Available at www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

Signal Processing 83 (2003) 695–712

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

# Efficient selection of feature sets possessing high coefficients of determination based on incremental determinations

Ronaldo F. Hashimoto[a,b], Edward. R. Dougherty[a,c,*], Marcel Brun[a,b], Zheng-Zheng Zhou[d], Michael L. Bittner[e], Jeffrey M. Trent[e]

[a] *Department of Electrical Engineering, Texas A&M University, 3128 TAMU, College Station, TX 77843-3128, USA*
[b] *Departamento de Ciência de Computação, Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, Brazil*
[c] *Department of Pathology, University of Texas M.D. Anderson Cancer Center, Houston, TX 77030, USA*
[d] *NuTec Sciences, Inc., USA*
[e] *National Human Genome Research Institute of the National Institutes of Health, Bethesda, MD 20892, USA*

Received 7 March 2002; received in revised form 27 August 2002

**Abstract**

Feature selection is problematic when the number of potential features is very large. Absent distribution knowledge, to select a best feature set of a certain size requires that all feature sets of that size be examined. This paper considers the question in the context of variable selection for prediction based on the coefficient of determination (CoD). The CoD varies between 0 and 1, and measures the degree to which prediction is improved by using the features relative to prediction in the absence of the features. It examines the following heuristic: if we wish to find feature sets of size $m$ with CoD exceeding $\delta$, what is the effect of only considering a feature set if it contains a subset with CoD exceeding $\lambda < \delta$? This means that if the subsets do not possess sufficiently high CoD, then it is assumed that the feature set itself cannot possess the required CoD. As it stands, the heuristic cannot be applied since one would have to know the CoDs beforehand. It is meaningfully posed by assuming a prior distribution on the CoDs. Then one can pose the question in a Bayesian framework by considering the probability $P(\theta > \delta \,|\, \max\{\theta_1, \theta_2, \ldots, \theta_v\} < \lambda)$, where $\theta$ is the CoD of the feature set and $\theta_1, \theta_2, \ldots, \theta_v$ are the CoDs of the subsets. Such probabilities allow a rigorous analysis of the following decision procedure: the feature set is examined if $\max\{\theta_1, \theta_2, \ldots, \theta_v\} \geqslant \lambda$. Computational saving increases as $\lambda$ increases, but the probability of missing desirable feature sets increases as the increment $\delta - \lambda$ decreases; conversely, computational saving goes down as $\lambda$ decreases, but the probability of missing desirable feature sets decreases as $\delta - \lambda$ increases. The paper considers various loss measures pertaining to omitting feature sets based on the criteria. After specializing the matter to binary features, it considers a simulation model, and then applies the theory in the context of microarray-based genomic CoD analysis. It also provides optimal computational algorithms.


*Keywords:* Coefficient of determination; Feature selection; Gene microarray; Optimal classifier

## 1. Introduction

The feature-selection problem in pattern recognition is to select a subset of $k$ random variables from a set on $n$ random variables that provides an

* Corresponding author. Department of Electrical Engineering, Texas A&M University, 3128 TAMU, College Station, TX 77843-3128, USA. Tel.: +1-409-845-7441; fax: +1-409-845-6259.
*E-mail address:* edward@ee.tamu.edu (E.R. Dougherty).

0165-1684/03/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
doi:10.1016/S0165-1684(02)00468-1

optimal classifier with minimum error among all optimal classifiers for subsets of size $k$. The inherent combinatorial nature of the problem is readily seen from a classic theorem of Cover and Van Campenhout [2]: if $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_r\}$ is the set of possible feature sets formed from the set $\{X_1, X_2, \ldots, X_n\}$ of random variables under the assumption that $i < j$ if $\mathbf{X}_i \subset \mathbf{X}_j$, and if $Y$ is a binary target random variable, then there exists a multivariate distribution of $X_1, X_2, \ldots, X_n, Y$ such that $\varepsilon[\mathbf{X}_1] > \varepsilon[\mathbf{X}_2] > \cdots > \varepsilon[\mathbf{X}_r]$, where $\varepsilon[\mathbf{X}_j]$ is the Bayes error for $\mathbf{X}_j$, given by $\varepsilon[\mathbf{X}_j] = P(Y \neq \psi_0(\mathbf{X}_j))$, where $\psi_0$ is the optimal classifier based on $\mathbf{X}_j$. The requirement that $i < j$ if $\mathbf{X}_i \subset \mathbf{X}_j$ is necessary because the subset condition implies that $\varepsilon[\mathbf{X}_i] \geqslant \varepsilon[\mathbf{X}_j]$. A consequence of the theorem is that all $k$-element subsets must be checked unless there is distributional knowledge that mitigates the search requirement. Various heuristic suboptimal algorithms have been developed to circumvent the full combinatorial search that requires estimating the Bayes error of $C(n,k)$ subsets [3,12,15], and a branch and bound algorithm has been used for efficient selection of the best feature set [10].

We will take an approach that is well-suited to the context in which we are confronting feature selection. There is a very large number $n$ of variables, and we desire very small feature sets ($k = 1, 2, 3, 4$). Moreover, we are not necessarily interested in the best feature set of size $k$, but rather finding good feature sets of size $k$; indeed, our specific concern is finding collections of good feature sets for a family of target random variables. To be precise, we have a set $\mathbf{X}$ of $n$ binary random variables $X_1, X_2, \ldots, X_n$ and we want to find small subsets of the variables to predict other variables in $\mathbf{X}$. The number of variables $n$ often exceeds 500, perhaps significantly so. To check all feature sets would require computing optimal estimators and errors for all subsets up to maximum size $m$, which would require computing $C(n,1) + C(n,2) + \cdots + C(n,m)$ errors if we wish to consider feature sets up to size $m$.

Instead of stating the feature-selection problem in terms of error, we can equivalently state it in terms of the coefficient of determination (CoD), which provides a normalized measure of the degree to which $Y$ can be better predicted using the observations in a feature set than it can be in the presence of no observations. Specifically, for any feature set $\mathbf{X}$, the CoD relative to the target variable $Y$ is defined to be

$$\theta_{\mathbf{X}} = \frac{\varepsilon_\bullet - \varepsilon_{\mathbf{X}}}{\varepsilon_\bullet},$$

where $\varepsilon_\bullet$ is the error of the best estimate for $Y$ in the absence of other observations and $\varepsilon_{\mathbf{X}}$ is the Bayes error for $\mathbf{X}$. The coefficient has historically been used to measure the effect of linear regression [16], and has been recently employed in nonlinear signal processing [5] and for measuring multivariate interaction among genes based on gene expression [5,8], and for constructing probabilistic Boolean networks [13]. It is this last application that has motivated the current analysis. It is evident from the definition that $0 \leqslant \theta_{\mathbf{X}} \leqslant 1$ and $\mathbf{X} \subseteq \mathbf{Z}$ implies $\theta_{\mathbf{X}} \leqslant \theta_{\mathbf{Z}}$. In terms of the coefficient, the feature-selection problem is to find a subset of $k$ random variables possessing the highest coefficient among all subsets of size $k$.

The particular application we have in mind, and will discuss in detail following the development of the methodology, involves the prediction whether a gene is up- or down-regulated based on the up- or down-regulation of other genes using data from gene-expression microarrays [8,9]. The full search for this problem is currently done using massively parallel hardware, practically halts at $m = 3$ for about $n = 600$ genes, and takes 2 weeks using over 100 CPU's if all 600 targets are to be considered [14]. Since gene-expression data is severely limited with current genomic technology, for statistical reasons there is usually little to be gained by going beyond three predictors; nonetheless, it is of great practical benefit to reduce the computation so that two-gene prediction can be accomplished easily on a workstation, three-gene prediction can be accomplished for a small subset of targets on a workstation, and prediction for full target sets can be accomplished significantly faster on a parallel system. Moreover, with the rapid evolution of microarray technology, sample sizes are increasing and four-predictor sets may soon exhibit statistically significant gains over three-predictor sets sufficiently often to be of serious interest.

## 2. Distributional selection of variables

A simple approach to avoiding the full combinatorial search for the optimal two-feature set is to skip

the computation of the CoD for two variables relative to a target variable if the individual CoDs of the variables relative to the target are both small. For a fixed target $Y$, if $\theta_i$, $\theta_{i,j}$ and $\theta_{i,j,k}$ denote the CoD for $\{X_i\}$, $\{X_i,X_j\}$ and $\{X_i,X_j,X_k\}$, respectively, then a suboptimal algorithm is defined by saying that if both $\theta_i$ and $\theta_j$ are very small, then we ignore $\theta_{i,j}$, thereby saving its computation. The problem is it may be that $\theta_i=\theta_j=0$, while $\theta_{i,j}=1$. This extreme case is not just a mathematical possibility but actually happens in real-world applications, including genomics. Indeed, it could be that $\theta_i=\theta_j=\theta_k=0$, while $\theta_{i,j,k}=1$, and so on. It is just this kind of behavior that has motivated the development of software for high-performance parallel processing. But this extreme behavior is rare. More typically, if both $\theta_i$ and $\theta_j$ are very small, then $\theta_{i,j}$ is small.

To take advantage of this behavior, rather than consider each CoD as a deterministic quantity associated with a fixed multivariate distribution of $\{Y,X_1,X_2,\ldots,X_n\}$, we will consider the multivariate distribution to be random, thereby making the CoDs random, and allowing us to discuss the probability that $\theta_{i,j}$ is small given that $\theta_i$ and $\theta_j$ are small. Specifically, we can consider the probability that $\theta_{i,j}>\delta$ given that $\theta_i<\lambda$ and $\theta_j<\lambda$. If we are only concerned with feature sets for which the CoDs exceed a given value $\delta$, then (on average) little is lost by not computing the joint CoD if both marginal CoDs are beneath $\lambda$, and $\lambda$ is sufficiently smaller than $\delta$. We skip the computation of $\theta_{i,j}$ if and only if $\max\{\theta_i,\theta_j\}<\lambda$.

More generally, given a function $g:[0,1]\times[0,1]\to\mathbb{R}$ and a number $\lambda\in\mathbb{R}$, we can use the following criterion: $\theta_{i,j}$ will not be computed if $g(\theta_i,\theta_j)<\lambda$. For instance, instead of $g(\theta_i,\theta_j)=\max\{\theta_i,\theta_j\}$, one might use $g(\theta_i,\theta_j)=\theta_i+\theta_j$. In this paper, we confine ourselves to the maximum and, in this case, we can assume $\lambda\in[0,1]$.

