HAL open science

Bayesian Learning with Local Support Vector Machines for Cancer Classification with Gene Expression Data

Elena Marchiori, Michèle Sebag

▶ To cite this version:

Elena Marchiori, Michèle Sebag. Bayesian Learning with Local Support Vector Machines for Cancer Classification with Gene Expression Data. EvoBIO, Apr 2005, Lausanne, Switzerland. pp.74-83. inria-00177278

HAL Id: inria-00177278

https://inria.hal.science/inria-00177278v1

Submitted on 7 Oct 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Bayesian learning with local support vector machines for cancer classification with gene expression data

Elena Marchiori[1] and Michèle Sebag[2]

[1] Department of Computer Science
Vrije Universiteit Amsterdam, The Netherlands
elena@cs.vu.nl
[2] Laboratoire de Recherche en Informatique, CNRS-INRIA
Universit Paris-Sud Orsay, France
sebag@lri.fr

Abstract. This paper describes a novel method for improving classification of support vector machines (SVM) with recursive feature selection (SVM-RFE) when applied to cancer classification with gene expression data. The method employs pairs of support vectors of a linear SVM-RFE classifier for generating a sequence of new SVM classifiers, called local support classifiers. This sequence is used in two Bayesian learning techniques: as ensemble of classifiers in Optimal Bayes, and as attributes in Naive Bayes. The resulting classifiers are applied to four publically available gene expression datasets from leukemia, ovarian, lymphoma, and colon cancer data, respectively. The results indicate that the proposed approach improves significantly the predictive performance of the baseline SVM classifier, its stability and robustness, with satisfactory results on all datasets. In particular, perfect classification is achieved on the leukemia and ovarian cancer datasets.

1 Introduction

This paper deals with tumor classification with gene expression data. Microarray technology provides a tool for estimating expression of thousands of genes simultaneously. To this end, DNA arrays are used, consisting of a large number of DNA molecules spotted in a systematic order on a solid substrate. Depending on the size of each DNA spot on the array, DNA arrays are called microarrays when the diameter of DNA spot is less than 250 microns, and macroarrays when the diameter is bigger than 300 microns. DNA microarrays contain thousands of individual DNA sequences printed in a high density array on a glass microscope slide using a robotic instrument. The relative abundance of these spotted DNA sequences in the two DNA and RNA samples may be assessed by monitoring the differential hybridization of the two samples to the sequences on the array. For mRNA samples, the two samples are reverse-transcribed into cDNA, labeled using different fluorescent dyes mixed (red-fluorescent dye Cy5 and green-fluorescent dye Cy3). After these samples are hybridized with the arrayed DNA probes, the slides are

imaged using scanner that makes fluorescence measurements for each dye. The log ratio between the two intensities of each dye is used as the gene expression data (cf. [11]) $expression(gene) = log_2(int(Cy5)/int(Cy3))$, were $int(Cy5)$ and $int(Cy3)$ are the intensities of the two fluorescent dyes.

Four main machine learning tasks are used to analyze DNA microarray data: clustering, e.g. for identifying tumor subtypes, classification, e.g. for tumor diagnostic, feature selection for potential tumor biomarker identification, and gene regulatory network modeling. This paper deals with classification.

Many machine learning techniques have been applied to classify gene expression data, including Fisher linear discriminat analysis [10], k-nearest neighbour [18], decision tree, multi-layer perceptron [17, 25], support vector machine (SVM) [6, 13, 15, 21], boosting and ensemble methods [14, 7, 23, 3, 9]. A recent comparison of classification and feature selection algorithms applied to tumor classification can be found in [7, 23].

This paper introduces a method that improves the predictive performance of a linear SVM with Recursive Feature Elimination (SVM-RFE) [15] on four gene expression datasets. The method is motivated by previous work on aggregration of classifiers [4, 5], where it is shown that gains in accuracy can be obtained by aggregrating classifiers built from perturbed versions of the train set, for instance using bootstrapping. Application of aggregration of classifiers to microarray data is described e.g. in [10, 7, 3, 9].

In this paper a novel approach is proposed, for generating a sequence of classifiers from the support vectors of a baseline linear SVM-RFE classifier. Each pair of support vectors of the same class are used to generate an element of the sequence, called local support classifier (*lsc*). Such classifier is obtained by training SVM-RFE on data consisting of the two selected support vectors and all the support vectors of the other class.

The sequence of *lsc*'s provides an approximate description of the data distribution by means of a set of linear decision functions, one for each region of the input space in a small neighbourhoods of two support vectors having equal class label.

