APTER: Aggregated Prognosis Through Exponential Reweighting

Liu Yang [1,*], Kristiaan Pelckmans [1] *

[1]Department of Information Technology, Polacksbacken, Uppsala, Sweden

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

This paper considers the task of learning to make a prognosis of a patient based on its micro-array expression levels. The method is an application of the aggregation method as recently proposed in the literature on theoretical machine learning, and excels by its computational convenience and capability to deal with high-dimensional data. This paper gives a formal analysis of the method, yielding rates of convergence similar to what traditional techniques obtain, while it is shown to cope well with an exponentially large set of features. Those results are supported by numerical simulations on a range of publicly available datasets. It is empirically found that the proposed technique combined with a recently proposed preprocessing technique gives excellent performances. All employed software and datasets are available on .

1 INTRODUCTION

Learning how to make a prognosis of a patient is an important ingredient to the task of building an automatic system for personalised medical treatment. A prognosis here is understood as a useful characterisation of the (future) time of an event of interest. In cancer research, a typical event is the relapse of a patient after receiving treatment. The traditional approach to process observed event times is addressed in the analysis of survival data, see e.g. (1) for an excellent review of this mature field in statistics. Most of those techniques are based on parametric or semi-parametric assumptions on how the data was generated.

Probably the most prevalent technique is Cox' Proportional Hazard (PH) approach, where inference is made by maximising a suitable partial likelihood function. This approach has proven to be very powerful in many applications of survival analysis, but it is not clear that the basic assumption underlying this technique holds in the analysis of the microarray datasets. Specifically, the proportional hazard assumption is hard to verify and might not even be valid. This in turn jeopardises the interpretation of the results. This is especially so since the data has typically a high dimensionality while typically a few (complete) cases are available, incurring problems of ill-conditioning. Many authors suggested fixes to this problem. Some of such work proposed in the early 2000, was studied numerically and compared in (2). In applied work, one ore resorts to a proper form of preprocessing in order to use Cox' PH model, see e.g. (3).

Since prognosis involves essentially a form of prediction, it is naturally to phrase this problem in a context of modern machine learning. This insight allowed a few authors to come up with algorithms which are deviating from likelihood-based analysis. We mention here (4) and references therein.

This work takes this route even further. It studies the question *how can new insights in machine learning help to build a more powerful algorithm*? As dictated by the application, we are especially interested to deal with high-dimensional data. That is, cases where many ($O(10^4)$) covariates might potentially be relevant, while only relatively few cases ($O(10^2)$) are available. Furthermore, we are not so much interested in *recovering* the mechanisms underlying the data since that is probably too ambitious a goal. Instead, we merely aim at making a good *prognosis*. It is this rationale that makes the present technique essentially different from likelihood-based, or penalised likelihood-based approaches as e.g. the PH-L_1 (5) or Danzig Selector (6) for survival analysis, and points us resolutely at methods of machine learning and empirical risk minimisation.

The contribution of this work is threefold. Firstly, discussion of the application of prognosis leads us to formulate a criterion which does not resort to a standard approach of classification, function approximation or maximum (partial) likelihood inference. Secondly, we point to the use of aggregation methods in a context of bio-informatics, give a subsequent algorithm (APTER) and derive a competitive performance guarantee. Thirdly, we present extensive empirical evidence which supports the theoretical insights, and affirms its use for the analysis of microarray data for survival analysis. The experiments can be reproduced using the software made public at `http://`.

1.1 Organization and Notation

This paper is organized as follows. The next section discusses the setting of survival analyses and the aim of prognosis. Section 3 describes and analyses the proposed algorithm. Section 4 gives empirical results of this algorithms on artificial and microarray datasets. Section 4 concludes with a number of open questions.

This paper follows the notational convention to represent deterministic single quantities as lower-case letters, vectors are denoted in bold-face, and random quantities are represented as upper-case letters. In this paper, the following notational conventions are used: random variable are denoted as capital letters $X, Y, Z, \ldots$. Vectors are denoted in boldface $\mathbf{x}, \mathbf{y}, \ldots$. Deterministic quantities are represented as lowercase letters $i, n, f, \ldots$. Expectation with respect to any random variable in the

[*]to whom correspondence should be addressed

expression is denoted as $\mathbb{E}$. The shorthand notation $\mathbb{E}_n[\cdot]$ denotes expectation with respect to all n samples seen thus far, while $\mathbb{E}_{n-1}[\cdot]$ denotes expectation with respect to the first $n-1$ samples. $\mathbb{E}^n[\cdot]$ denotes expectation with respect to the nth sample only, such that the rules of probability imply that $\mathbb{E}_n[\cdot] = \mathbb{E}_{n-1}\mathbb{E}^n[\cdot]$.

The data is represented as a set of size n of tuples

$$\{(\mathbf{x}_i, Y_i, \delta_i)\}_{i=1}^n, \tag{1}$$

Let $0 < Y_1 \leq Y_2 \leq \cdots \leq Y_n$ be an ordered sequence of observed event times associated to n subjects. An event can be either a failure with time T_i, or a left censoring C_i, expressed as the time elapse from t_0. In this paper we assume that all n subjects share the same time of origin t_0. It will be convenient to assume that each subject has a failure and left censoring time with values T_i and C_i respectively. Then only the minimum time can be observed, or $Y_i = \min(T_i, C_i)$. It will be convenient to define the *past event set* $P(t) \subset \{1, \ldots, n\}$ at time t. That is, $P(t)$ denotes the set of all subjects which have experienced an event strictly before time t. Let for $i = 1, \ldots, n$ the indicator $\delta_i \in \{0, 1\}$ denote wether a failure is observed ($\delta_i = 1$), or if the subject i is censored ($\delta_i = 0$), or $\delta_i = I(Y_i < C_i)$. Then

$$P(t) = \{i : \; Y_i < t, \delta_i = 1\}. \tag{2}$$

Furthermore, associate to each subject $i = 1, \ldots, n$ a covariate $\mathbf{x}_i \in \mathbb{R}^d$ of dimension d. In the present setting, $d = O(1000)$, while $n = O(100)$ at best.

2 PROGNOSIS IN SURVIVAL ANALYSIS

In this section we formalize the task of learning how to make a prognosis, based on observed cases. The general task of prognosis in survival analysis can then be phrased as follows:

DEFINITION 1 (Prognosis). *Given a subject with covariate* $\mathbf{x}_* \in \mathbb{R}^d$*, what can we say about the value of its associated* T_**?*

Motivated by the popular essay by S.J. Gould[1], we like to make statements as 'my covariates indicate that with high probability I will outlive 50% of the subjects suffering the same disease', or stated more humanely as 'my covariates indicate that I belong to the *good* half of the people having this disease'. The rationale is that this problem statement appears easier to infer than estimating the full conditional hazard or survival functions, while it is more informative than single median survival rates.

