Feature Selection for Survival Analysis in Bioinformatics

Charles Englebert[1,2], Thomas Quinn[3], Isabelle Bichindaritz[1]
[1] *State University of New York, Oswego, NY, USA*
[2] *Supelec, Metz, France*
[3] *Deakin University, Geelong, Australia*

Abstract: The development of microarray technology has made it possible to assemble biomedical datasets that measure the expression profile of thousands of genes simultaneously. However, such high-dimensional datasets make computation costly and can complicate the interpretation of a predictive model. To address this, feature selection methods are used to extract biological information from a large amount of data in order to filter the expression dataset down to the smallest possible subset of accurate predictor genes. Feature selection has three main advantages: it decreases computational costs, mitigates the possibility of overfitting due to high inter-variable correlations, and allows for an easier clinical interpretation of the model. In this paper we compare three methods of feature selection : iterative Bayesian Model Averaging (BMA), Random Survival Forest (RSF) and Cox Proportional Hazard (CPH) and four methods of survival analysis: Analysis Random Survival Forest (RSF), Cox Proportional Hazard (CPH), Alan Additive Filter (AAF) and Deepsurv (neural network). Feature selection methods permit to extract the most relevant features for survival analysis from the dataset automatically and are compared with a hand selected set of features. Certain survival analysis methods also permit to perform feature selection such as RSF and CPH. All the data we used came from the Metabric breast cancer dataset. For every feature selection method, we compare varied numbers of selected features. Our results indicate that feature selection improves the performance of survival analysis methods. Overall, the best survival analysis performance was obtained by combining RSF for feature selection and Deepsurv for survival prediction.

I. PRINCIPLES OF SURVIVAL ANALYSIS

Survival analysis is about predicting the time duration until an event occurs [7]. In our case, we are interested in using microarray data to estimate the longevity of a patient diagnosed with breast cancer. However, the approach is generalizable to any highly dimensional data used for survival analysis tasks.

A. Modeling Survival Analysis

In survival analysis, we do not necessarily know the actual longevity of every individual since the experiment may have stopped within their lifetime. Those individuals who have not been subject to the death event during the study are labeled as *right censored*. This includes any patient who drops out of the study early. As such, for each individual, we consider either the survival time (if we have the death date) or a censored time (if we do not have the date of the death but instead the date of the last visit to the doctor). An instance in the survival data is usually represented as (x_i, t_i, δ_i) where x_i is the feature vector, t_i is the observed time, and δ_i is the indicator. An indicator of 1 is used to indicate an uncensored instance (i.e., the death of a patient) while 0 is used to indicate a censored instance. The survival function $S(t|x) = P(O > t|x)$ represents the probability of being alive after time t, where O represent the total survival time and x the data from which is inferred the prediction.

B. Concordance Index

We cannot evaluate the predictive ability of a survival model using classical loss functions such as L2 because we do not possess the curve which represents the probability of dying for our data [5]. Instead, we only know the time at which a patient actually died. One measure used is the concordance index, or c-index. This measure evaluates the ordering of predicted survival times: 0.5 is the expected result from random predictions, 1.0 is perfect concordance, and 0.0 is perfect anti-concordance (i.e., one must multiply predictions by -1 to get a c-index of 1.0). Throughout this paper, we use the c-index to evaluate the accuracy of our predictions.

II. METHODS OF SURVIVAL ANALYSIS

A. Cox Model

Cox hazard regression is a standard method for survival analysis. It consists in modeling the hazard function defined by:

$$\lambda(t|x) = \lim_{\Delta t \to 0} \frac{Pr(t <= O <= t + \Delta t | O >= t|x)}{\Delta t} \quad (1)$$

which represents how the risk of an event per unit time changes over time, as:

$$\lambda(t|x) = \lambda_0(t) exp(\beta^T x) \quad (2)$$

where $\beta = (\beta_1, .., \beta_2)^T$ is a vector of parameters, $\lambda_0(t)$ is a baseline hazard function, O represent the total survival time, and x the data from which is inferred the prediction. We also define $h(x) = \beta^T x$ as the risk function: it represents how the risk of an event per time unit changes over time.

B. Alan Additive Filter

Alan Additive filter works the same way as Cox Proportional Hazard but instead it uses :

$$\lambda(t) = \beta_0(t) + \sum_{i=1}^{T} \beta_i(t).x_i \tag{3}$$

as a regression function.