We propose to use this sequence of classifiers in Bayesian learning (cf. [20]). The first technique applies Naive Bayes to the transformed data, where an example is mapped into the binary vector of its classification values. The resulting classifier is called Naive Bayes Local Support Classifier (NB-LSC). The second technique applies Optimal Bayes to the sequence of *lsc*'s classifiers. The resulting classifier is called Optimal Bayes Local Support Classifier (OB-LSC).

The two classifiers are applied to four publically available datasets for cancer classification with gene expression. The results show a significant improvement in predictive performance of OB-LSC over the baseline linear SVM-RFE classifier, and a gain in stability. In particular, on the leukemia and ovarian cancer datasets perfect classification is obtained, and on the other datasets performance comparable to the best published results we are aware of.

The rest of the paper is organized as follows. The next two sections describe the baseline and new methods. Sections 4 contains a short description of the

data. Section 5 reports results of experiments and discuss them. Finally, the paper ends with conclusive considerations on research issues to be tackled in future work.

2 Support Vector Machines

This section describes in brief SVM-RFE, the local support classifier construction procedure, and the integration of the resulting classifier sequence in Naive Bayes and Optimal Bayes classification.

2.1 SVM

In linear SVM binary classification [24, 8] patterns of two classes are linearly separated by means of a maximum margin hyperplane, that is, the hyperplane that maximizes the sum of the distances between the hyperplane and its closest points of each of the two classes (the margin). When the classes are not linearly separable, a variant of SVM, called soft-margin SVM, is used. This SVM variant penalizes misclassification errors and employs a parameter (the soft-margin constant C) to control the cost of misclassification.

Training a linear SVM classifier amounts to solving the following constrained optimization problem:

$$min_{\mathbf{w},b,\xi_k} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m}\xi_i \quad \text{s.t.} \quad \mathbf{w}\cdot\mathbf{x}_i + b \geq 1 - \xi_i$$

with one constraint for each training example $\mathbf{x}_i$. Usually the dual form of the optimization problem is solved:

$$min_{\alpha_i} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \mathbf{x}_i\cdot\mathbf{x}_j - \sum_{i=1}^{m}\alpha_i$$

such that $0 \leq \alpha_i \leq C$, $\sum_{i=1}^{m}\alpha_i y_i = 0$. SVM requires $O(m^2)$ storage and $O(m^3)$ to solve.

The resulting decision function $f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + b$ has weight vector $\mathbf{w} = \sum_{k=1}^{m}\alpha_k y_k \mathbf{x_k}$. Examples $\mathbf{x}_i$ for which $\alpha_i > 0$ are called *support vectors*, since they define uniquely the maximum margin hyperplane.

Maximizing the margin allows one to minimize bounds on generalization error. Because the size of the margin does not depend on the data dimension, SVM are robust with respect to data with high input dimension. However, SVM are sensitive to the presence of (potential) outliers, (cf. [15] for an illustrative example) due to the regularization term for penalizing misclassification (which depends on the choice of C).

2.2 SVM-RFE

The weights w_i provide information about feature relevance, where bigger weight size implies higher feature relevance. In this paper feature x_i is scored by means of the absolute value of w_i. Other scoring functions based on weight features are possible, like, e.g., w_i^2, which is used in the original SVM-RFE algorithm [15].

SVM-RFE is an iterative algorithm. Each iteration consists of the following two steps. First feature weights, obtained by training a linear SVM on the training set, are used in a scoring function for ranking features as described above. Next, the feature with minimum rank is removed from the data. In this way, a chain of feature subsets of decreasing size is obtained.

In the original SVM-RFE algorithm one feature is discarded at each iteration. Other choices are suggested in [15], where at each iteration features with rank lower than a user-given theshold are removed. In general, the threshold influences the results of SVM-RFE [15]. In this paper we use a simple instance of SVM-RFE where the user specifies the number of features to be selected, 70% of the actual number of features are initially removed, and then 50% at each further iteration. These values are chosen after cross-validation applied to the training set.

3 Local Support Classifiers

We propose to describe the distribution of the two classes by means of a sequence of classifiers, generated from pairs of support vectors ofSVM-RFE. Each of these classifiers, called local support classifier (*lsc*), is obtained using data generated from two support vectors of the same class, and all support vectors of the other class. In this way, each classifier uses only a local region near the two selected support vectors when separating the two classes. Each classifier generated from two (distinct) support vectors of the same class provides an approximate description of the distribution of the other class *given* the two selected support vectors.

Before describing the procedure for constructing *lsc*'s, some notation used throughout the paper is introduced.