The requirement that $\max\{\theta_i,\theta_j\}$ exceed $\lambda$ for the calculation of $\theta_{i,j}$ under the supposition that $\theta_{i,j}$ must exceed $\delta$ to be significant can be viewed as a condition on the incremental determinations for the set $\{X_i,X_j\}$ relative to the sets $\{X_i\}$ and $\{X_j\}$, where the incremental determination for feature set $\mathbf{Z}$ relative to a subset $\mathbf{X}\subseteq\mathbf{Z}$ is defined to be $\theta_{\mathbf{Z}}-\theta_{\mathbf{X}}$ [5]. The condition can be interpreted as the assumption that increments cannot exceed $\delta-\lambda$. Under this assumption, $\max\{\theta_i,\theta_j\}<\lambda$ implies $\theta_{i,j}\leqslant\delta$ and therefore there is no point to computing $\theta_{i,j}$. A number of issues are relevant to the condition that $\max\{\theta_i,\theta_j\}\geqslant\lambda$ for computing $\theta_{i,j}$.

One issue is computational savings. For the criterion $\max\{\theta_i,\theta_j\}<\lambda$ not to compute $\theta_{i,j}$, the probability

$$\gamma_{ij}(\lambda)=P(\max\{\theta_i,\theta_j\}<\lambda)$$

provides a measure of *computational saving*: the probability of not computing $\theta_{i,j}$. Assuming continuous distributions, $\gamma_{ij}(\lambda)\to 0$ as $\lambda\to 0$ and $\gamma_{ij}(\lambda)\to 1$ as $\lambda\to 1$.

Typically, we are interested in CoDs above some threshold level. Accordingly, we say that $\theta_{i,j}$ is *significant at level* $\delta$ if $\theta_{i,j}>\delta$. If we skip the computation of $\theta_{i,j}$ when $\max\{\theta_i,\theta_j\}<\lambda$, then the probability

$$\rho_{ij}(\lambda,\delta)=P(\theta_{i,j}>\delta\,|\,\max\{\theta_i,\theta_j\}<\lambda)$$

provides a measure of *risk of losing significant CoDs*. It gives the probability of a pairwise CoD being significant at level $\delta$, given that it is not computed (at level $\lambda$). A tolerance for such risk can be expressed by formulating a requirement that $\rho_{ij}(\lambda,\delta)<\tau$. In this case, $\tau$ quantifies our tolerance level by providing an upper bound on the probability with which we are willing to miss a significant relationship on account of omitting the computation. We would like $\lambda$ to be as high as possible to save as much computation as possible, and we would like to have $\tau\approx 0$; however, the problem is that as $\lambda$ increases, the probability of $\theta_{i,j}$ exceeding $\delta$ also tends to increase.

Expanding the conditional probability defining $\rho_{ij}(\lambda,\delta)$ and dividing both sides of the equation by $P(\theta_{i,j}>\delta)$ yields

$$\begin{aligned}\frac{\rho_{ij}(\lambda,\delta)}{P(\theta_{i,j}>\delta)}&=\frac{P(\theta_{i,j}>\delta,\max\{\theta_i,\theta_j\}<\lambda)}{P(\theta_{i,j}>\delta)P(\max\{\theta_i,\theta_j\}<\lambda)}\\&=\frac{P(\max\{\theta_i,\theta_j\}<\lambda\,|\,\theta_{i,j}>\delta)}{P(\max\{\theta_i,\theta_j\}<\lambda)}\\&=\frac{\kappa_{ij}(\lambda,\delta)}{P(\max\{\theta_i,\theta_j\}<\lambda)},\end{aligned}$$

where

$$\kappa_{ij}(\lambda,\delta)=P(\max\{\theta_i,\theta_j\}<\lambda\,|\,\theta_{i,j}>\delta)$$

provides a measure of *loss for significant CoDs*. $\kappa_{ij}(\lambda,\delta)$ gives the probability of not computing a significant CoD. As in the case of $\rho_{ij}(\lambda,\delta)$, we would like

$\kappa_{ij}(\lambda,\delta)$ to be small. Also, as with $\rho_{ij}(\lambda,\delta)$, pushing up $\lambda$ to increase computational savings, also pushes up $\kappa_{ij}(\lambda,\delta)$.

The effect on $\kappa_{ij}(\lambda,\delta)$ of increasing $\lambda$ for fixed $\delta$ is easily seen by expanding $\kappa_{ij}(\lambda,\delta)$. If $\lambda_1 < \lambda_2$, then $\kappa_{ij}(\lambda_1,\delta) \leqslant \kappa_{ij}(\lambda_2,\delta)$. $\kappa_{ij}(\lambda,\delta)$ is important, because it quantifies the noncomputation of significant pairwise CoDs; however, if we take the view that we only wish to discover whether or not $\theta_{i,j}$ is significant, without necessarily finding its value, then the problem can be looked at slightly differently. According to the definition of the CoD, if $\max\{\theta_i,\theta_j\} > \delta$, then $\theta_{i,j} > \delta$. Hence, if $\max\{\theta_i,\theta_j\} > \delta$, there is no need to compute $\theta_{i,j}$. Omitting such computations means that the computational savings are enhanced, with $\gamma_{ij}(\lambda)$ being replaced by

$$\eta_{ij}(\lambda,\delta) = P(\max\{\theta_i,\theta_j\} < \lambda) + P(\max\{\theta_i,\theta_j\} > \delta).$$

We will say that $\theta_{i,j}$ is *nonredundantly significant* at level $\delta$ if $\max\{\theta_i,\theta_j\} \leqslant \delta$ and $\theta_{i,j} > \delta$. Having computed the single-variable CoDs, we need only find those that are nonredundantly significant.

Continuing to take the view that we only want to discover significant CoDs, not their values, we can adjust our notion of loss, replacing $\kappa_{ij}(\lambda,\delta)$ by

$$v_{ij}(\lambda,\delta) = P(\max\{\theta_i,\theta_j\} < \lambda \mid \theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} \leqslant \delta)$$

which is the probability of not computing nonredundantly significant CoDs. Assuming $\lambda < \delta$, we show that $v_{ij}(\lambda,\delta) \geqslant \kappa_{ij}(\lambda,\delta)$:

$$\begin{aligned}
&\kappa_{ij}(\lambda,\delta)\\
&= \frac{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} < \lambda)}{P(\theta_{i,j} > \delta)}\\
&= \frac{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} < \lambda)}{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} \leqslant \delta) + P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} > \delta)}\\
&\leqslant \frac{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} < \lambda)}{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} \leqslant \delta)}\\
&= \frac{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} < \lambda, \max\{\theta_i,\theta_j\} \leqslant \delta)}{P(\theta_{i,j} > \delta, \max\{\theta_i,\theta_j\} \leqslant \delta)}\\
&= v_{ij}(\lambda,\delta).
\end{aligned}$$

## 3. Increasing the predictor variables

In the preceding section, we paid particular attention to the case of two-predictor variables. In this section, we extend those considerations to estimation using three or more variables. The criterion for not computing the CoD can be just an extension of the criterion for two variables:

$$h(\theta_i,\theta_j,\theta_k) < \lambda,$$

where $h:[0,1]\times[0,1]\times[0,1] \to \mathbb{R}$. Another possibility is to use a function that depends on the CoDs for two variables:

$$h(\theta_{i,j},\theta_{j,k},\theta_{i,k}) < \lambda,$$

where, $\theta_{i,j}$ is defined to be 0 if it has been omitted at the previous stage. Criteria depending on other combinations of CoDs of lower order are possible. In this paper we will consider the condition

$$\max\{\theta_i,\theta_j,\theta_k\} < \lambda.$$

The following proposition is straightforward and shows that the condition $\max\{\theta_{i,j},\theta_{j,k},\theta_{i,k}\} < \lambda$ is more restrictive than the condition $\max\{\theta_i,\theta_j,\theta_k\} < \lambda$. This means that, for a fixed $\lambda$, the computational saving is bigger using the condition $\max\{\theta_i,\theta_j,\theta_k\} < \lambda$.

**Proposition 1.** *If* $\max\{\theta_{i,j},\theta_{j,k},\theta_{i,k}\} < \lambda$, *then* $\max\{\theta_i,\theta_j,\theta_k\} < \lambda$.

Under the condition $\max\{\theta_i,\theta_j,\theta_k\} < \lambda$, $\rho_{ij}(\lambda,\delta)$ becomes $\rho_{ijk}(\lambda,\delta)$. The form of $\rho_{ij}(\lambda,\delta)$ remains the same except that $\max\{\theta_i,\theta_j\}$ is replaced by $\max\{\theta_i,\theta_j,\theta_k\}$. Analogous comments apply to $\gamma_{ij}(\lambda,\delta)$, $\kappa_{ij}(\lambda,\delta)$, $\eta_{ij}(\lambda,\delta)$, and $v_{ij}(\lambda,\delta)$.

Similar considerations apply to using more than three variables.

## 4. Error of the optimal binary predictor

The theory of feature selection based on the distribution of the CoDs applies to binary-decision pattern recognition by considering only binary target variables. A special case occurs when all variables are binary. This is the situation that occurs in binary signal processing and, of importance to functional

genomics, in probabilistic Boolean networks [13]. The variable-selection theory possesses an intuitive analytic form for binary variables.