C. Deepsurv

Deepsurv is a deep learning method for survival analysis based on the Faraggi and Simon network [2]. The implementation of the model is based on Theano [4], which defines the likelihood of the model as:

$$L(\theta) = \prod_{k=1}^{K} \frac{exp(h_\beta(x_k))}{\sum_{m=1}^{M} exp(h_\beta(x_m)} \tag{4}$$

where θ represents the parameterized weights of the network on which the learning is made, h_β is the risk function of the Cox model, m is the number of patients still at risk at time t_i, and K is the total number of patients in the dataset. The hazard function is then $\lambda(t|x) = \lambda_0(t)exp(h(x))$.

As many numbers included in $]0, 1[$ are multiplied it is relevant to sum the logarithm of those numbers instead. This is what is done in Deepsurv. The loss function (l) that was chosen for this model is then defined as the negative log of this likelihood. This method yields equation 7.

With this network and 16 hand-selected features, the team that developed Deepsurv achieved a concordance index of 0.69 [2].

D. Random Survival Forest (RSF)

Random Forest is a method that operates by constructing a multitude of decision trees at training time and outputting the class (in case of a classification analysis) or the mean regression (in case of a regression analysis). This popular machine learning method was adapted to survival analysis. We used the R package *randomForest*. Random forest also permits to rank features and so it provides another features selection method.

III. FEATURE SELECTION

The aim of this paper is to prove the usefulness of feature selection before survival analysis. High-dimensional data pose a great challenge for computational techniques. We aim to extract biological information from this large amount of data to filter the expression dataset down to the smallest possible subset of accurate predictor genes. Reducing the number of predictor genes has three main advantages: it decreases computational costs, mitigates the possibility of overfitting due to high inter-variable correlations, and allows for an easier clinical interpretation of the model.

A. Cox Proportional Hazard for feature selection

COX Proportional Hazard permits to perform feature selection. The method consists in ranking the features in descending order of their log likelihood.

B. Bayesian Model Averaging

For high-dimensional data, many features can represent the data equally well. Here, we use the term *model* to refer to any such subset of selected features. We focus our attention on one feature selection method in particular, Bayesian Model Averaging (BMA), used here to select subsets of genes from microarray data [1]. Instead of choosing a single *model* and proceeding as if the data was actually generated from it, BMA combines the effectiveness of multiple models by taking the weighted average of their posterior distributions [1]. The core idea behind Bayesian Model Averaging is expressed in the following equation:

$$P(\Delta|M_k, D) = \sum_{k=1}^{K} P(\Delta|M_k, D)P(M_k|D) \tag{5}$$

where Δ is the quantity of interest we want to compute, it can be be a blood pressure or a temperature, D is the data, and M_k is the k^{th} model. The probability of the quantities to take a certain value is assumed to be the mean given by all models weighted by their own probability. There are three apparent issues at this point: obtaining the subsets of relevant models $\{M_k\}$, determining $P(\Delta|M_k, D)$, and determining $P(M_k|D)$. The BMA algorithm addresses these: once we have $P(M_k|D)$, we can deduce the predictive utility of gene x_i by:

$$P(x_i|D) = \sum_{M_k/x_i \in M_k} P(M_k|D) \tag{6}$$

Note that the number of models that exist for microarray data is usually very large. Given G candidate explanatory genes in an expression set, then there are 2^G possible models to consider. Meanwhile, the number of genes in microarray datasets typically varies from 10^2s to 10^4s.

C. Bayesian Model Averaging for High-Dimensional Data

The BMA algorithm described above can only deal with data that have at most 30 features. However, the usual practice of employing stepwise backward elimination to reduce the number of genes down to 30 is not applicable in a situation where the number of predictive variables is greater than the number of samples. To address this issue, Yeung et al. developed an iterative BMA algorithm that takes a rank-ordered list of genes and successively applies the traditional BMA algorithm until all genes have been processed [3]. We apply iterative BMA to our data, using a Cox proportional hazards regression to create the rank-ordered list of genes. Genes are ranked in descending order of their log likelihood (l) based on 2, defined as:

$$l(\theta) = \sum_{k=1}^{K} h_\theta(x_k) - log \sum_{m=1}^{M} exp(h_\theta(x_m)) \qquad (7)$$

where θ represents the parameterized weights of the network on which the learning is made, h_θ is the risk function of the Cox model, M is the number of patients still at risk at time t_i, and K is the total number of patients in the dataset. The baseline hazard function, $\lambda_0(t)$, does not appear in the equation as it is assumed to be the same for all patients.