Now, we look for an *expert* $f : \mathbb{R}^d \to \mathbb{R}$ which can decide for any 2 different subjects $0 < i, j \leq n$ which one of them will *fail* first. In other words, we look for a f such that for as many couples (i, j) as possible, one has

$$(T_i - T_j)\left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right) \geq 0. \tag{3}$$

Since T_k is not observed in general due to censoring, the following (rescaled) proxy is used instead

$$\sum_{i=1}^n \frac{1}{|P(Y_i)|} \sum_{j \in P(Y_i)} I\left(f(\mathbf{x}_i) < f(\mathbf{x}_j)\right), \tag{4}$$

where $I(z) = 1$ if z holds true, and zero otherwise. In case $|P(Y_i)| = 0$, the ith summand in the sum is omitted. This is standard practice in all subsequent formulae. Note that this quantity is similar to the so called Concordance Index (C_n) as proposed by Harell (7). The purpose of this paper is to propose and analyze an algorithm for finding such f from a large set $\{f\}$ based on observations, under the requirements imposed by the specific setup.

If given one expert $f : \mathbb{R}^d \to \mathbb{R}$, its 'loss' of a prognosis of a subject with covariate $\mathbf{x}_* \in \mathbb{R}^d$, and time of event Y_* would be

$$\ell_*(f) = \frac{1}{|P(Y_*)|} \sum_{k \in P(Y_*)} I\left(f(\mathbf{x}_*) \leq f(\mathbf{x}_k)\right). \tag{5}$$

That is, $\ell_*(f)$ is the fraction of samples which experience an event before the time Y_* associated to the subject with the covariate $\mathbf{x}_*$, although they were prognosed with a higher score by expert f. Now we consider having m such experts $\{f_i\}_{i=1}^m$, and we will learn which of them performs best. We represent this using a vector $\mathbf{p} \in \mathbb{R}^m$ with $\mathbf{p}_i \geq 0$ for all $i = 1, \ldots, m$, and with $1_m^T \mathbf{p} = 1$. Then, we will use this *weighting* of the experts to make an informed prognosis of the event at T_* of a subject with covariate $\mathbf{x}_* \in \mathbb{R}^d$. Its associated loss is given as

$$\ell_*(\mathbf{p}) = \sum_{i=1}^m \mathbf{p}_i \left(\frac{1}{|P(T_*)|} \sum_{k \in P(T_*)} I\left(f_i(\mathbf{x}_*) \leq f_i(\mathbf{x}_k)\right) \right). \tag{6}$$

This represents basically which expert is assigned most value to for making a prognosis. For example, in lung-cancer we may expect that an expert based on smoking behaviour of a patient has a higher weight than an expert based of the psychology of the subject. Note that we include the $' =' $ case in (6) in order to avoid the trivial cases where f is constant. So, we have formalised the setting as learning such $\mathbf{p}$ in a way that the smallest possible loss $\ell_*(\mathbf{p})$ will be (or can be expected to be) made.

3 THE APTER ALGORITHM

When using a *fixed* vector $\hat{\mathbf{p}}$, we are interested in the *expected* loss of the rule given by $\hat{\mathbf{p}}$. Assume that $\hat{\mathbf{p}}$ is independent from the sample with index n, then the expected loss of a new sample $(\mathbf{x}_n, T_n)$ becomes $\mathbb{L}(\hat{\mathbf{p}}) = \mathbb{E}^n \ell_n(\hat{\mathbf{p}}) =$

$$\mathbb{E}^n \left[\sum_{i=1}^m \hat{\mathbf{p}}_i \frac{1}{|P(T_n)|} \sum_{k \in P(T_n)} I\left(f_i(\mathbf{x}_n) \leq f_i(\mathbf{x}_k)\right) \right]. \tag{7}$$

Note that bounds will be given for this quantity which are valid for any $\mathbf{x}_n \in \mathbb{R}^d$ which may be provided. In order to device a method which guarantees properties of this quantity, we use the mirror averaging algorithm as studied in A. Tsybakov, P. Rigollet, A. Juditsky in (8). This algorithm is based on ideas set out in (9). It is a highly interesting result of those authors that the resulting estimate has better properties in terms of oracle inequalities compared to techniques based on sample averages. Presently, this fast rates are not obtained since the involved loss functions are not exponentially concave as in (8), Definition 4.1. Instead of this property, we resort to use of Hoeffding's inequality which gives us a result with rate $O(\sqrt{\frac{\ln m}{n}})$.

[1] 'The Median Isn't the Message ' as in `http://www.prognosis.org/what_does_it_mean.php`

Algorithm 1 APTER: Aggregate Prognosis Through Exponential Reweighting

```
(0) Let $\mathbf{p}_i^0 = \frac{1}{m}$ for $i = 1, \ldots, m$.
for all $k = 1, \ldots, n$ do
    (1) The prognosis associated to the $m$ experts $\{f_i\}_{i=1}^m$ are scored whenever any new event (censored or not) is recorded for a subject $k \in \{1, \ldots, n\}$ at time $Y_k$ as
        $\ell_k(f_i) = \frac{1}{|P(Y_k)|} \sum_{l \in P(Y_k)} I\left(f_i(\mathbf{x}_k) \leq f_i(\mathbf{x}_l)\right)$    (8)
    and the cumulative loss is $L_k(f_i) = \sum_{s=1}^k \ell_k(f_i)$.
    (2) The vector $\mathbf{p}^k$ is computed for $i = 1, \ldots, m$ as follows
        $\mathbf{p}_i^k = \frac{\exp(-\nu L_k(f_i))}{\sum_{j=1}^m \exp(-\nu L_k(f_j))}$.    (9)
end for
(3) Aggregate the hypothesis $\{\mathbf{p}^k\}_k$ into $\hat{\mathbf{p}}$ as follows:
    $\hat{\mathbf{p}} = \frac{1}{n} \sum_{k=0}^{n-1} \mathbf{p}^k$.    (10)
```

This algorithm comes with the following guarantee. We need the following property

Definition 2. *For any $t = 1, \ldots, n$ and $i = 1, \ldots, m$ we have that*

$$\mathbb{E}_n\left[g\left(\frac{L_n(f_i)}{n}\right)\right] = \mathbb{E}_n[g(\ell_t(f_i))]. \tag{11}$$

for any regular function $g : \mathbb{R} \to \mathbb{R}$. Equivalently, we have that $\mathbb{E}_n g(\cdot)$ denotes any distributional property of the random variable at hand.