Consider a set of $n$ binary random variables $X_1, X_2, \ldots, X_n$ to predict (estimate) another binary random variable $Y$. Let $\mathbf{X}^i$ denote the *configuration* of the random vector $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ whose binary expression equals the integer $i$. For instance, for $n = 3$, $\mathbf{X}^0 = 000$, $\mathbf{X}^1 = 001, \ldots, \mathbf{X}^7 = 111$. For $n$ variables, there are $m = 2^n$ possible configurations. Assuming the binary random variables $X_1, X_2, \ldots, X_n, Y$ possess a joint probability distribution, for each configuration $\mathbf{X}^i$, we have the probabilities $r_i = P(\mathbf{X} = \mathbf{X}^i)$ and $p_i = P(\mathbf{Y} = 1|\mathbf{X}^i)$. The best predictor for $Y$ in terms of the mean absolute error (MAE) and in the absence of observations is its mean $E[Y] = \mu = \sum_{i=0}^{m-1} p_i r_i$ followed by the binary threshold function $T : [0,1] \to \{0,1\}$ defined as $T(x) = 0$ if and only if $x \leqslant 0.5$. The error of the thresholded mean predictor is given by

$$\varepsilon_\bullet = \begin{cases} \mu & \text{if } \mu \leqslant 0.5 \\ 1-\mu & \text{if } \mu > 0.5 \end{cases} = \min\{\mu, 1-\mu\}.$$

If we consider observations, then we can design another predictor for $Y$ and decrease the error. Let $\{X_{j_1}, X_{j_2}, \ldots, X_{j_\ell}\}$ be a subset of the variables $\{X_1, X_2, \ldots, X_n\}$ such that $1 \leqslant j_1 < j_2 < \cdots < j_\ell \leqslant n$. To develop an analytical formula for the error of the best predictor of $Y$ in terms of the MAE based on the observations of the variables $X_{j_1}, X_{j_2}, \ldots, X_{j_\ell}$, let $\mathscr{X}^k$ denote the configuration of the random vector $\mathscr{X} = (X_{j_1}, X_{j_2}, \ldots, X_{j_\ell})$ whose binary expression equals the integer $k$. For a fixed configuration $\mathscr{X}^k$ of $\mathscr{X}$, there will be some configurations $\mathbf{X}^i$ of the vector $\mathbf{X}$ that match $\mathscr{X}^k$. For instance, suppose $\mathbf{X} = (X_1, X_2, X_3, X_4, X_5)$ and we wish to predict $Y$ using only the variables $X_2$ and $X_4$. Thus, $\mathscr{X} = (X_2, X_4)$ and, for $k = 1$, we have $\mathscr{X}^1 = 01$. The configurations of $\mathbf{X}^i$ that match $\mathscr{X}^1$ are given in the following table:

| $i$ | $\mathbf{X}^i$ | $i$ | $\mathbf{X}^i$ | $i$ | $\mathbf{X}^i$ | $i$ | $\mathbf{X}^i$ |
|---|---|---|---|---|---|---|---|
| 2 | 00010 | 3 | 00011 | 6 | 00110 | 7 | 00111 |
| 18 | 10010 | 19 | 10011 | 22 | 10110 | 23 | 10111 |

If we consider the configurations for $\mathbf{X}^i$ and $\mathscr{X}^k$ as

$$\mathbf{X}^i = (X_1^i, \ldots, X_{j_1}^i, \ldots, X_{j_2}^i, \ldots, X_{j_\ell}^i, \ldots, X_n^i)$$

and

$$\mathscr{X}^k = (X_{j_1}^k, X_{j_2}^k, \ldots, X_{j_\ell}^k),$$

then, for a fixed $k$, we can define the set $\{i\colon \mathbf{X}^i \sim \mathscr{X}^k\}$ as

$$\{i\colon \mathbf{X}^i \sim \mathscr{X}^k\} = \{i\colon X_{j_1}^i = X_{j_1}^k, X_{j_2}^i = X_{j_2}^k, \ldots, X_{j_\ell}^i = X_{j_\ell}^k\}.$$

For instance, if $k = 1$, the set $\{i\colon \mathbf{X}^i \sim \mathscr{X}^1\}$ for the preceding table is $\{2, 3, 6, 7, 18, 19, 22, 23\}$.

Now, let us introduce two probabilities:

$$A_k(\mathscr{X}) = P(\mathscr{X} = \mathscr{X}^k) = \sum_{\{i\colon \mathbf{X}^i \sim \mathscr{X}^k\}} r_i,$$

$$B_k(\mathscr{X}) = P(Y = 1 \text{ and } \mathscr{X} = \mathscr{X}^k) = \sum_{\{i\colon \mathbf{X}^i \sim \mathscr{X}^k\}} p_i r_i.$$

In terms of these probabilities, the error for predicting $Y$ using the variables in the vector $\mathscr{X}$ is

$$\varepsilon_{\mathscr{X}} = \sum_{k=0}^{2^\ell - 1} \min\{B_k(\mathscr{X}), A_k(\mathscr{X}) - B_k(\mathscr{X})\}.$$

If we use all variables, $\mathscr{X} = \mathbf{X}$, then

$$A_k(\mathbf{X}) = P(\mathbf{X} = \mathbf{X}^k) = \sum_{\{i\colon \mathbf{X}^i \sim \mathbf{X}^k\}} r_i = r_k,$$

$$B_k(\mathbf{X}) = P(Y = 1 \text{ and } \mathbf{X} = \mathbf{X}^k) = \sum_{\{i\colon \mathbf{X}^i \sim \mathbf{X}^k\}} p_i r_i = p_k r_k$$

and the error of the best predictor is

$$\varepsilon_{\mathbf{X}} = \sum_{k=0}^{2^n - 1} \min\{p_k r_k, r_k - p_k r_k\} = \sum_{k=0}^{2^n - 1} \min\{p_k, 1 - p_k\} r_k.$$

## 5. Simulations

In this section, we study variable selection in the context of a binary model for the joint probability distribution $P(X_1, \ldots, X_n, Y)$. We employ a model with three parameters, $\kappa, \rho$, and $\sigma_\rho$, used previously in the analysis of estimation error [1]. A realization

of the joint probability $P(Y,X_1,\ldots,X_n)$ is defined by the probabilities $r_i$ and the conditional probabilities $p_i$. The probabilities $r_i$ are generated randomly from the Gamma distribution with parameters $\mu = 1/m$ and $\sigma = \kappa/m$, where $m = 2^n$ is the number of possible configurations. A normalization is used to satisfy the probability requirement $\sum_{i=0}^{m-1} r_i = 1$:

$$r_i = \frac{r_i'}{\sum_{i=0}^{m-1} r_i'},$$

where $r_i'$ comes from the Gamma distribution with $\mu = 1/m$ and $\sigma = \kappa/m$. The conditional probabilities $p_i$ are defined by $p_i = 1 - \rho$ for $i = 0, 1, \ldots, \alpha - 1$ and $p_i = \rho$ for $i = \alpha, \ldots, m-1$, where the integer $\alpha$ defines how many configurations produce the output $Y = 1$ and $\alpha$ is taken from the uniform distribution between 0 and $m$. In practical situations it is unlikely to have constant error contribution $\rho = \min\{p_i, 1 - p_i\}$ for all configurations. Hence, Gaussian noise with $\mu = 0$ and $\sigma = \sigma_\rho$ is added to the conditional probabilities $p_i$.

Given a realization $r_i$ and $p_i$ taken from the model $(\kappa, \rho, \sigma_\rho)$, we can directly compute the CoDs $\theta_i$ and $\theta_{i,j}$. The first simulation results presented in this section are based on $n = 5$ variables, 10,000 generated replications, and the model parameters $\kappa = 1.5$, $\rho = 0.2$, and $\sigma_\rho = 0.2$.

Fig. 1(a) displays the computational saving $\gamma_{ij}(\lambda)$ obtained from the model. The graphic can be used to select a value for $\lambda$ to obtain a desired computational saving. For example, for at least 90% of computational saving, we must select $\lambda \geqslant 0.4$.

The risk $\rho_{ij}(\lambda, \delta)$ of losing significant CoDs obtained from the model is shown in Fig. 1(b). Once $\lambda$ is selected, this graphic can be used to check if the risk of losing CoDs $\theta_{i,j} > \delta$ is smaller than a desired tolerance. For example, if the tolerance for the risk pertaining to coefficients $\theta_{i,j} > 0.6$ is 10%, then for $\lambda = 0.4$ this tolerance is satisfied, since the risk is less than 1%.

Fig. 1(c) shows the probability $\kappa_{ij}(\lambda, \delta)$ of not computing significant CoDs. Once $\lambda$ is selected, the graphic can be used to check if the risk of losing CoDs $\theta_{i,j} > \delta$ is smaller than a desired tolerance. For example, if the tolerance for losing coefficients $\theta_{i,j} > 0.7$ is 20%, then for $\lambda = 0.4$ this condition is satisfied, since the risk is less than 10%.

Fig. 1(d) shows the probability $\nu_{ij}(\lambda, \delta)$ of not computing nonredundantly significant coefficients. Again considering tolerance, if the tolerance for noncomputation of coefficients $\theta_{i,j} > 0.7$ is 20%, then for $\lambda = 0.4$ this condition is satisfied, since the risk is less than 10%.

Fig. 1(e) shows the loss for significant CoDs versus computational saving. This graphic shows that when computational saving is high, the respective loss of significant CoDs is not so high. For example, for a 90% computational saving, the loss of significant coefficients is less than 10%.

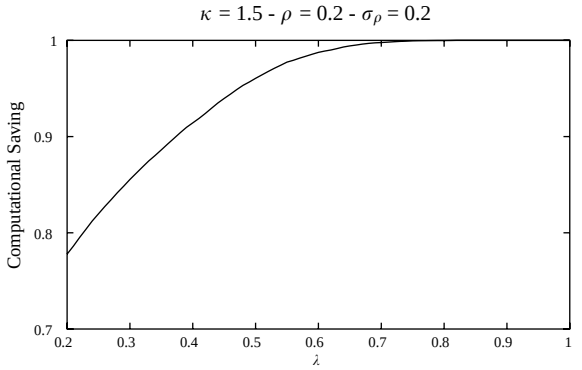
The curves in Fig. 1 are model dependent, for model $(n, \kappa, \rho, \sigma_\rho) = (5, 1.5, 0.2, 0.2)$. Fig. 2 shows a set of curves for a different model with $(n, \kappa, \rho, \sigma_\rho) = (5, 2.8, 0.3, 0.1)$. Notice in Fig. 2(b) the smaller risk, with the risk being essentially 0 for all $\lambda$ when $\delta = 0.7$ or 0.8. Notice in Fig. 2(c) the basically vertical curvers for $\delta = 0.7$ and 0.8. This means that even for an extremely small increment $\delta - \lambda$, the loss is essentially 0 for $\delta = 0.7$ and 0.8, so long as $\lambda < \delta$. A very different situation is depicted for the model $(n, \kappa, \rho, \sigma_\rho) = (5, 1.5, 0.1, 0.2)$ in Fig. 3. Not only are the risks in Fig. 3(b) much greater, but the curves in succeeding parts of the figure rise more slowly.