- D denotes the training set,
- c denotes the classifier obtained by training a linear SVM on D,
- S_p and S_n denote the set of positive and negative support vectors of c, respectively,
- $Pair_p$ and $Pair_n$ denote the set of pairs of distinct elements of S_p and S_n, respectively.

The following procedure, called `LSC`, takes as input one $(\mathbf{s}, \mathbf{s}')$ in $Pair_p$ and outputs a linear SVM classifier $C_{s,s'}$ by means of the following two steps.

1. Let $Xp = \{\mathbf{s}, \mathbf{s}'\}$. Assign positive class label to these examples.
2. Let $C_{s,s'}$ be the classifier obtained by training a linear SVM on data $Xp \cup Sn$.

An analogous procedure is applied to generate $C_{s,s'}$ from pairs $(\mathbf{s}, \mathbf{s}')$ in $Pair_n$.

When applied to all pairs of support vectors in $Pair_p, Pair_n$, `LSC` produces a sequence of *lsc*'s. Such sequence of classifiers induces a data transformation, called seq_D, which maps example $\mathbf{x}$ in the sequence $seq_D(\mathbf{x})$ of class values $C_{s,s'}(\mathbf{x})$, with $(\mathbf{s}, \mathbf{s}')$ in $Pair_p \cup Pair_n$.

The construction of the sequence of*lsc*'s requires computation that grows quadratically with the number of support vectors. However, this is not a severe problem, since the number of examples, hence of support vectors, is small for this type of data. Furthermore, `LSC` is applied to each pair of support vectors independently, hence can be executed in parallel.

3.1 Naive Bayes and Optimal Bayes Classification

Naive Bayes (NB) is based on the principle of assigning to a new example the most probable target value, given the attribute values of the example. In order to apply directly NB to the original gene expression data, gene values need to be discretized, since NB assumes discrete-valued attributes. Examples transformed using seq_D contain binary attributes, hence discretization is not necessary.

Let $\mathbf{x}$ be a new example. Suppose $seq_D(\mathbf{x}) = (x_1, \ldots, x_N)$.

First, the prior probabilities p_y of the two target values are estimated by means of the frequency of positive and negative examples occurring in the train set D, respectively. Next, for each attribute value x_i, the probability $P(x_i \mid y)$ of x_i given target value y is estimated as the frequency with which x_i occurs as value of i-th attribute among the examples of D with class value y. Finally, the classification of $\mathbf{x}$ is computed as the y that maximizes the product

$$p_y \prod_{i=1}^{N} P(x_i \mid y).$$

The resulting classifier is denoted by NB-LSC.

Optimal Bayes (OB) classifier is based on the principle of maximizing the probability that a new example is classified correctly, given the available data, classifiers, and prior probabilities over the classifiers.

OB maps example $\mathbf{x}$ to the class that maximizes the weighted sum

$$\sum_{C_{s,s'}} w_{s,s'} I(C_{s,s'}(\mathbf{x}) = y),$$

where $w_{s,s'}$ is the accuracy of $C_{s,s'}$ over D, and I is the indicator function, which returns 1 if the test contained in its argument is satisfied and 0 otherwise. The resulting classifier is denoted by OB-LSC.

4 Datasets

There are several microarray datasets from published cancer gene expression studies, including leukemia cancer dataset, colon cancer dataset, lymphoma

dataset, breast cancer dataset, NCI60 dataset, ovarian cancer, and prostate dataset. Among them four datasets are used in this paper, available e.g. at `http://sdmc.lit.org.sg/GEDatasets/Datasets.html`. The first and third dataset contain samples from two variants of the same disease, the second and last dataset consist of tumor and normal samples of the same tissue. Table 1 shows input dimension and class sizes of the datasets. The following short description of the datasets is partly based on [7].

Table 1. Datasets description

Name	Tot	Positive	Negative	Genes
Colon	62	22	40	2000
Leukemia	72	25	47	7129
Lymphoma	58	26	32	7129
Ovarian	54	30	24	1536

4.1 Leukemia

The Leukemia dataset consists of 72 samples: 25 samples of acute myeloid leukemia (AML) and 47 samples of acute lymphoblastic leukemia (ALL). The source of the gene expression measurements is taken from 63 bone marrow samples and 9 peripheral blood samples. Gene expression levels in these 72 samples are measured using high density oligonucleotide microarrays [2]. Each sample contains 7129 gene expression levels.