This essentially means that we do not expect the loss to be different when it is measured at different points in time. Thus:

Theorem 1 (APTER). *Given m experts $\{f_i\}_{i=1}^m$, and the loss function ℓ as defined in eq. (6). Then run the APTER algorithm with $\nu = \sqrt{\frac{2\ln m}{n}}$ resulting in $\hat{\mathbf{p}}$. Then*

$$\mathbb{E}_{n-1}\left[\mathbb{L}(\hat{\mathbf{p}}) - \min_{i=1,\ldots,m} \mathbb{L}(f_i)\right] \leq \sqrt{\frac{2\ln m}{n}}. \tag{12}$$

This result is in some way surprising. It says that we can get competitive performance guarantees without a need for optimizing the performance over a set of hypothesis. Note that an optimization formulation lies on the basis of a maximum (partial) likelihood method or a risk minimization technique as commonly employed in a machine learning setting. There is an implicit link with optimization and aggregation through the method of mirror descent, see e.g. (10) and (11). The lack of an explicit optimization stage results in the considerable computational speedups.

Note that the performance guarantee degrades only as $\sqrt{\log(m)}$ in terms of the number of experts m.

3.1 Choice of Experts and APTER$_p$

The following experts are used in the application in microarray case studies. Here, we use simple univariate rules. That is, the experts are based on individual features (gene expression levels) of the dataset. The rationale is that a single gene expression might be responsible for the observed behaviours.

Let $\mathbf{e}_i$ be the ith unit vector, and let ± denote both the positive as well as the negated version. Then, the experts $\{f_i\}$ are computed as

$$f_i(\mathbf{x}) = \pm\mathbf{e}_i^T\mathbf{x},$$

so that $m = 2d$, and every gene expression level can both be used for *over-expression* or *under-expression.*

In practice however, evidence is found that the following features work even better:

$$f_i(\mathbf{x}) = s_i\mathbf{e}_i^T\mathbf{x},$$

where the sign $s_i \in \{-1, 1\}$ is given by wether the ith expression has a concordance index with the observed outcome larger or equal to 0.5, as estimated on the set used for training. This means that $m = d$. This technique is referred to as APTER$_p$.

Note that this task is also addresses the application of Boosting methods. There, a popular choice is the use of random trees as in (12).

3.2 Preprocessing using SIS and ISIS

It is found empirically that preprocessing using ISIS as described in (13) improves the numerical results. However, the rational for this technique comes from a different angle. That is, it is conceived as a screening technique for PH-L$_1$-type of algorithms. Let $\mathbf{m} = (\mathbf{m}_1, \ldots \mathbf{m}_d)^T \in \mathbb{R}^d$ be defined as

$$\mathbf{m} = \sum_{i=1}^n Y_i\mathbf{x}_i. \tag{13}$$

For any given $\gamma \in (0, 1)$. Here, $[\gamma n]$ denotes the integer part of γn. We define the set M_γ as (13):

$$M_\gamma = \left\{1 \leq i \leq d : |\mathbf{m}_i| \text{ is among the first } [\gamma n] \text{ largest entries of } \mathbf{m}\right\}. \tag{14}$$

This set then gives the indices of the features which are retained in the further analysis. It is referred to as Sure Independence Screening (SIS) (13). In the second step, APTER is applied using only the retained features. Note that in the paper (13), one suggests the use of a SCAD penalty for Cox partial Likelihood approach.

An extension of SIS is an Iterative SIS (ISIS), see (13). The idea is to pick up important features, missed by SIS. This goes as follows, rather than having a single preprocessing (SIS) step, the procedure is repeated as follows. At the end of a SIS-APTER step, a new response Y' can be computed by application of the found regression coefficients. This new response variables can then be reused in a SIS step, resulting in fresh $[\gamma n]$ features. This procedure is repeated until one has enough *distinct* features.

Since $[\gamma n]$ features are then given as input to the actual training procedure, we will refer to this value as m in the experiments, making this connection between screening and training more explicit.

4 EMPIRICAL RESULTS

This section present empirical results supporting the claim of efficiency. First, we describe the setup of the experiments. The description of the real-world datasets is delayed to the Appendix.

4.1 Setup

The following measure of quality (the Concordance index or C_n, see e.g (14)) of a prognostic index scored by the function $f : \mathbb{R}^d \to \mathbb{R}$ is used. Let the data be denoted as $\{(\mathbf{x}_i, Y_i, \delta_i)\}_{i=1}^n$, where $\mathbf{x}_i$ are the covariates, Y_i contains the survival time and δ_i is the censoring indicator as before. Consider any $f : \mathbb{R}^d \to \mathbb{R}$, then the C_n is defined as

$$C_n(f) = \frac{\sum_{i:\delta_i=0} \sum_{Y_j > Y_i} I(f(\mathbf{x}_i) < f(\mathbf{x}_j))}{|\varepsilon|}. \tag{15}$$

Here $|\varepsilon|$ denotes the number of the pairs which have $Y_i < Y_j$ when Y_i is not censored. The indicator function $I(\pi) = 1$ if π holds, and 0 otherwise. That is, if $C_n(f) = 1$, one has that f scores a higher prognostic index to the subject with will experience the event later ('good'). A $C_n(f) = 0.5$ says that the prognostic index given by f is arbitrary with respect to event times ('bad'). Observe that this measure is not quite the same as $\ell_n(f)$ and $L_n(f)$ as used in the design of the APTER algorithm. Note that this function goes along the lines of the Area under the ROC curve or the Mann-Whitney statistic, adapted to handling censored data.

The data is assigned randomly to training data of size $n_t = \lfloor 2n/3 \rfloor$ and test data of size $n - n_t$. The training data is used to follow the training procedures, resulting in $\hat{f}$. The test data is used to compute the performance expressed as $C_n(\hat{f})$. The results are randomised 50 times (i.e. a random assignments to training and test set), and we report the median value as well as ± the variance.

All datasets are observational, implying that there is no need for the application of the online version, so will be concerned here only with the APTER algorithm. The parameter $\nu > 0$ is tuned in the experiments using cross-validation on the dataset which is used for training. It was found that proper tuning of this parameter is crucial for achieving good performance.

The following ten algorithms are run on each of these datasets:

(a) APTER: The approach as given in Alg. 1 where experts $\{f_i, f_i'\}$ are taken as $f_i(\mathbf{x}) = \mathbf{e}_i^T\mathbf{x}$ and $f_i'(\mathbf{x}) = -\mathbf{e}_i^T\mathbf{x}$. In this way we can incorporate positive effects due to over-expression and under-expression of a gene. This means that $m = 2d$.

(b) APTER$_p$: The approach as given in 1 where experts $\{f_i\}$ are given as $f_i(\mathbf{x}) = s_i\mathbf{e}_i^T\mathbf{x}$ where the sign $s_i \in \{-1, 1\}$ is given by the C_n of the ith expression with the observed effect, estimated on the set used for training. This means that $m = d$.