The model not only depends on $\kappa$, $\rho$ and $\sigma_\rho$, but also on $n$. In Figs. 4 and 5, $n = 10$. Although similar families of curves occur as with $n = 5$, these result from different choices of $\kappa$, $\rho$ and $\sigma_\rho$ than for $n = 5$. Note the similarities between the curves in Fig. 4, corresponding to the model $(n, \kappa, \rho, \sigma_\rho) = (10, 9.0, 0.12, 0.01)$, with the curves in Fig. 1. Also note the similarities between the curves in Fig. 5 corresponding to the model, $(n, \kappa, \rho, \sigma_\rho) = (10, 1.5, 0.2, 0.2)$, with the curves in Fig. 2.

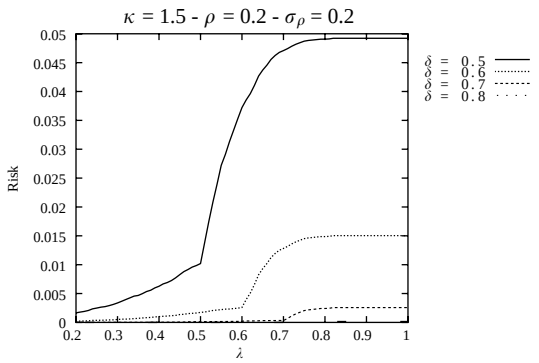
## 6. Glioma application

In this section, we apply the variable-selection theory to real gene-expression data from glioma patients. We will see that there are similarities between the behavior of the simulation-based coefficients and those from actual cancer patients. More importantly, the loss of high coefficients will be seen to be relatively small even when significant computational savings are achieved.

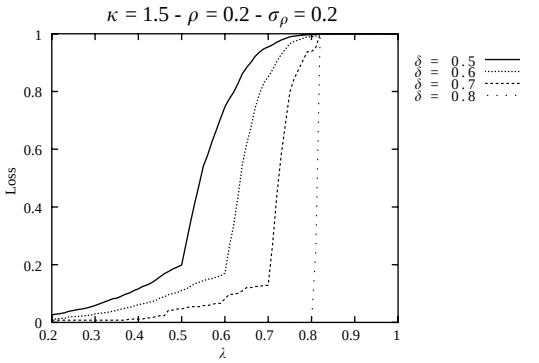
The study of expression-level prediction has recently been made possible by the development of cDNA microarrays, in which transcript levels can be



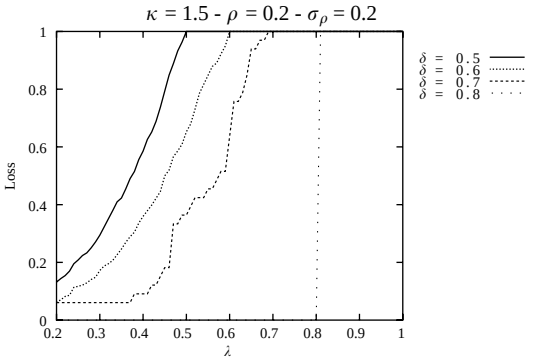

(a) Computational saving.



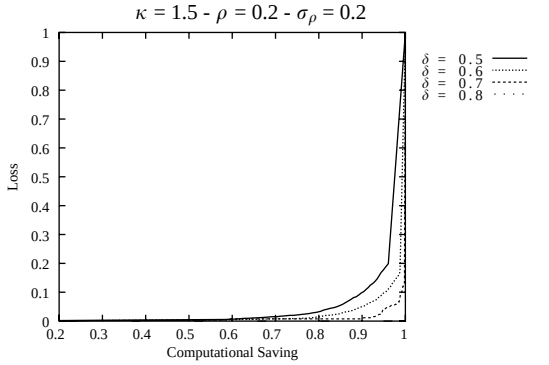

(b) Risk of losing significant CoDs.




(c) Loss for significant CoDs.

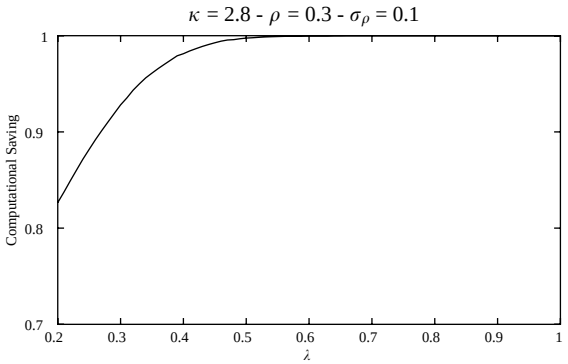



(d) Loss for nonredundant significant CoDs.

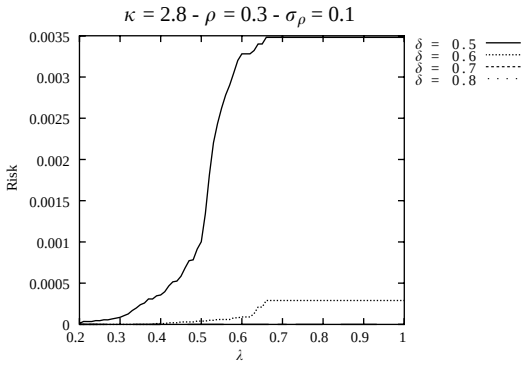



(e) Loss for significant CoDs × Computational saving.

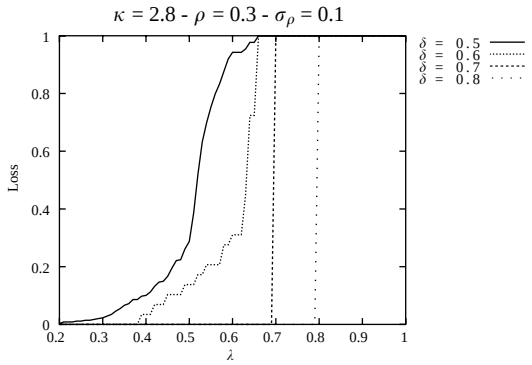
Fig. 1. Simulations for model $(n, \kappa, \rho, \sigma_\rho) = (5, 1.5, 0.2, 0.2)$: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, (d) loss for nonredundant significant CoDs, and (e) loss for significant CoDs $\times$ computational saving.



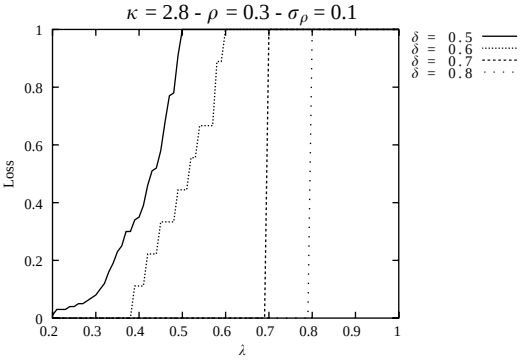

(a) Computational saving.



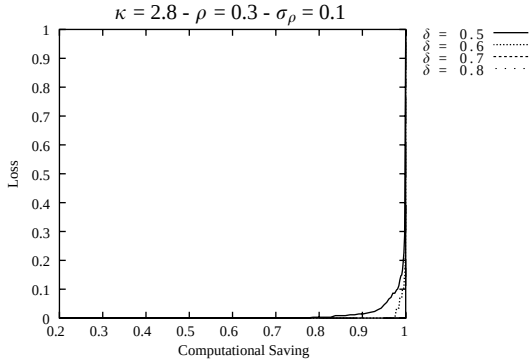

(b) Risk of losing significant CoDs.




(c) Loss for significant CoDs.

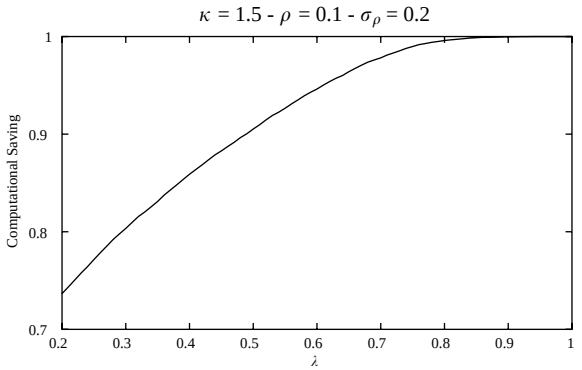



(d) Loss for nonredundant significant CoDs.

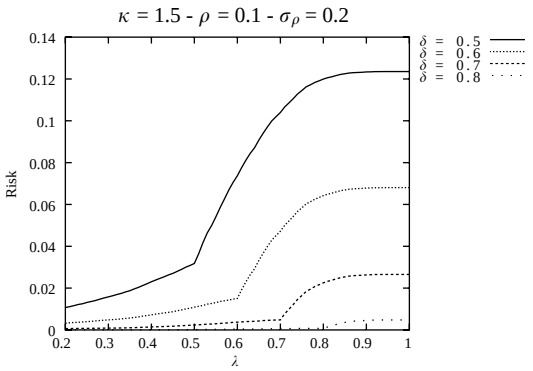



(e) Loss for significant CoDs × Computational saving.

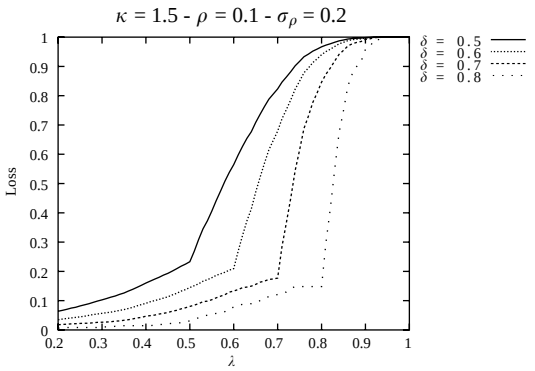
Fig. 2. Simulations for model $(n, \kappa, \rho, \sigma_\rho) = (5, 2.8, 0.3, 0.1)$: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, (d) loss for nonredundant significant CoDs, and (e) loss for significant CoDs × computational saving.