D. Summary of BMA Algorithm

The iterative BMA algorithm iterates through the user-specified list of p top-ranked genes, applying the traditional BMA algorithm for survival analysis to each group of variables in a given BMA window of size $W = 25$. Then, genes with high posterior probabilities are retained while genes with low posterior probabilities are eliminated. This entails:

```
1. Define parameters:
    (a) $D$ training set
    (b) $G$ total number of genes in the dataset
    (c) $n$ number of samples in the training set
    (d) $p$ number of top-ranked genes to process
    (e) $W$ the size of the BMA window (i.e., the number of genes processed using traditional BMA)
2. Import training set $D$
3. Rank-order all $G$ genes by applying Cox proportional hazard regression to each individual gene
4. Apply BMA to $X = (x_1, x_2, ..., x_W)$
5. $toBeProcessed \leftarrow (x_{W+1}, ..., x_p)$
6. Repeat until all $p$ genes are processed:
    (a) Remove from $X$ any gene $i$ with $Pr(x_i|D) < 1\%$
    (b) If all genes have $Pr(x_i|D) < 1\%$, then determine the minimum $Pr(x_i|D)$, $minProbne_0$, in the current BMA window. Next, remove from $X$ any gene $i$ with $Pr(x_i|D) < (minProbne_0 + 1\%)$
    (c) Replace genes removed from $X$ with top-ranked genes from $toBeProcessed$, removing them from the $toBeProcessed$ set
    (d) Apply BMA to $X$
7. Output selected genes in $X$ and their corresponding posterior probability.
```

IV. RESULTS

A. Dataset

We applied our methods to the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) dataset which originally used expression profiles to identify new breast cancer subgroups in an effort to help physicians provide better treatment recommendations. This dataset consists of gene expression data and clinical features for 1,981 patients, 43.85% of which have had an observed death due to breast cancer (with a median survival time of 1,907 days).

We applied our three algorithms of feature selection on this dataset. We also reproduced the feature set of size 14 presented by the Deepsurv team which contains : four prognostic meta-genes (CIN, MES, LYM, and FGD3-SUSD3), the age at diagnosis, the number of positive lymph nodes, the tumor size, the ER status, the HER2 status, four indicators known to be predictive of breast cancer: ERBB2, MKI67, PGR and ESR1, and the prescribed treatment (i.e., chemotherapy, radiotherapy, or hormonotherapy) [2]. Metagenes are sets of genes coexpressed in multiple cancer types. Those features are calculated from genes expression. The four prognostic meta-genes were previously found to predict accurately the survival time of patients by the winners of the Sage Bionetworks-DREAM Breast Cancer Prognosis Challenge [6].

B. Selected Features

We were interested in knowing which selected features are in common with the hand selected features. Here is a table which summarizes the common features:

BMA	RSF	COX
CHEMOTHERAPY	ER STATUS	3 CIN
RADIO THERAPY	HER2 STATUS	1 FGD3-SUSD3
HORM THERAPY		

It is interesting to note that each method seems specialized in a particular type of feature. BMA has selected only clinical features in common with the hand selected features. More exactly it has selected only clinical features related to treatment. RSF includes the two selected hormone-related features ER (estrogen) and HER2 (human epidermal growth factor receptor), which are known to be related to breast cancer. COX selected four genes related to the metagenes CIN and FGD3-SUSD3 [6].

C. Influence of the Number of Features

We aim to identify the influence of the number of selected features. To do so we selected variable numbers of features for each feature section method. The sizes used are : 12, 14, 16, 18, 20, 22, 32, 64 and 128. We then tested each survival method on every subset generated. The Deepsurv method needs a lot of fine tuning and is very long to train. For this reason, we don't have as many results with Deepsurv as with the three other methods. Also, Deepsurv is extremely sensitive to the dataset used. It needs to be fine tuned differently when we change feature selection method. This makes the method not appropriate for comparing feature selection methods. We will present the results we obtained with RSF, COX and AAF here and we will discuss Deepsurv in a later section.