(c) MINLIP$_p$: The approach based on ERM and s_i as discussed in (15).

(d) MODEL2: Another approach based on ERM as discussed in (15).

(g) PLS: An approach based on preprocessing the data using PLS and application of Cox regression, as described in (2).

(f) PH-L$_1$: An approach based on a L_1 penalized version of Cox regression, as described in (5).

(g) PH-L$_2$: An approach based on a L_2 penalized version of Cox regression, as described in (16).

(h) ISIS-APTER$_p$: An approach which uses ISIS as preprocessing, and applies APTER$_p$ on the resulting features (13).

(i) ISIS-SCAD: An approach which uses ISIS as preprocessing, and applies SCAD on the resulting features (13).

(j) Rankboost: An approach based on boosting the c-index (17).

Those algorithms are applied to an artificial dataset (as described below) as well as on a host of real-world datasets (as described in Appendix 2).

4.2 Artificial Data

The technique is tested on artificial data which was generated as follows. A disjunct training set and test set, both of $n = 100$ 'patients' was generated. For each 'patient', $d = m$ features are sampled randomly from a standard distribution, so that $\mathbf{x}_i \in \mathbb{R}^d$.

We say that we have only k *informative* features. Then, a time T_i of a corresponding event is computed for $i = 1, \dots, n$ as

$$T_i = \frac{-\log Z_i}{10 \exp\left(\sum_{j=1}^k \mathbf{x}_{i,j}\right)} \tag{16}$$

where Z_i is a random value generated from a uniform distribution on the unit interval $]0, 1[$, and $\mathbf{x}_{i,j}$ is the j-th covariate for the ith patient. The the right-censoring time is randomly generated from the exponential distribution with rate 0.10. After application of the censoring rule to the event time T_i, we arrive at the *survival* time Y_i.

In a first experiment, $d = m$ is fixed as 100, but only the first $k \leq d$ features have an effect on the outcome. Figure (1.a) shows the evolution of the performance ($C_n(\hat{f})$) for increasing values of k. In a second experiment we fix $k = 10$, and record the performance for increasing values of $d = m$, investigating the effect of a growing number of *ambient* dimension on the performance of APTER. Results are displayed in Figure (1.b).

Thirdly, we investigate how well the numerical results align with the result of Theorem 1. The results are given in Figure (2). The "c-index error" (Cerr) is given for different values of m and n. Cerr is computed as the difference between the C_n obtained by APTER - denoted as $\hat{f}$ - and the C_n of the single "best" expert $f_j(\mathbf{x}_i) = \mathbf{x}_{i,j}$:

$$C\text{err} = \max_j C_n(f_j) - C_n\left(\hat{f}\right). \tag{17}$$

This formula is similar to equation (12). This figure indicates that Cerr increases logarithmically in m, and in terms of $\frac{1}{\sqrt{n}}$. This supports the result of Theorem 1.

4.3 Real Datasets

In order to benchmark APTER and its variations against state-of-the-art approaches, we run the algorithms as well on a wide range of large-dimensional real datasets. Those datasets are publicly available, and all experiments can be reproduced using the code available at [2]. The dataset are collected in a context of bioinformatics, and a full description of this data can be found in Appendix 2. The experiments are divided in three categories:

[2] `http://...`

	NSBCD (115×549)	DBCD (295×4919)	DLBCD (240×7399)	Veer (78×4751)	Vijver (295×70)	Beer (86×7129)	AML (116×6283)
APTER	0.73±0.10	0.69±0.06	0.58±0.04	0.65±0.10	0.44±0.06	0.60±0.13	0.58±0.05
APTER$_p$	0.77±0.05	0.74±0.04	0.59±0.03	**0.68±0.08**	0.62±0.04	0.73±0.08	0.60±0.05
MINLIP$_p$	0.74±0.05	0.71±0.04	0.59±0.04	0.65±0.10	0.61±0.06	0.69±0.09	0.55±0.07
MODEL2	0.75±0.04	0.74±0.04	0.62±0.03	**0.67±0.09**	0.61±0.06	**0.74±0.08**	0.56±0.06
PLS	**0.78±0.05**	0.74±0.03	0.53±0.05	0.58±0.10	0.62±0.07	0.66±0.12	0.57±0.06
PH-L2	0.69±0.07	0.73±0.04	**0.65±0.04**	0.64±0.08	0.61±0.08	0.73±0.08	0.54±0.06
PH-L1	0.69±0.06	0.74±0.04	0.60±0.04	0.60±0.06	**0.65±0.06**	0.69±0.02	0.61±0.06
Rankboost	0.75±0.04	0.72±0.03	0.62±0.02	0.62±0.02	**0.65±0.02**	0.71±0.02	0.53±0.01
ISIS-SCAD	0.69±0.04	0.72±0.04	**0.65±0.07**	**0.68±0.04**	0.62±0.02	0.72±0.04	**0.63±0.02**
ISIS-APTER$_p$	**0.78±0.06**	**0.76±0.08**	0.62±0.07	0.66±0.05	0.62±0.06	**0.75±0.09**	0.59±0.05

Table 1. Numerical results of the experiments of 10 different methods on 7 microarray datasets.

- The algorithms are run on seven microarray datasets, in order to asses performance on typical sizes for those datasets. Here we see that there is no clear overall winner amongst the algorithms, but the proposed algorithm (ISIS-APTER$_p$) does do repeatedly very well, and performs best on most (3) datasets. Results are given in Table (1).

  In order to see wether the positive performance is not due to irregularities of the data, we consider the following *null* experiment. Consider the AML dataset, but lets shuffle the observed phenotypes (the observed Y) between different subjects. So any relation between the expression level and the random phenotype must be due to plain chance (by construction). We see in graph h that indeed the distribution of the methods based on this *shuffled* data nears a neutral C_n on the test set of 0.50. This means that the 10% improvement as found in the real experiment (graph g) is substantial with respect to the randomizations, and not due to chance alone.

- The results of the algorithm is compared on the micro-array dataset as reported in (18), and analysed further in (19). Here we found that the obtained performance is significantly larger than what was reported earlier, while we do not have to resort to the *clustering* preprocessing as advocated in (18; 19). This data has a very high dimensionality ($d = 44.928$) and has only a few cases ($n = 191$). Results are given in Table (2) and the box plots of the performances due to the 50 randomisations, are given in Figure (3).