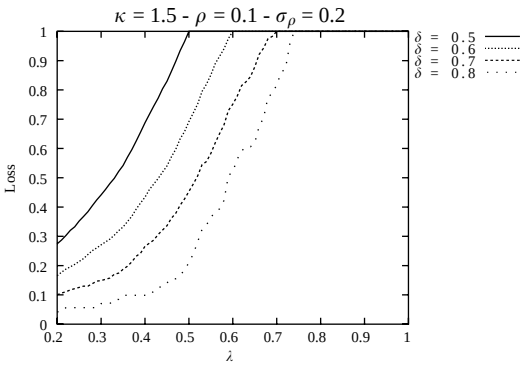

(a) Computational saving.



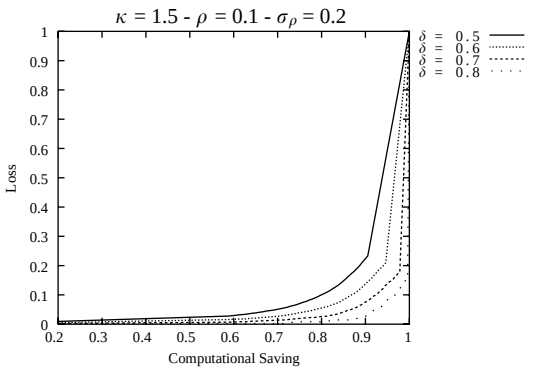

(b) Risk of losing significant CoDs.




(c) Loss for significant CoDs.fi

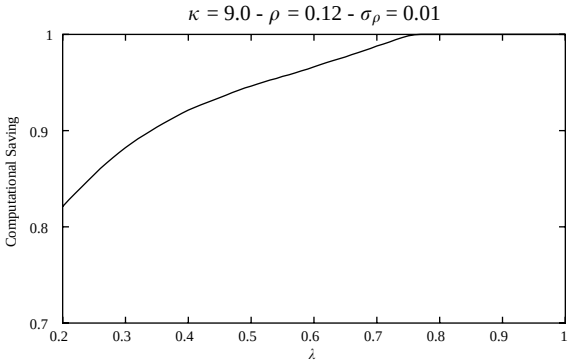



(d) Loss for nonredundant significant CoDs.

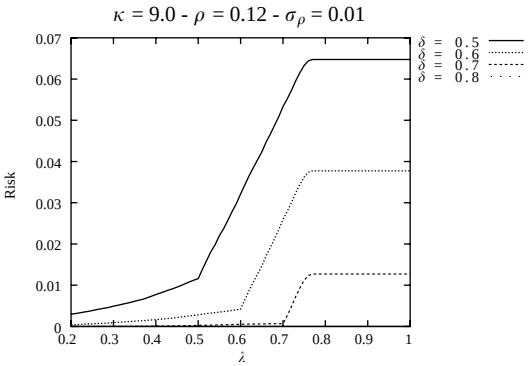



(e) Loss for significant CoDs × Computational saving.

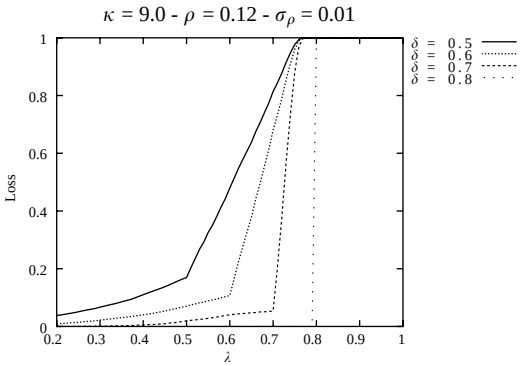
Fig. 3. Simulations for model $(n, \kappa, \rho, \sigma_\rho) = (5, 1.5, 0.1, 0.2)$: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, (d) loss for nonredundant significant CoDs, and (e) loss for significant CoDs $\times$ computational saving.



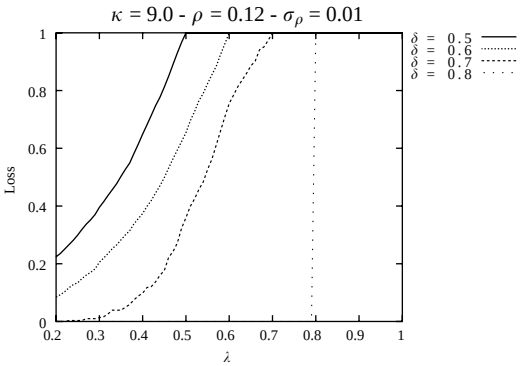

(a) Computational saving.



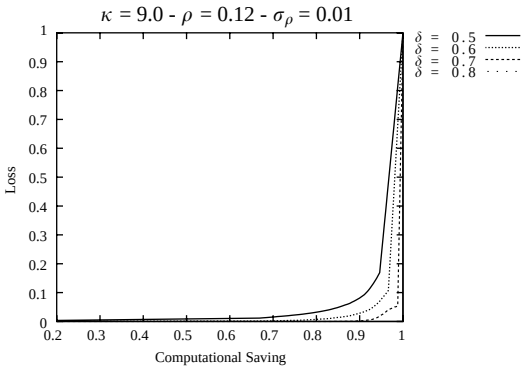

(b) Risk of losing significant CoDs.




(c) Loss for significant CoDs.

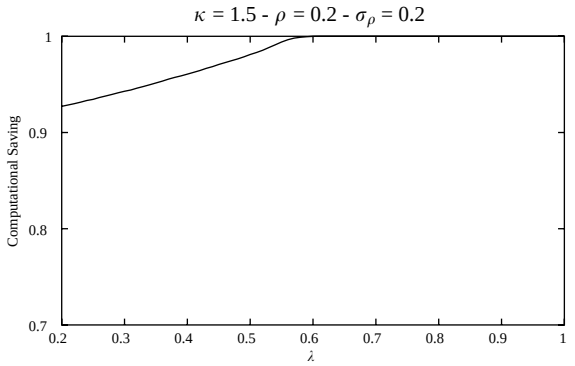



(d) Loss for nonredundant significant CoDs.

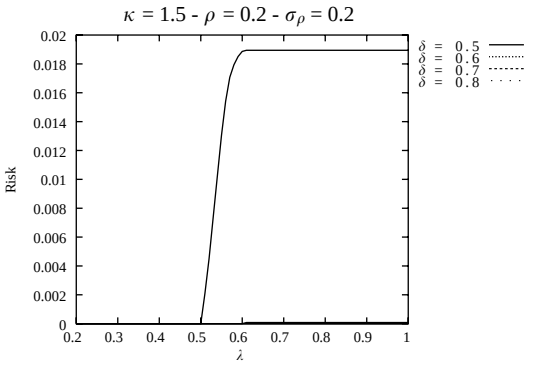



(e) Loss for significant CoDs × Computational saving.

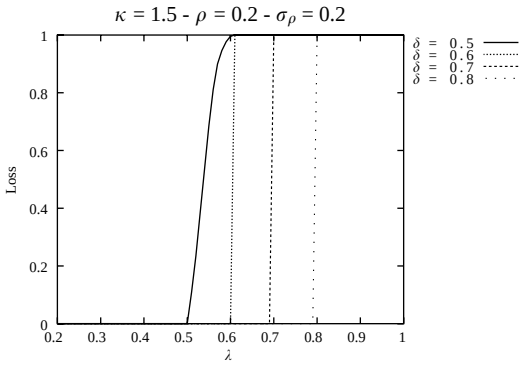
Fig. 4. Simulations for model $(n, \kappa, \rho, \sigma_\rho) = (10, 9.0, 0.12, 0.01)$: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, (d) loss for nonredundant significant CoDs, and (e) loss for significant CoDs $\times$ computational saving.



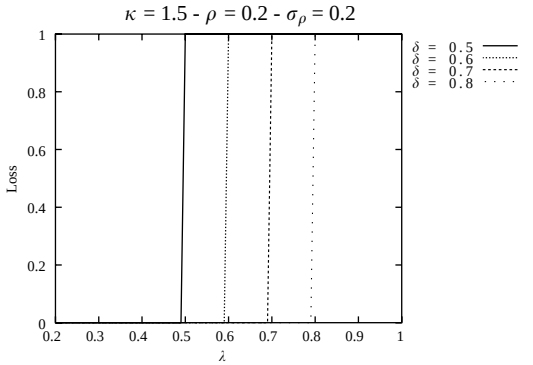

(a) Computational saving.



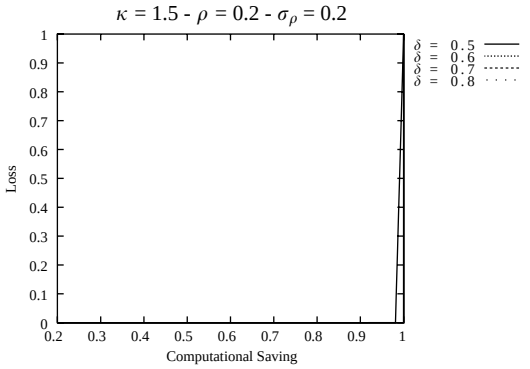

(b) Risk of losing significant CoDs.




(c) Loss for significant CoDs.




(d) Loss for nonredundant significant CoDs.




(e) Loss for significant CoDs · Computational saving.

Fig. 5. Simulations for model $(n, \kappa, \rho, \sigma_\rho) = (10, 1.5, 0.2, 0.2)$: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, (d) loss for nonredundant significant CoDs, and (e) loss for significant CoDs $\times$ computational saving.

determined for thousands of genes simultaneously [4,6,11]. Microarray data has been used to design discrete nonlinear predictors and find significant CoDs among the genes, the variables being gene expression levels [8,9]. The data are discrete because the analog expression levels are quantized. Here we consider binary quantization: [0 (down-regulated), 1 (up-regulated)]. Ternary quantization can be used when a test of significance is applied to determine down or up regulation: [−1 (down-regulated), 1 (up-regulated), or 0 (invariant)]. External stimuli are quantified as 1 [present] and 0 [not present]. Because there are many genes and a very small number of microarrays, it is not practical to precisely design classifiers, but it is possible to estimate CoDs. Here we consider binarized expression data for 597 genes derived from 26 human glioma surgical tissue samples [7].