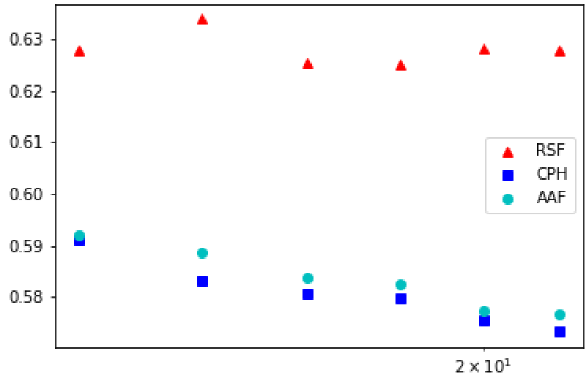

FIG. 1. Alan additive filter, Random survival Forest and Cox Proportional Hazard on sub-sets of features of various sizes generated by Bayesian Model Averaging

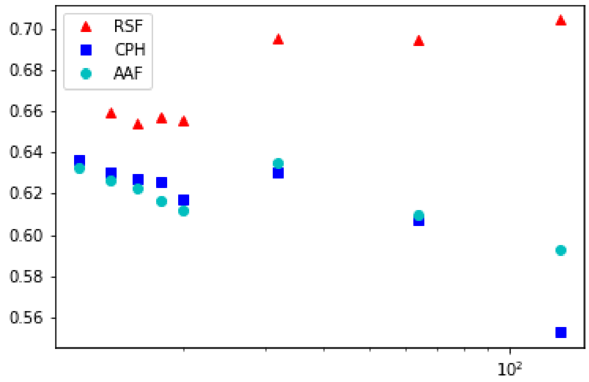

FIG. 2. Alan additive filter, Random survival Forest and Cox Proportional Hazard on sub-sets of features of various sizes generated by Cox Proportional Hazard

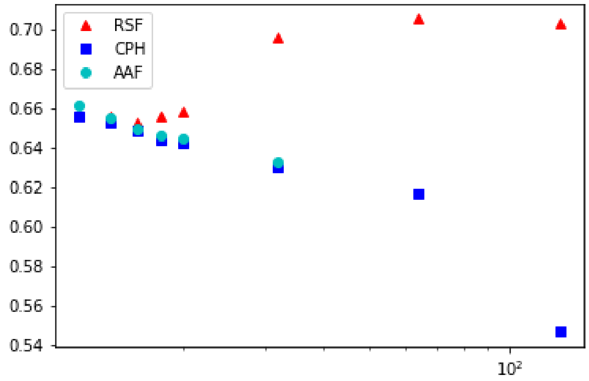

FIG. 3. Alan additive filter, Random survival Forest and Cox Proportional Hazard on sub-sets of features of various sizes generated by Random Survival Forest

It is interesting to notice in Fig. 1, Fig. 2, and Fig. 3 that the number of features seems to have a negative impact on the predictions of BMA and AAF. This is counter-intuitive as a dataset with many more features carries more information and then is expected to provide more accurate prediction. We also note that RSF clearly outperforms the two other methods presented here on every dataset and gets more accurate when the number of features increases.

D. Influence of Prevalent Features

The hand selected feature set composed of 16 features is used by the Deepsurv team to reach its best concordance index of 0.695. This feature set provides a concordance index of 0.7024 with RSF, 0.6635 with CPH and 0.6630 with AAF. None of our automatic feature selection methods could provide results as high as these on this same number of features. We have tried to evaluate the influence of particular features present in the hand selected set. To do this, we elaborated two methods. First we replaced subsets of features in the hand selected set of features by features selected by automatic feature selection methods. For this first method we expect to see a negative effect on the concordance index compared to the results given by the hand selected features. We reproduced the hand selected feature set without the indicators ERBB2, MKI67, PGR and ESR1, one without the treatments, hormonal information, chemotherapy, radiotherapy, hormonotherapy, ER, and HER2 and one without the metagenes, CIN, MES, LYM, and FGD3-SUSD3.