  Finally, we discuss the application of the method on the same high-dimensional ($d = 44.928$) dataset as before, but we study the impact of the parameter m given to ISIS, which returns in turn the data to be processed by APTER. The performances for different values of m are given in Fig. (4.a). The best performance is achieved for $m = 800$, which is the value which was used in the earlier experiment reported in Fig. (3). Here we compare only to a few other approaches, namely the PH-L$_1$, MINLIP$_p$ and MODEL2 approach which are either optimisation-based. Panel (4.b) reports the time needed to perform training/ tuning and randomisation corresponding to a fixed value of m. Panel (4.c) reports the size of the memory used up for the same procedure. Here it is clearly seen that APTER$_p$ results in surprisingly good performance, given that it uses up less computations and memory. It is even so that the optimisation-based techniques cannot finish for large m in reasonable time or without the problems of the memory management, despite the fact that a very efficient optimisation solver (Yalmip) was used to implement those.

Dataset	Method	$C_n(\hat{f})$
FL (191×44.928)	APTER	0.70±0.05
	APTER$_p$	0.73±0.04
	MINLIP$_p$	0.70±0.03
	MODEL2	0.72±0.04
	PLS	0.66±0.03
	PH-L$_2$	0.69±0.07
	PH-L$_1$	0.67±0.05
	RankBoost	0.67±0.03
	ISIS-SCAD	0.71±0.03
	ISIS-APTER$_p$	**0.74±0.05**
	Dave's Method (see (19))	0.71±0.02

Table 2. Numerical results of the experiments of 10 different methods on the Follicular Lymphoma dataset.

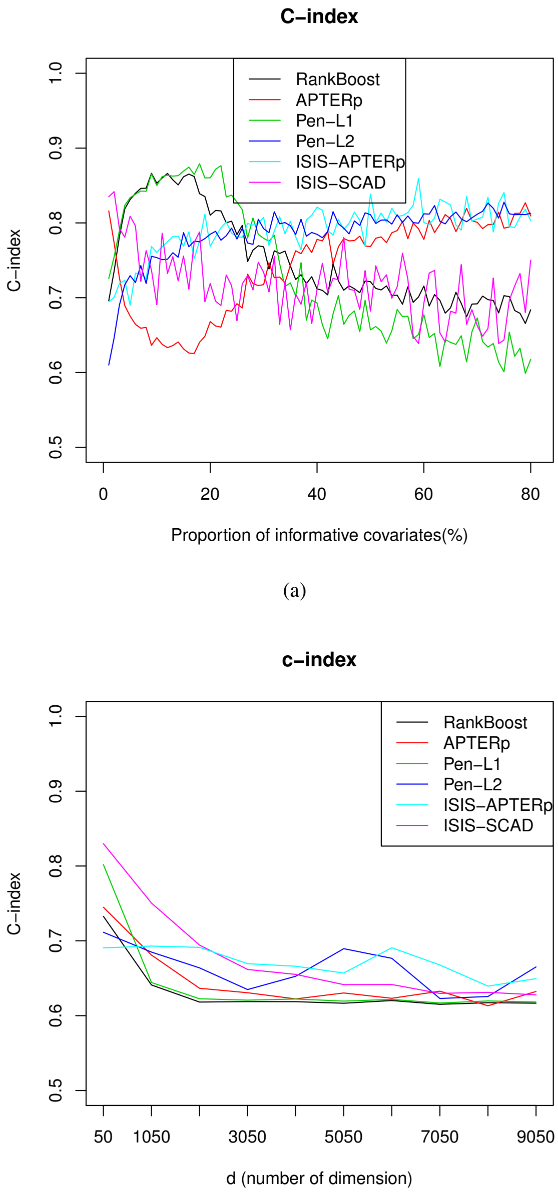
4.4 Discussion of the Results

This results uncover some interesting properties of the application of the proposed algorithms in this bio-informatics setting.

First of all, the APTER and APTER$_p$ method is orders of magnitudes faster (computationally) compared to the bulk of methods based on optimization formulations (either using Maximum (penalized) Partial Likelihood, Empirical Risk Minimization or multivariate preprocessing techniques). This does not affect the performance in any way, contrary to what intuition would suggest. In fact, the performance on typical micro-array data of the *vanilla* APTER or APTER$_p$ (without ISIS) is among the best ones.

Secondly, inclusion of preprocessing with ISIS - also very attractive from a computational perspective - is boosting up significantly the performance of APTER. We have no theoretical explanation for this, since ISIS was designed to complement L_1 or Danzig-selector approaches. While the authors of ISIS advocate the used of a SCAD norm, we find that APTER$_p$ is overall a better choice for the mentioned datasets.

Furthermore, the empirical results indicate that the statistical performance is preserved by using APTER$_p$ combined with ISIS,

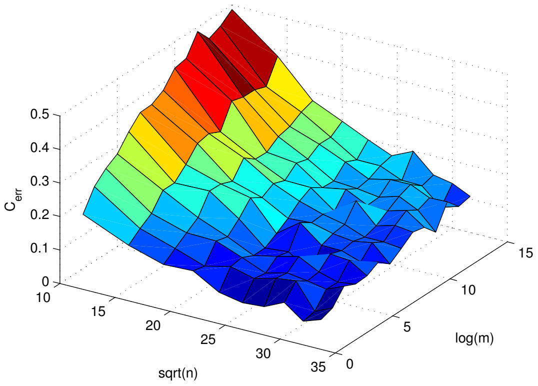


(a)

(b)

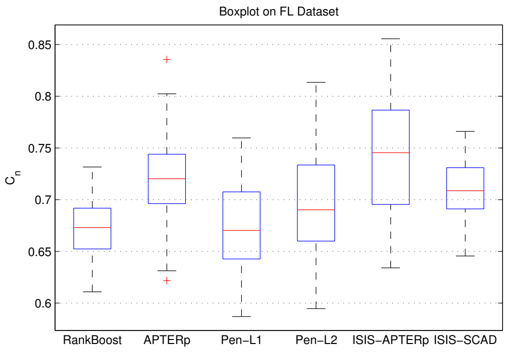
Fig. 1. Comparison of the numerical results obtained on the artificial data sets (a) when keeping $d = 100$ fixed, and (b) when keeping $k = 10$ fixed.

and may even improve over performances obtained using existing approaches. This is remarkable since the computational power is orders of magnitude smaller than most existing approaches based on (penalised) PL of ERM. We find also that empirical results align quite closely the theoretical findings as illustrated with an experiment on artificial data.



(a)

Fig. 2. The evolution of the 'C-index error' C_{err} obtained by APTER$_p$ for different values of (n, m).