We have chosen 47 target genes from the 597 genes. To compute the computational savings and risks, for each target gene, we have estimated the CoDs for all combinations of one, two and three predictors from the other 596 genes. This has been done by using a massively parallel computer. CoD estimation involves estimating the optimal classifier from the data, using cross-validation to estimate the Bayes error, and then forming an estimate $\hat{\theta}$ of $\theta$ according to the definition of the CoD. In the estimation setting it may not be true that the data directly yields $\max\{\hat{\theta}_i, \hat{\theta}_j\} \leqslant \hat{\theta}_{i,j}$; however, given that $\max\{\theta_i, \theta_j\} \leqslant \theta_{i,j}$, if this relation is not directly given by the data, then the estimate of $\theta_{i,j}$ is taken to be $\max\{\hat{\theta}_i, \hat{\theta}_j\}$ [8]. Figs. 6–8 present the computational savings $\hat{\gamma}_{ij}(\lambda)$, the risk $\hat{\rho}_{ij}(\lambda,\delta)$ of losing significant CoDs, and the probability $\hat{\kappa}_{ij}(\lambda,\delta)$ of not computing significant coefficients, respectively, for the glioma data across all 47 target genes when using two and three predictors (the "hat" notation indicating that these values have been computed from sample data, not a theoretical probability distribution).

Table 1 shows the number of computed and not computed significant coefficients for various pairs of $\lambda$ and $\delta$. It also shows the computational savings, which are substantial. The loss of significant CoDs (not computed column) is quite small when $\delta - \lambda = 0.2$, and very small when $\delta - \lambda = 0.3$.

Fig. 9 shows the probability $\hat{\nu}_{ij}(\lambda,\delta)$ of not computing nonredundantly significant coefficients. As in the model-based simulations, the results are not as striking

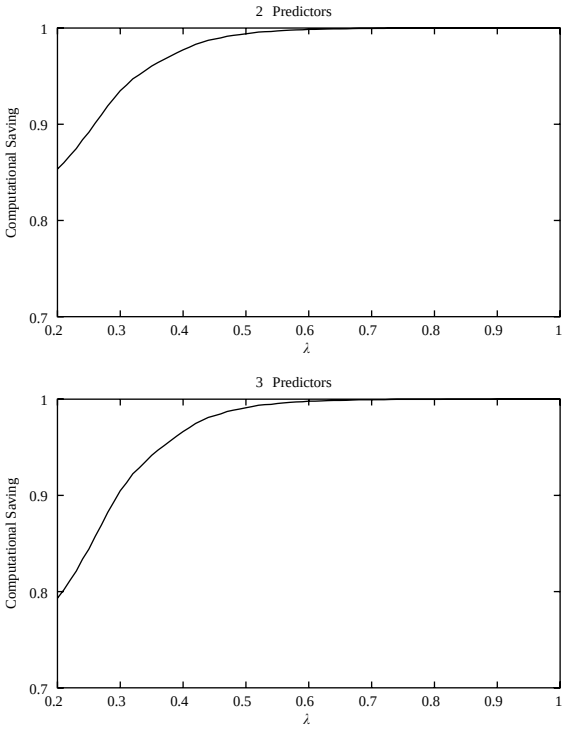



Fig. 6. Computational saving for glioma data.

as for the probability $\hat{\kappa}_{ij}(\lambda,\delta)$ of not computing significant coefficients, shown in Fig. 8. Nonetheless, for high $\delta$ values the probabilities are modest for $\lambda = 0.3$, $\hat{\nu}_{ij}(0.3, 0.7) \approx \hat{\nu}_{ij}(0.3, 0.8) \approx 0.2$ for both two and three predictors, and $\hat{\nu}_{ij}(0.3, 0.8) < 0.3$ for three predictors. Referring to Fig. 6, we see that, for both two and three predictors, there is significant computational savings, with $\hat{\gamma}_{ij}(0.3) > 0.9$. Moreover, when comparing $\hat{\nu}_{ij}(\lambda,\delta)$ to $\hat{\kappa}_{ij}(\lambda,\delta)$, it must be remembered that the loss is mitigated because the redundantly significant coefficients are not computed, and therefore, although we know they are significant, we do not know their actual values, which we would know in the case of $\hat{\kappa}_{ij}(\lambda,\delta)$ and which often must be known—for instance, in the case of probabilistic Boolean networks.

Finally, the loss of significant coefficients versus computational saving for $\lambda = 0.3$ for the glioma data for two and three predictors is presented in Fig. 10.

In practice, one would like to be able to use the various sets of curves to make computational decisions based on increments of interest. One potential

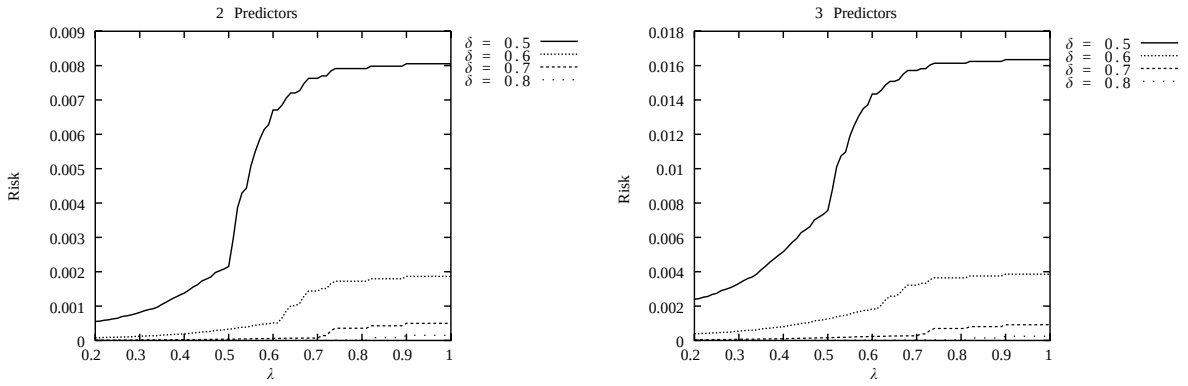



Fig. 7. Risk of losing significant CoDs for glioma data.

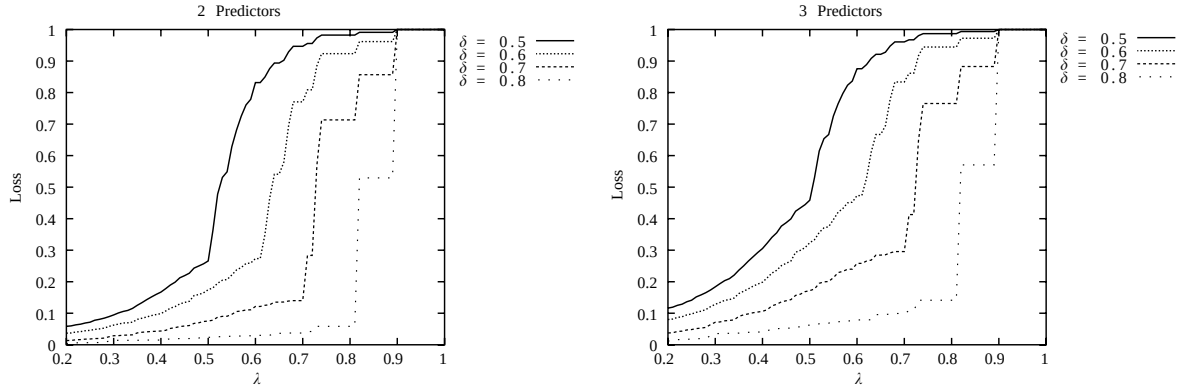



Fig. 8. Loss for significant CoDs for glioma data.

approach to this problem is to have a model whose parameters can be estimated from the data, and then generate the curves for the model. A key obstacle to this approach is finding a model that is sufficiently complex to produce curves that sufficiently approximate the curves for the population of interest. If this can be accomplished, then a second issue would be to find satisfactory estimators of the model parameters. Leaving this problem to future research, we here consider a different approach.

If one has a large number of genes, say 500 or 1000, the CoD computations for three-predictor variables are prohibitive and require extensive time on a supercomputer. One way to use the theory presented in this paper is to choose a manageable subset of the total and compute the relevant curves for that subset. If these curves sufficiently approximate the curves for the full set, then computational decisions can be based on the curves from the subset. To test the feasibility of this approach, we have randomly chosen 50 subsets of 25 genes from the set of 47 genes, computed the relevant curves for each of the 50 subsets, and found the mean curves for $\gamma_{ij}(\lambda)$, $\rho_{ij}(\lambda,\delta)$, $\kappa_{ij}(\lambda,\delta)$, and $v_{ij}(\lambda,\delta)$. These mean curves are shown in Fig. 11 for three predictors. Note how close they are to the full-glioma-set curves. This indicates that, on average, choosing a subset of 25 genes provides accurate curves.

A salient issue remains: if we randomly select a single subset of 25 genes from a set of 1000 genes and use the curves for the subset of 25 genes to make decisions, what kind of deviation from the mean of all possible 25-gene subsets can we expect for the four types of curves? To address this question, we have computed the variance curves for $0.2 \leqslant \lambda \leqslant 0.6$ (below which computational savings are too low and above which

Table 1
Computed and not computed significant CoDs

| Number of predictors | $\lambda$ | $\delta$ | Computational saving | Significant coefficients of determination that are computed | Significant coefficients of determination that are not computed |
|---|---|---|---|---|---|
| 02 | 0.40 | 0.60 | 97.74% | 14016 (90.11%) | 1538 (9.89%) |
| 03 | 0.40 | 0.60 | 96.64% | 5104453 (80.16%) | 1263560 (19.84%) |
| 02 | 0.40 | 0.70 | 97.74% | 3970 (95.64%) | 181 (4.36%) |
| 03 | 0.40 | 0.70 | 96.64% | 1345615 (89.39%) | 159741 (10.61%) |
| 02 | 0.40 | 0.80 | 97.74% | 1245 (98.50%) | 19 (1.50%) |
| 03 | 0.40 | 0.80 | 96.64% | 394852 (95.94%) | 16722 (4.06%) |
| 02 | 0.50 | 0.70 | 99.40% | 3838 (92.46%) | 313 (7.54%) |
| 03 | 0.50 | 0.70 | 99.10% | 1246813 (82.83%) | 258543 (17.17%) |
| 02 | 0.50 | 0.80 | 99.40% | 1234 (97.63%) | 30 (2.37%) |
| 03 | 0.50 | 0.80 | 99.10% | 385966 (93.78%) | 25608 (6.22%) |
| 02 | 0.60 | 0.80 | 99.86% | 1226 (96.99%) | 38 (3.01%) |
| 03 | 0.60 | 0.80 | 99.80% | 379110 (92.11%) | 32464 (7.89%) |