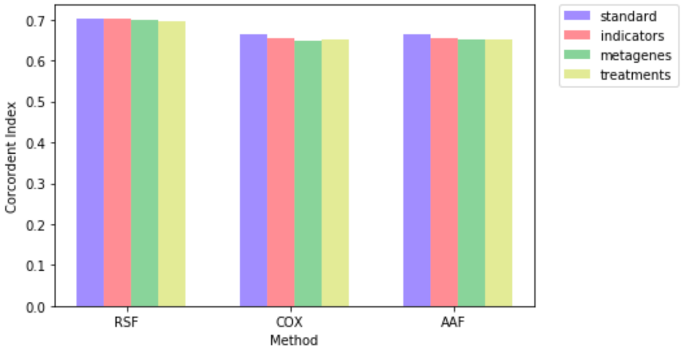

FIG. 4. Alan additive filter, Random survival Forest and Cox Proportional Hazard on subsets of features generated from the hand selected feature set from which was removed certain subsets of features and replaced by others obtained by automatic feature selection methods

One can see in Fig. 4 that the effect of removing features is very slight but we can still notice a negative effect. Those experiments do not permit to understand the difference of accuracy with automatic feature selection methods.

The second method consisted in adding those same features to automatically selected feature sets. We chose to use the RSF method as it provided the best results. In this case we expect to see a positive effect on the concordance index as can be verified on Fig. 5, although the impact is slight.

E. Prediction with Deepsurv

A deep neural network needs fine-tuning in order to find the appropriate parameters which provide the best predictions. If the size of the input changes, the architecture of the network changes, and the fine-tuning must also change. The proposed analysis focused here on four subsets of the Metabric dataset: one with the 12 most likely features, one with the 14 most likely features, one with the 16 most likely features and one with the features selected by hand by the DeepSurv team.

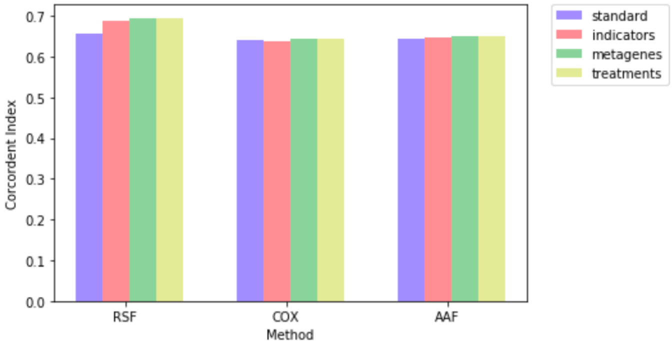

FIG. 5. Alan additive filter, Random survival Forest and Cox Proportional Hazard on subsets of features generated from RSF feature selection set from which was removed certain subsets of features and replaced by others taken from hand selected features

12 BMA	14 hand-selected	14 RSF	64 RSF
0.5388	0.6676	0.6720	0.7173

These results show that RSF automatically selected 14 features perform at least as well as the hand-selected features with Deepsurv, which is an important result. Another important result is that a moderate additional number of features improves Deepsurv's performance even more, however only with the RSF feature selection method.

V. CONCLUSION

Although more work is needed to reach a definitive conclusion, our results indicate that feature selection can play a helpful role when performing survival analysis on high-dimensional data. In addition to the uncontested role in decreasing computational costs, the survival prediction shows improved concordance index. RSF though takes advantage of the information contained in a moderate additional number of features. Overall, the best survival analysis performance was obtained by combining RSF for feature selection and Deepsurv for survival prediction.

[1] Amalia Annest. *Iterative Bayesian Model Averaging: A Method for the Application of Survival Analysis to High-Dimensional Microarray Data*. University of Washington,

2008.

[2] Jonathan Bates Alexander Cloninger Tingting Jiang Jared L. Katzman, Uri Shaham and Yuval Kluger. *Deep Survival: A Deep Cox Proportional Hazards Network*. BioMed Central, 2016.

[3] Adrian E. Raftery Ka Yee Yeung, Roger E. Bumgarner. *Bayesian model averaging: development of an improved multi-class, gene selection and classification tool for microarray data*. Oxford Academic, 2005.

[4] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.

[5] Balaji Krishnapuram Vikas C. Raykar, Harald Steck. ”on ranking in survival analysis: Bounds on the concordance index”.

[6] Tai-Hsien Ou Yang Wei-Yi Cheng and Dimitris Anastassiou. *Biomolecular events in cancer revealed by attractor metagenes*. PLoS Comput Biol, 2013.

[7] Jiawen Yao Xinliang Zhu and Junzhou Huang. *Deep Multi-Instance Learning for Survival Prediction from Whole Slide Pathological Images*. BioMed Central, 2016.