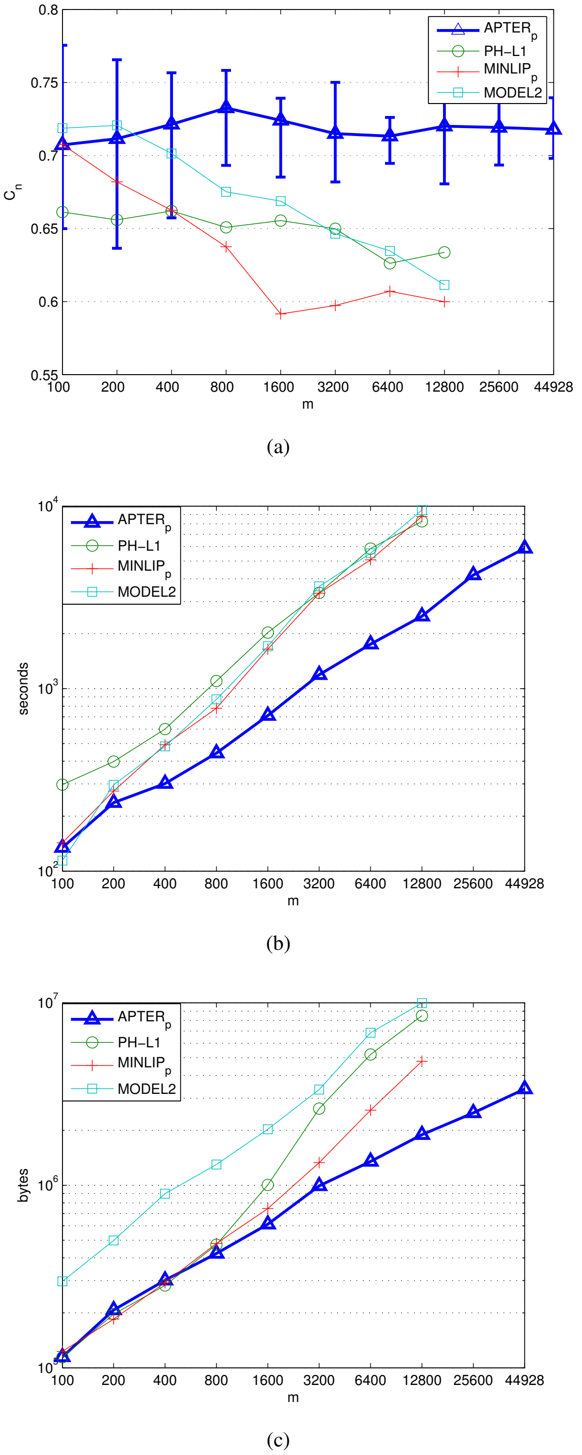


(a)

Fig. 3. Boxplots of the numerical results obtained on the FL dataset. Results are expressed in terms of the $C_n(f)$ on a test set, where f is trained and tuned on a disjunct training set. The boxplots are obtained using 50 randomizations of the split training-testset.

5 CONCLUSIONS

This paper presents statistically and computationally compelling results that a method based on online learning and aggregation can be used for analysis of survival data in high dimensions. Theoretical findings are complemented with empirical results on micro-array dataset. We feel that this result is surprising not only that it outperforms methods in ERM or (penalised) PL, but provides as well a tool with much lower computational complexity as the former ones since no direct optimization is involved. The wide host of empirical, reproducible results support the claim of efficiency. This analysis presents many new opportunities, both applied (towards GWAs) as theoretical (can we improve the rates of convergence by choosing other loss functions?).



(a)

(b)

(c)

Fig. 4. results of the choice of m in ISIS, based on the Follicular Lymphoma dataset(18; 19). (a) Performance expressed as $C_n(\hat{f})$ on the test sets (medium of 50 randomizations). (b) Computation time for running tuning, training and randomisation for a fixed value of m. (c) Usage of memory of the same procedure.

REFERENCES

[1] J. D. Kalbfleisch and R. L. Prentice, "The statistical analysis of failure time data," 2011.

[2] H. M. Bøvelstad, S. Nygård, H. L. Størvold, M. Aldrin, Ø. Borgan, A. Frigessi, and O. C. Lingjærde, "Predicting survival from microarray datała comparative study," *Bioinformatics*, vol. 23, no. 16, pp. 2080–2087, 2007.

[3] L. J. van't Veer, H. Dai, M. J. Van De Vijver, Y. D. He, A. A. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen *et al.*, "Gene expression profiling predicts clinical outcome of breast cancer," *Nature*, vol. 415, no. 6871, pp. 530–536, 2002.

[4] V. Van Belle, K. Pelckmans, J. A. Suykens, and S. Van Huffel, "Learning transformation models for ranking and survival analysis." *Journal of machine learning research*, vol. 12, no. 3, 2011.

[5] J. J. Goeman, "L1 penalized estimation in the cox proportional hazards model," *Biometrical Journal*, vol. 52, no. 1, pp. 70–84, 2010.

[6] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for coxs proportional hazards model via coordinate descent," *Journal of Statistical Software*, vol. 39, no. 5, pp. 1–13, 2011.

[7] M. Gönen and G. Heller, "Concordance probability and discriminatory power in proportional hazards regression," *Biometrika*, vol. 92, no. 4, pp. 965–970, 2005.

[8] A. Juditsky, P. Rigollet, A. B. Tsybakov *et al.*, "Learning by mirror averaging," *The Annals of Statistics*, vol. 36, no. 5, pp. 2183–2206, 2008.

[9] A. Nemirovsky and D. Yudin, "Problem complexity and method efficiency in optimization," 1983.

[10] C. Jakobson, M. Feinsod, and Y. Nemirovsky, "Low frequency noise and drift in ion sensitive field effect transistors," *Sensors and Actuators B: Chemical*, vol. 68, no. 1, pp. 134–139, 2000.

[11] P. J. Bickel, B. Li, A. B. Tsybakov, S. A. van de Geer, B. Yu, T. Valdés, C. Rivero, J. Fan, and A. van der Vaart, "Regularization in statistics," *Test*, vol. 15, no. 2, pp. 271–344, 2006.

[12] V. Van Belle, K. Pelckmans, J. A. Suykens, and S. Van Huffel, "Feature selection in survival least squares support vector machines with maximal variation constraints," *Bio-Inspired Systems: Computational and Ambient Intelligence*, pp. 65–72, 2009.

[13] J. Fan and J. Lv, "Sure independence screening for ultrahigh dimensional feature space," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 5, pp. 849–911, 2008.

[14] V. C. Raykar, H. Steck, B. Krishnapuram, C. Dehing-Oberije, and P. Lambin, "On ranking in survival analysis: Bounds on the concordance index." in *NIPS*, 2007.

[15] V. Van Belle, K. Pelckmans, S. Van Huffel, and J. A. Suykens, "Improved performance on high-dimensional survival data by application of survival-svm," *Bioinformatics*, vol. 27, no. 1, pp. 87–94, 2011.