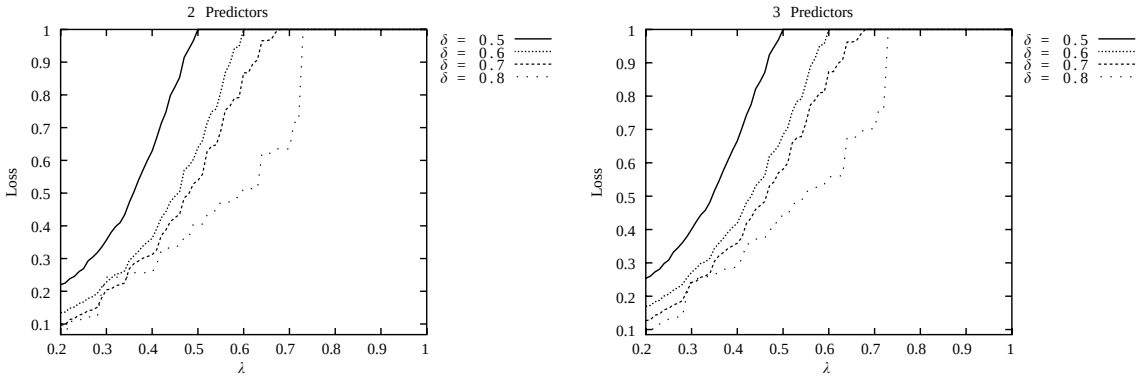



Fig. 9. Loss for nonredundant significant CoDs for glioma data.

computational savings exceed 99%) corresponding to $\gamma_{ij}(\lambda)$, $\rho_{ij}(\lambda,\delta)$, $\kappa_{ij}(\lambda,\delta)$, and $\nu_{ij}(\lambda,\delta)$ for three predictors using the same 50 randomly selected subsets. These are shown in Fig. 12. For the cases of $\rho_{ij}(\lambda,\delta)$, $\kappa_{ij}(\lambda,\delta)$, and $\nu_{ij}(\lambda,\delta)$, the variance is very small for the lower values of $\delta$, which is when the probabilities are higher. The variance is higher for higher values of $\delta$, but in this case the probabilities are much smaller, so that a larger variance does not portend as much chance of having a large risk or loss when using the mean curves. Indeed, one can adjust the mean curves by adding a standard deviation at each value of $\lambda$ to lessen the possibility of an optimistic assessment.

## 7. Algorithm

In this section, we present an algorithm for selecting subsets (with a fixed cardinality $p$, $1 < p \leqslant n$) from the set $\{X_1, X_2, \ldots, X_n\}$ to compute CoDs under the maximum requirement, $\max\{\theta_{j_1}, \theta_{j_2}, \ldots, \theta_{j_p}\} < \lambda$. To motivate the section, let us consider the simple case

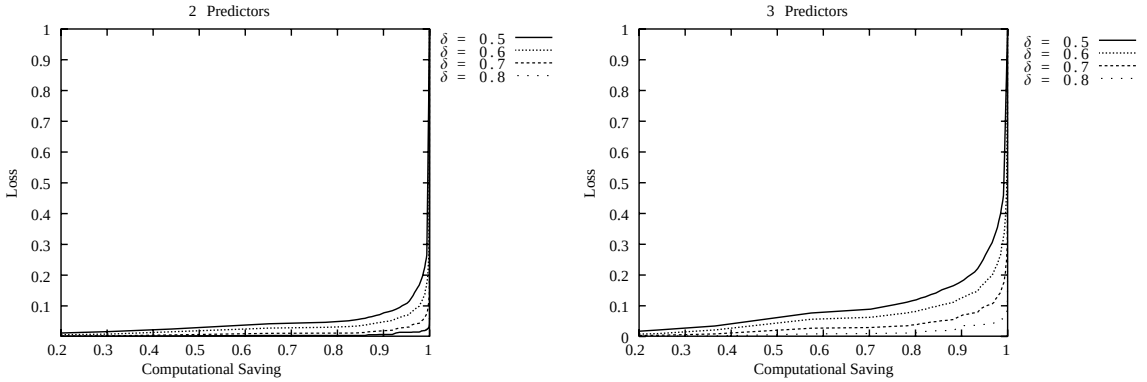
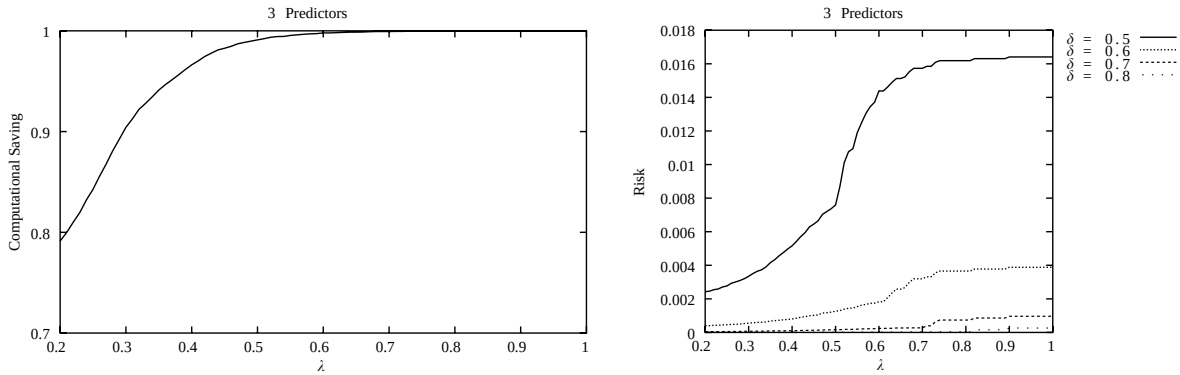
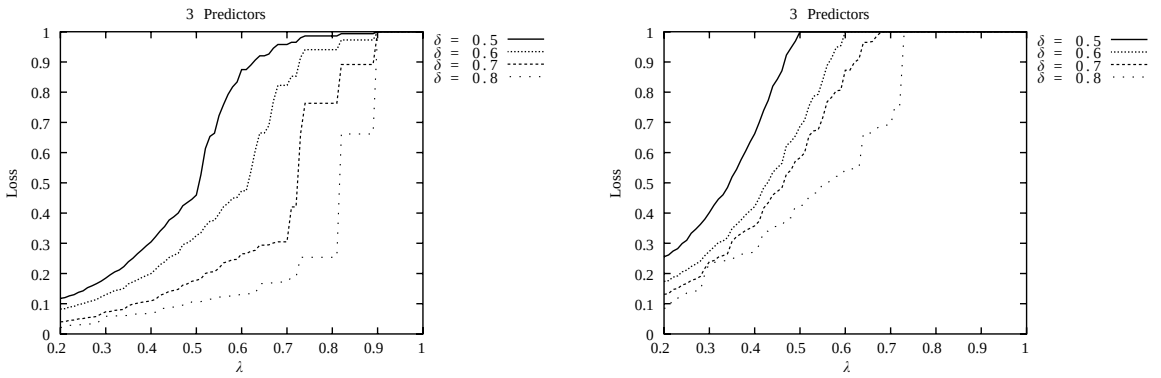



Fig. 10. Loss for significant CoDs × computational saving for glioma data.




(a) Computational saving.

(b) Risk of losing significant CoDs.




(c) Loss for significant CoDs.

(d) Loss for nonredundant significant CoDs.

Fig. 11. Mean curves for glioma data: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, and (d) loss for nonredundant significant CoDs.

where $p=2$. In this case, a trivial selection procedure is given by the following algorithm:

```
for each i1 in {1,2,...,n-1} do
  for each i2 in {i1+1,...,n} do
    if max{θ_{i1}, θ_{i2}} ⩾ λ then
      compute θ_{i1,i2}.
```

Although this algorithm selects exactly the CoDs according to the criterion $\max\{\theta_{i_1},\theta_{i_2}\}\geqslant\lambda$, there is no difference (in terms of time complexity) between it and the algorithm that computes all CoDs for two variables. In fact, the former performs $C(n,2)$ comparisons and the latter $C(n,2)$ computations. We present an efficient algorithm for selecting subsets to compute CoDs.

If one of the coefficients in $\{\theta_{i_1},\theta_{i_2},\ldots,\theta_{i_p}\}$ is greater than or equal to $\lambda$, then $\max\{\theta_{i_1},\theta_{i_2},\ldots,\theta_{i_p}\}\geqslant\lambda$. Thus, we can proceed in the following way. Sort the coefficients in $\{\theta_1,\theta_2,\ldots,\theta_n\}$ to obtain $\{\theta_{j_1},\theta_{j_2},\ldots,\theta_{j_n}\}$ such that $\theta_{j_1}\geqslant\theta_{j_2}\geqslant\cdots\geqslant\theta_{j_n}$, and then apply the following algorithm:

```
Algorithm Select:
  Input: {θ_{j1}, θ_{j2},..., θ_{jn}},
    such that θ_{j1} ⩾ θ_{j2} ⩾ ··· ⩾ θ_{jn},
  λ ∈ [0,1] and 1 < p ⩽ n.
Output: θ_{j_{i1}, j_{i2},..., j_{ip}} such that
  max{θ_{j_{i1}}, θ_{j_{i2}},..., θ_{j_{ip}}} ⩾ λ.

Let q = min{|{i: θ_i ⩾ λ}|, n - p + 1}.
for each i1 in {1,2,...,q} do
  for each i2 in {i1+1,...,n-p+2} do
  for each i3 in {i2+1,...,n-p+3} do
                    ⋱
    for each i_{p-1} in {i_{p-2}+1,...,n-1} do
    for each ip in {i_{p-1}+1,...,n} do
    compute θ_{j_{i1}, j_{i2},..., j_{ip}}.
```

We state two propositions without proof. The first assures correctness. The second gives the time complexity.

