some features that are not as good as they look in that moment, SBG is driven toward worse local minima of the error function as compared to SBG$^+$.

Table 1. Average test errors (in %) / Average gene subset sizes.

	1NN		LDA		SVM	
	SBG$^+$	SBG	SBG$^+$	SBG	SBG$^+$	SBG
CT	18.1/37.4	20.0/73.8	19.0/70.5	22.2/79.2	18.1/15.5	18.7/14.2
LK	8.1/7.2	10.9/28.3	16.7/30.0	17.7/32.5	7.8/6.1	9.2/37.2
LC	3.3/17.4	3.4/20.0	2.7/4.1	3.4/13.4	3.4/4.5	3.5/8.8
PC	14.0/18.3	15.5/19.3	24.8/23.5	26.4/44.3	21.9/12.9	22.0/8.1
BC	26.2/60.2	29.3/34.2	27.4/22.4	36.7/52.6	23.7/13.0	25.6/17.5

Comparison to other results in the literature using the same data sets is a delicate undertaking, especially concerning resampling techniques. We have found that many times there are no true test sets: feature subsets or model parameters (or both) are optimized by means of one or several runs of cross-validation. This procedure is dangerous given that, although test observations have not been used to create the models, they have been used to decide upon competing ones. Moreover, in our experiments, SVM parameters were not optimized beyond educated guesses, so there is still room for improvement. The interested reader can consult the results reported in [9, 10, 11].

4 CONCLUSIONS

By making algorithms accumulate all the "log of merit" of the features and assigning less importance to the current evaluation, our experimental results indicate a general improvement in performance, without any additional effort. It is relevant to point out that the presented algorithmic modification may be of little help if an algorithm has many opportunities to rectify its decisions. However, even in this case, the forward or backward steps will be more informed, possibly making the search algorithm deliver better solutions at earlier stages.

REFERENCES

[1] P. Langley, 'Selection of relevant features in machine learning', in *Proceedings of the AAAI Fall Symposium on Relevance*, (1994).
[2] E. Alpaydin, 'Combined 5x2cv f-test for comparing supervised classification learning algorithms', *Neural computation*, **11**, 1885–92, (1999).
[3] U. Alon et al., 'Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays', in *Proc. Natl. Acad. Sci. USA*, **96**, 6745-6750, (1999).
[4] T. Golub et al., 'Molecular classification of cancer: class discovery and prediction by gene expression monitor.', *Science*, **286**, 531–7, (1999).
[5] G. Gordon et al., 'Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma'. *Cancer Research*, **62**, 4963–4967, (2002).
[6] D. Singh et al., 'Gene expression correlates of clinical prostate cancer behavior'. *Cancer Cell*, **1**, 203–209, (2002).
[7] L. Veer et al., 'Gene expression profiling predicts clinical outcome of breast cancer'. *Nature*, **415**, 530–536, (2002).
[8] S. Dudoit, J. Fridlyand, and T. Speed, 'Comparison of discrimination methods for the classification of tumors using gene expression data', *Journal of the Amer. Stat. Assoc.*, **97**(457), 77–87, (2002).
[9] L. Wang et al., 'Hybrid huberized SVMs for microarray classification and gene selection'. *Bioinformatics* **24**(3), 412–419, (2008).
[10] H.L. Bu et al., 'Reducing error of tumor classification by using dimension reduction with feature selection'. Intl. Symp. on Optim. and Sys. Biol., 232–241, (2007).
[11] J.H. Hong, S.B. Cho, 'Cancer classification with incremental gene selection based on dna microarray data'. *IEEE/ACM Trans. on Comp. Biol. and Bioinf.*, 70–74 (2008).