[16] J. Goeman, "Penalized: L1 (lasso) and l2 (ridge) penalized estimation in glms and in the cox model," *R package version 09-21 2008*, 2008.

[17] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences," *The Journal of machine learning research*, vol. 4, pp. 933–969, 2003.

[18] S. S. Dave, G. Wright, B. Tan, A. Rosenwald, R. D. Gascoyne, W. C. Chan, R. I. Fisher, R. M. Braziel, L. M. Rimsza, T. M. Grogan *et al.*, "Prediction of survival in follicular lymphoma based on molecular features of tumor-infiltrating immune cells," *New England Journal of Medicine*, vol. 351, no. 21, pp. 2159–2169, 2004.

[19] R. S. Lin, "Re-analysis of molecular features in predicting survival in follicular lymphoma," *report, http://statweb.stanford.edu/~tibs/FL/report/RayLin_Lab_rotation2006s.pdf*, 2006.

[20] M. J. Van De Vijver, Y. D. He, L. J. van't Veer, H. Dai, A. A. Hart, D. W. Voskuil, G. J. Schreiber, J. L. Peterse, C. Roberts, M. J. Marton *et al.*, "A gene-expression signature as a predictor of survival in breast cancer," *New England Journal of Medicine*, vol. 347, no. 25, pp. 1999–2009, 2002.

[21] T. Sørlie, R. Tibshirani, J. Parker, T. Hastie, J. Marron, A. Nobel, S. Deng, H. Johnsen, R. Pesich, S. Geisler *et al.*, "Repeated observation of breast tumor subtypes in independent gene expression data sets," *Proceedings of the National Academy of Sciences*, vol. 100, no. 14, pp. 8418–8423, 2003.

[22] H. C. van Houwelingen, T. Bruinsma, A. A. Hart, L. J. van't Veer, and L. F. Wessels, "Cross-validated cox regression on microarray gene expression data," *Statistics in medicine*, vol. 25, no. 18, pp. 3201–3216, 2006.

[23] A. Rosenwald, G. Wright, W. C. Chan, J. M. Connors, E. Campo, R. I. Fisher, R. D. Gascoyne, H. K. Muller-Hermelink, E. B. Smeland, J. M. Giltnane *et al.*, "The use of molecular profiling to predict survival after chemotherapy for diffuse large-b-cell lymphoma," *New England Journal of Medicine*, vol. 346, no. 25, pp. 1937–1947, 2002.

[24]D. G. Beer, S. L. Kardia, C.-C. Huang, T. J. Giordano, A. M. Levin, D. E. Misek, L. Lin, G. Chen, T. G. Gharib, D. G. Thomas *et al.*, "Gene-expression profiles predict survival of patients with lung adenocarcinoma," *Nature medicine*, vol. 8, no. 8, pp. 816–824, 2002.

[25]L. Bullinger, K. Döhner, E. Bair, S. Fröhling, R. F. Schlenk, R. Tibshirani, H. Döhner, and J. R. Pollack, "Use of gene-expression profiling to identify prognostic subclasses in adult acute myeloid leukemia," *New England Journal of Medicine*, vol. 350, no. 16, pp. 1605–1616, 2004.

1 PROOF OF THEOREM 1

The following results will be used.

LEMMA 1 (Hoeffding). *Let $\lambda \in \mathbb{R}$, and let X be a random variable taking values in $[a, b] \subset \mathbb{R}$, then*

$$\ln \mathbb{E}\left[\exp(\lambda X)\right] \leq \lambda \mathbb{E}[X] + \frac{\lambda^2 (b-a)^2}{8}. \tag{18}$$

With assumption of eq. (11) in hand, the following result holds:

LEMMA 2. *Given m experts $\{f_i : \mathbb{R}_d \to \mathbb{R}\}_{i=1}^m$, a loss function $\ell : \mathbb{R} \to \mathbb{R}$ satisfying eq. (11), and let $\{(\mathbf{x}_k, Y_k, \delta_k)\}_{k=1}^n$ take values in $\mathbb{R}^d \times \mathbb{R}_0 \times \{0, 1\}$. Let the APTER algorithm (1) be run with $\nu > 0$, then*

$$\mathbb{E}_{n-1}\left[\mathbb{L}(\hat{\mathbf{p}}) - \min_{i=1,\ldots,m} \mathbb{L}(f_i)\right] \leq \frac{\ln m}{\nu n} + \frac{1}{\nu}\mathbb{E}_n[R_n], \tag{19}$$

with

$$R_n = \frac{1}{\nu}\sum_{t=1}^n \ln \hat{E} \exp -\nu\left(\ell_n(f) - \hat{E}\ell_n(f)\right). \tag{20}$$

and $\hat{E}g(f) = \sum_{i=1}^m \hat{\mathbf{p}}_i g(f_i)$ for any g.

PROOF. Consider the evolution of the normalization terms W_t where

$$W_t = \sum_{i=1}^m \exp -\nu L_t(f_i), \tag{21}$$

is characterized. Specifically, we see that

$$\begin{aligned}\ln \frac{W_n}{W_0} &= \ln \sum_{i=1}^m \exp(-\nu L_n(f_i)) - \ln m \\ &\geq -\nu \min_{i=1,\ldots,m} L_n(f_i) - \ln m, \end{aligned} \tag{22}$$

as before. Hence

$$\begin{aligned}\frac{1}{n\nu}\mathbb{E}_n\left[\ln \frac{W_n}{W_0}\right] &\geq - \min_{i=1,\ldots,m} \mathbb{E}_n\left[\frac{1}{n}L_n(f_i)\right] - \frac{\ln m}{n\nu} \\ &\geq - \min_{i=1,\ldots,m} \mathbb{E}_n\left[\ell_n(f_i)\right] - \frac{\ln m}{n\nu} \\ &\geq - \min_{i=1,\ldots,m} \mathbb{E}_{n-1}\mathbb{L}(f_i) - \frac{\ln m}{n\nu}. \end{aligned} \tag{23}$$

On the other hand we have that

$$\begin{aligned}\ln \frac{W_t}{W_{t-1}} &= \ln \frac{\sum_{i=1}^m \exp(-\nu L_t(f_i))}{\sum_{j=1}^m \exp(-\nu L_{t-1}(f_j))} \\ &= \ln \sum_{i=1}^m \mathbf{p}_i^{t-1}\left(\exp -\nu \ell_t(f_i)\right). \end{aligned} \tag{24}$$