4.2 Colon

The Colon dataset consists of 62 samples of colon epithelial cells taken from colon-cancer patients. Each sample contains 2000 gene expression levels. Although the original data consists of 6000 gene expression levels, 4000 out of 6000 were removed based on the confidence in the measured expression levels. 40 of 62 samples are colon cancer samples and the remaining are normal samples. Each sample is taken from tumors and normal healthy parts of the colons of the same patients and measured using high density oligonucleotide arrays [1].

4.3 Lymphoma

B cell diffuse large cell lymphoma (B-DLCL) is a heterogeneous group of tumors, based on significant variations in morphology, clinical presentation, and response to treatment. Gene expression profiling has revealed two distinct tumor subtypes of B-DLCL: germinal center B cell-like DLCL and activated B cell-like DLCL [19]. Lymphoma dataset consists of 24 samples of germinal center B-like and 23 samples of activated B-like.

4.4 Ovarian

Ovarian tissue from 30 patients with cancer and 23 without cancer were analyzed for mRNA expression using glass arrays spotted for 1536 gene clones. Attribute i of patient j is the measure of the mRNA expression of the i-th gene in that tissue sample, relative to control tissue, with a common control employed for all experiments [22].

5 Numerical Experiments

The two classifiers NB-LSC and OB-LSC, described in Section 3.1, are applied to the four gene expression datasets the baseline SVM-RFE algorithm. In all experiments the same value of the SVM parameter $C = 10$ is used, while the number of selected genes was set to 30 for the lymphoma dataset and 50 for all other datasets. These values are chosen by means of cross-validation applied to the training set.

Because of the small size of the datasets, Leave One Out Cross Validation (LOOCV) is used to estimate the predictive performance of the algorithms [12].

Table 2. Results of LOOCV: average sensitivity, specificity and accuracy (with standard deviation between brackets).

Method	Dataset	Sensitivity	Specificity	Accuracy
SVM-RFE	Colon	0.90 (0.3038)	1.00 (0.00)	0.9355 (0.2477)
NB-LSC		0.75 (0.4385)	1.00 (0.00)	0.8387 (0.3708)
OB-LSC		0.90 (0.3038)	1.00 (0.00)	0.9355 (0.2477)
SVM-RFE	Leukemia	0.96 (0.20)	1.00 (0.00)	0.9861 (0.1179)
NB-LSC		1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
OB-LSC		1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
SVM-RFE	Ovarian	0.7000 (0.4661)	0.9583 (0.2041)	0.8148 (0.3921)
NB-LSC		1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
OB-LSC		1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
SVM-RFE	Lymphoma	0.6923 (0.4707)	0.6562 (0.4826)	0.6724 (0.4734)
NB-LSC		1.00 (0.00)	0.6562 (0.4826)	0.8103 (0.3955)
OB-LSC		1.00 (0.00)	0.8750 (0.3360)	0.9310 (0.2556)

Table 2 reports results of LOOCV. They indicate a statistically significant improvement of OB-LSC over the baseline SVM-RFE classifier, and a gain in stability, indicated by lower standard deviation values. In particular, on the ovarian and leukemia datasets both NB-LSC and OB-LSC achieve perfect classification.

Moreover, while the performance of SVM on the Lymphoma dataset is rather scare (possibly due to the fact that we did not scale the data), OB-LSC obtains results competitive to the best results known (see Table 3).

Table 3. Comparison of results with best average accuracy reported in previous papers on tumor classification. The type of classifiers considered in the paper are given between brackets. An entry '-' means that the corresponding dataset has not been considered.

	Colon	Leukemia	Lymphoma	Ovarian
Furey et al (SVM)	0.90	0.94	-	-
Li et al 00 (Logistic regression)	-	0. 94	-	-
Li et al 01 (KNN)	0.94	-	0.94	-
Ben-Dor et al (Quadratic SVM, 1NN, AdaBoost)	0.81	0.96	-	-
Dudoit et al (1NN, LDA, BoostCART)	-	0.95	0.95	-
Nguyen et al (Logistic discriminant, QDA)	0.94	0.96	0.98	-
Cho et al (Ensemble SVM, KNN)	0.94	0.97	0.96	-
Liu et al 04 (Ensemble NN)	0.91	-	-	-
Dettling et al 03 (Boosting)	0.85	-	-	-
OB-LSC	0.94	1.00	0.93	1.00

Table 3 reports results of OB-LSC and the best result among those contained nine papers on tumor classification and feature selection using different machine learning methods [7]. Note that results reported in this table have been obtained using different cross-validation methods, mainly by repeated random partitioning the data into train and test set using 70 and 30 % of the data, respectively. Because the resulting estimate of predictive performance may be more biased than the one of LOOCV [12], those results give only an indication for comparing the methods. Only the results on the colon dataset from Liu et al 04 and Dettling et al 03 [9, 3] are obtained using LOOCV. The methods proposed in these latter papers use boosting and bagging, respectively. The results they obtain seem comparable to OB-LSC.