**Proposition 2.** *Let* $1<p\leqslant n$ *and* $1\leqslant i_1<i_2<\cdots<i_p\leqslant n$. *The CoD* $\theta_{j_{i_1},j_{i_2},\ldots,j_{i_p}}$ *is computed by* Select *if and only if* $\max\{\theta_{j_{i_1}},\theta_{j_{i_2}},\ldots,\theta_{j_{i_p}}\}\geqslant\lambda$.

**Proposition 3.** *Let* $1<p\leqslant n$ *and* $q=\min\{|\{i\colon\theta_i\geqslant\lambda\}|,\ n-p+1\}$. *The time complexity of the algorithm* Select *is* $F(n,p,q)$, *where*

$$F(n,p,q)=\begin{cases}C(n,p) & \text{if } q=n-p+1,\\ C(n,p) & \\ \quad -C(n-q,p) & \text{if } q<n-p+1\end{cases}$$

*and where* $n$, $p$ *and* $q$ *are integers such that* $n>0$, $1<p\leqslant n$ *and* $0\leqslant q\leqslant n-p+1$.

To prove that the algorithm Select is the best one in terms of time complexity, we have to show that, given $1<p\leqslant n$ and $\lambda\in[0,1]$, the cardinality of the sets

$$\mathscr{S}(p,\lambda)=\{\{j_{i_1},j_{i_2},\ldots,j_{i_p}\}\colon\theta_{j_{i_1},j_{i_2},\ldots,j_{i_p}}\ \text{is computed by Select}\}$$

and

$$\mathscr{A}(p,\lambda)=\{\{j_{i_1},j_{i_2},\ldots,j_{i_p}\}\colon 1\leqslant i_1<i_2<\cdots<i_p\leqslant n\ \text{and}\ \max\{\theta_{j_{i_1}},\theta_{j_{i_2}},\ldots,\theta_{j_{i_p}}\}\geqslant\lambda\}$$

are both equal to $F(n,p,q)$, where $q=\min\{|\{i\colon\theta_i\geqslant\lambda\}|,\ n-p+1\}$. In fact, by Proposition 2, we have that $\mathscr{S}(p,\lambda)=\mathscr{A}(p,\lambda)$ and, by Proposition 3, $|\mathscr{S}(p,\lambda)|=F(n,p,q)$.

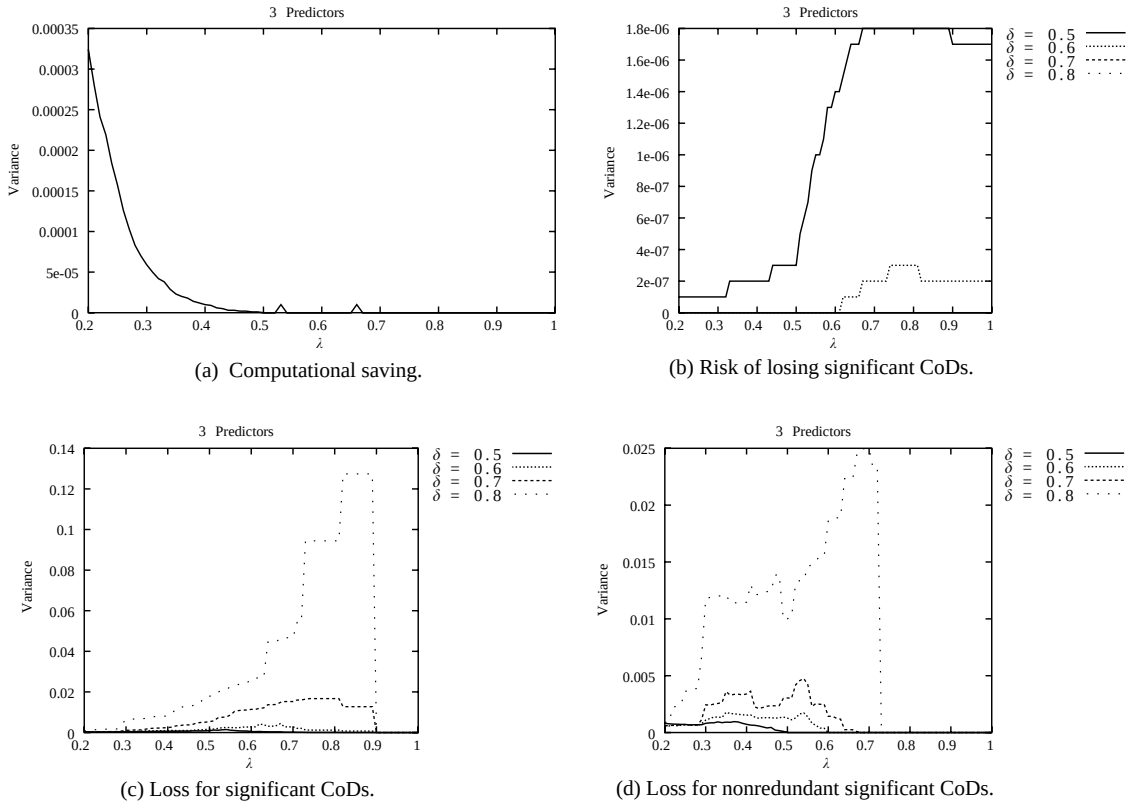
Now, by the results obtained from this section, we can easily compute the computational saving of the algorithm Select. In fact, given $\lambda$, we can compute $q=\min\{|\{i\colon\theta_i\geqslant\lambda\}|,\ n-p+1\}$, and the computational saving for Select is given by

$$1-\frac{F(n,p,q)}{C(n,p)}.$$

Note that this value must agree the computational saving $P(\max\{\theta_{i_1},\theta_{i_2},\ldots,\theta_{i_p}\}<\lambda)$.

To conclude, we comment on how to find the value for $\lambda$ given the sorted set $\{\theta_{j_1},\theta_{j_2},\ldots,\theta_{j_n}\}$ and a desired number $M<C(n,p)$ of CoDs we wish to compute. This situation arises when $M$ imposes a limit on the number of computations allowed, which would constitute a real-time requirement. Since $F(n,p,q)$ is the number coefficients computed by Select, then we must find $q<n-p+1$ such that

$$F(n,p,q)=C(n,p)-C(n-q,p)\leqslant M$$




(a) Computational saving.

(b) Risk of losing significant CoDs.

(c) Loss for significant CoDs.

(d) Loss for nonredundant significant CoDs.

Fig. 12. Variance curves for glioma data: (a) computational saving, (b) risk of losing significant CoDs, (c) loss for significant CoDs, and (d) loss for nonredundant significant CoDs.

and

$$M < C(n,q) - C(n-(q+1),p) = F(n,p,q+1).$$

Once the value for $q$ is found, it is easy to see that $\lambda = \theta_{j_q}$.

## 8. Conclusion

A rigorous analysis of feature selection based on incremental determination has been provided. This analysis shows the possible benefits of such a methodology based on a probabilistic analysis of both the loss of feature sets and the gain in computational efficiency. Given the desire in practical genomic applications to find potentially good predictor sets, in many circumstances the loss may well be offset by the enormous savings in computation, which could be sufficiently great to preclude an exhaustive search.

## Acknowledgements

This research was supported in part by the National Human Genome Research Institute, the University of Texas MD Anderson Cancer Center, FAPESP (Grant 98/15586-9) and NuTec Sciences, Inc.

## References

[1] M. Brun, D. Sabbagh, S. Kim, E.R. Dougherty, Corrected small-sample estimation for the error of the optimal binary filter, Bioinformatics, in press.

[2] T. Cover, J. Van Campenhout, On the possible orderings in the measurement selection problem, IEEE Trans. Systems Man Cybernet. 7 (1977) 657–661.

[3] P. Devijver, J. Kittler, Pattern Recognition: A Statistical Approach, Prentice-Hall, Englewood Cliffs, NJ, 1982.
[4] J.L. De Risi, V.R. Ilyer, P.O. Brown, Exploring the metabolic and genetic control of gene expression on a genomic scale, Science 278 (1997) 680–686.
[5] E.R. Dougherty, S. Kim, Y. Chen, Coefficient of determination in nonlinear signal processing, Signal Process. 80 (10) (2000) 2219–2235.
[6] D.J. Duggan, M.L. Bittner, Y. Chen, J.M. Trent, Expression using cDNA microarrays, Natur. Genet. 21 (1999) 10–14.
[7] G.N. Fuller, C.H. Rhee, K.R. Hess, L.S. Caskey, R. Wang, J.M. Bruner, W.K. Yung, W. Zhang, Reactivation of insulin-like growth factor binding protein 2 expression during glioblastoma transformation revealed by parallel gene expression profiling, Cancer Res. 59 (1999) 4228–4232.
[8] S. Kim, E.R. Dougherty, M.L. Bittner, Y. Chen, K. Sivakumar, P. Meltzer, J.M. Trent, General nonlinear framework for the analysis of gene interaction via multivariate expression arrays, J. Biomed. Opt. 5 (4) (October 2000) 411–424.
[9] S. Kim, E.R. Dougherty, Y. Chen, K. Sivakumar, P. Meltzer, J.M. Trent, M.L. Bittner, Multivariate measurement of gene expression relationships, Genomics 67 (2000) 201–209.
[10] P. Narenda, K. Fukunaga, A branch and bound algorithm for feature subset selection, IEEE Trans. Comput. 26 (1977) 917–922.
[11] M. Schena, D. Shalon, R.W. Davis, P.O. Brown, Quantitative monitoring of gene expression patterns with a complementary DNA microarray, Science 270 (1995) 467–470.
[12] G. Sebestyen, Decision-Making Processes in Pattern Recognition, Macmillan, New York, 1962.
[13] I. Shmulevich, E.R. Dougherty, W. Zhang, Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks, J. Bioinform. 18 (2002) 261–274.
[14] E.B. Suh, E.R. Dougherty, S. Kim, D.E. Russ, R.R. Martino, Parallel computing methods for analyzing gene expression relationships, in: Proceedings of the SPIE Microarrays: Optical Technologies and Informatics, San Jose, January, 2001.
[15] T. Vilmansen, Feature evaluation with measures of probabilistic independence, IEEE Trans. Comput. 22 (1973) 381–388.
[16] R.E. Walpole, R.H. Meyers, Probability and Statistics for Engineers and Scientists, 3rd Edition, Macmillan, New York, 1985.