Taking expectation over the n samples (denoted as $\mathbb{E}_n[\cdot]$) seen thus far, and summarizing over $t = 1, \ldots, n$ gives

$$\begin{aligned}&\frac{1}{n\nu}\sum_{t=1}^n \mathbb{E}_n\left[\ln W_t - \ln W_{t-1}\right] \\ &\quad= \frac{1}{n\nu}\sum_{t=1}^n \mathbb{E}_n\left[\ln \sum_{i=1}^m \mathbf{p}_i^{t-1}\exp -\nu\ell_t(f_i)\right] \\ &\quad= \frac{1}{n\nu}\sum_{t=1}^n \mathbb{E}_n\left[\ln \sum_{i=1}^m \mathbf{p}_i^{t-1}\exp -\nu\frac{L_n(f_i)}{n}\right] \\ &\quad= \frac{1}{n\nu}\sum_{t=1}^n \mathbb{E}_n\left[\ln \sum_{i=1}^m \mathbf{p}_i^{t-1}\exp -\nu\ell_n(f_i)\right] \\ &\quad\leq \frac{1}{\nu}\mathbb{E}_n\left[\ln \sum_{i=1}^m \hat{\mathbf{p}}_i \exp -\nu\ell_n(f_i)\right], \end{aligned} \tag{25}$$

where the last inequality follows from Jenssen's inequality, and from the formula of aggregation as in eq. (10). Now, this gives

$$\begin{aligned}&\frac{1}{\nu}\mathbb{E}_n\left[\ln \hat{E}\exp -\nu\ell_n(f)\right] \\ &\quad= \frac{1}{\nu}\mathbb{E}_n\left[\ln \hat{E}\exp -\nu\hat{E}\ell_n(f)\right] \\ &\quad+ \frac{1}{\nu}\mathbb{E}_n\left[\ln \hat{E}\exp -\nu\left(\ell_n(f) - \hat{E}\ell_n(f)\right)\right] \\ &\quad= -\mathbb{E}_{n-1}\mathbb{E}^n[\hat{E}\ell_n(f)] \\ &\quad+ \frac{1}{\nu}\mathbb{E}_n\left[\ln \hat{E}\exp -\nu\left(\ell_n(f) - \hat{E}\ell_n(f)\right)\right], \end{aligned} \tag{26}$$

where we defined for notational convenience $\hat{E}\mathbf{x} = \sum_{i=1}^m \hat{\mathbf{p}}_i \mathbf{x}_i$ for all $\mathbf{x} \in \mathbb{R}^m$, and $\hat{E}\ell_n(f) = \sum_{i=1}^m \hat{\mathbf{p}}_i \ell_n(f_i)$. Combining inequalities (23) and (26) gives

$$\mathbb{E}_{n-1}\left[\mathbb{L}(\hat{\mathbf{p}}) - \min_{i=1,\ldots,m} \mathbb{L}(f_i)\right] \leq \frac{\ln m}{\nu n} + \frac{1}{\nu}\mathbb{E}_n[R_n], \tag{27}$$

as desired. □

So we are left to proof that the term $\mathbb{E}_n[R_n]$ is bounded in our case. The proof of Theorem 1 is then given as follows.

PROOF. This follows by application of Hoeffding's inequality as in eq. (18) since

$$R_n = \ln \hat{E}\exp -\nu\left(\ell_n(f) - \hat{E}\ell_n(f)\right) \leq \frac{\nu^2}{2}, \tag{28}$$

where we use that $0 \leq \ell_n \leq 1$. Then combining with eq. (19) gives the result. □

2 BENCHMARK DATASETS

This appendix describes the real-world datasets. The datasets range from large-dimensional ($d = O(100)$) to huge-dimensional ($O(10,000)$) and record $n = O(100)$ subjects. We report the performance of different methods on:

- 7 public datasets containing micro-array expression levels and events (occurrence of disease) of the associated subjects as used in (2).
- The micro-array survival dataset as presented in (18) and analysed in the report (19).

Details are given below.

The 7 publicly available microarray datasets as used for benchmarking in (2), have the following properties.

(NSBCD): The Norway/Stanford Breast Cancer Data set is given in (21). In this database there are survival data of $n = 115$ women who have breast cancer, and $d = 549$ intrinsic genes introduced in (21) were measured. In the 115 patients, 33% (38) have experienced an event during the study. Missing values were imputed by the 10-nearest neighbour method.

(Veer): The survival data of sporadic lymph-node-negative patients with their gene expression profiles is given in (3). It has $n = 78$ patients with $d = 4751$ gene expressions selected from the 25,000 genes recorded with the microarray. 44 patients remained free of disease after their diagnosis for an interval of at least 5 years. The average follow-up time for these patients was 8.7 years. 34 patients had developed distant metastases within 5 years, and the mean time to metastases was 2.5 years.

(Vijver): The data set of $n = 295$ consecutive patients with primary breast carcinomas is from (3) All patients had stage I or II breast cancer and were younger than 53 years old. They gave the previously determined $d = 70$ marker genes that are associated with the risk of early distant metastases in young patients with lymph-node-negative breast cancer. The median follow-up among all 295 patients was 6.7 years (range, 0.05 to 18.3). There were no missing data. 88 patients have experienced an event during the study.

(DBCD): The Dutch Breast Cancer Data set is described in (22), and is a subset of the data from (3). There are survival data of $n = 295$ women who have breast cancer. The measures of $d = 4919$ gene expression were taken from the fresh-frozen-tissue bank of the Netherlands Cancer Institute. All the ages of the patients are smaller than or equal to 52 years. The diagnosis was made between 1984 and 1995 without previous history of cancer. The median of follow-up time was 6.7 years (range 0.05-18.3). In the 295 patients, 26.78% (79) have experienced an event during the study.

(DLBCL): The diffuse large-B-cell lymphoma data set is described in (23). This contains survival data of $n = 240$ patients who have diffuse large-B-cell lymphoma. $d = 7399$ different gene expression measurements are given. The median of follow-up time was 2.8 years. From the 240 patients, 58% have experienced an event during the study.

(Beer): The survival data of $n = 86$ patients with primary lung adenocarcinomas is from (24) There are $d = 7129$ expressed genes selected from Affymetrix hu6800 microarrays. 76 patients have experienced an event during the study.

(AML): The survival data of acute myeloid leukemia patients is described in (25). It contains $n = 116$ patients with acute myeloid leukemia and the expression levels of $d = 6283$ genes. 71 patients have experienced an event during the study.

The same datasets were used in (2) and (4) to benchmark a state-of-art methods, results that are reproduced here as well. The high-dimensional FL dataset has the following description.

(FL): Additionally, we use the micro-array dataset which was used in (18), and analysed in (19). This data set included the survival data of $n = 191$ patients with follicular lymphoma after diagnosis. The median age at diagnosis was 51 years (range, 23 to 81), and the median follow-up time was 6.6 years (range, less than 1.0 to 28.2); the median followup time among patients alive at last follow-up was 8.1 years. It contains $d = 44928$ gene expression levels selected from Affymetrix U133A and U133B microarrays.