The results indicate that OB-LSC is competitive with most recent classification techniques for this task, including non-linear methods.

6 Conclusion

This paper introduced an approach that improves predictive performance and stability of linear SVM for tumor classification with gene expression data on four gene expression datasets.

We conclude with two considerations on research issues still to be addressed. Our approach is at this stage still an heuristic, and needs further experimental and theoretical analysis. In particular, we intend to analyze how performance is related to the number of support vectors chosen to generate *lsc*'s. Moreover, we intend to investigate the use of this approach for feature selection, for instance whether the generated *lsc*'s can be used for ensemble feature ranking [16].

References

1. U. Alon, N. Barkai, and D.A. Notterman et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *PNAS*, 96:6745–50, 1999.
2. A. Ben-Dor, L. Bruhn, and N. Friedman et al. Tissue classification with gene expression profiles. *Journal of Computational Biology*, 7:559–584, 2000.
3. B.Liu, Q. Cui, and T.Jiang et al. A combinational feature selection and ensemble neural network method for classification of gene expression data. *BMC Bioinformatics*, 5(136), 2004.
4. L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
5. L. Breiman. Arcing classifiers. *The Annals of Statistics*, 26(3):801–849, 1998.
6. M. Brown, W. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. Furey, M. Jr, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. In *Proc. Natl. Acad. Sci.*, volume 97, pages 262–267, 2000.
7. S. Cho and H. Won. Machine learning in DNA microarray analysis for cancer classification. In *Proceedings of the First Asia-Pacific bioinformatics conference on Bioinformatics 2003*, pages 189 – 198. Australian Computer Society, 2003.
8. N. Cristianini and J. Shawe-Taylor. *Support Vector machines*. Cambridge Press, 2000.
9. M. Dettling and P. Buhlmann. Boosting for tumor classification with gene expression data. *BMC Bioinformatics*, 19:1061–1069, 2003.
10. S. Dudoit, J. Fridlyand, and T.P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457), 2002.
11. M.B. Eisen and P.O. Brown. Dna arrays for analysis of gene expression. *Methods Enzymbol*, (303):179–205, 1999.
12. T. Evgeniou, M. Pontil, and A. Elisseeff. Leave one out error, stability, and generalization of voting combinations of classifiers. *Mach. Learn.*, 55(1):71–97, 2004.
13. T. S. Furey, N. Christianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Hauessler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.
14. T. R. Golub, D. K. Slonim, and P. Tamayo et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–7, 1999.
15. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46(1-3):389–422, 2002.
16. K. Jong, J. Mary, A. Cornuejols, E. Marchiori, and M. Sebag. Ensemble feature ranking. In *Proceedings Eur. Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECML/PKDD 2004*, 2004.
17. J. Khan, J. S. Wei, M. Ringner, and et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7:673–679, 2001.
18. Li, Darden, Weinberg, Levine, and Pedersen. Gene assessment and sample classification for gene expression data using a genetic algorithm/k-nearest neighbor method. *Combinatorial Chemistry and High Throughput Screening*, 4(8):727–739, 2001.
19. I. Lossos, A. Alizadeh, and M. Eisen et al. Ongoing immunoglobulin somatic mutation in germinal center b cell-like but not in activated b cell-like diffuse large

cell lymphomas. In *Proc Natl Acad Sci U S A*, volume 97, pages 10209–10213, 2000.

20. Tom Mitchell. *Machine Learning*. McGraw Hill, 1997.
21. W.S. Noble. Support vector machine applications in computational biology. In B. Schoelkopf, K. Tsuda, and J.-P. Vert, editors, *Kernel Methods in Computational Biology*, pages 71–92. MIT Press, 2004.
22. M. Schummer, W. V. Ng, and R. E. Bumgarnerd et al. Comparative hybridization of an array of 21,500 ovarian cdnas for the discovery of genes overexpressed in ovarian carcinomas. *Gene*, 238(2):375–85, 1999.
23. A.C. Tan and D. Gilbert. Ensemble machine learning on gene expression data for cancer classification. *Applied Bioinformatics*, 2((3 Suppl)):75–83, 2003.
24. V.N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.
25. Y. Xu, F. M. Selaru, and J. Yin et al. Artificial neural networks and gene filtering distinguish between global gene expression profiles of barrett's esophagus and esophageal cancer. *Cancer Research*, 62(12):3493 – 3497, 2002.