

RESEARCH

Open Access



# Random rotation survival forest for high dimensional censored data

Lifeng Zhou, Hong Wang\* and Qingsong Xu

\*Correspondence:  
wh@csu.edu.cn  
School of Mathematics  
and Statistics, Central South  
University, South Shaoshan  
Road, Changsha 410075,  
Hunan, China

## Abstract

Recently, rotation forest has been extended to regression and survival analysis problems. However, due to intensive computation incurred by principal component analysis, rotation forest often fails when high-dimensional or big data are confronted. In this study, we extend rotation forest to high dimensional censored time-to-event data analysis by combining random subspace, bagging and rotation forest. Supported by proper statistical analysis, we show that the proposed method random rotation survival forest outperforms state-of-the-art survival ensembles such as random survival forest and popular regularized Cox models.

**Keywords:** Survival ensemble, Rotation forest, Time-to-event data, Censored data, High-dimensional data

## Background

Survival analysis of censored data plays a vital role in statistics with abundant applications in various fields such as biostatistics, engineering, finance and economics. As an example, regression analysis of time-to-event data finds wide applications in reliability studies in industrial engineering and inter-child birth times research in demography and sociology. To estimate the probability that a subject (patient or equipment) will survive past a certain time, various parametric, semi-parametric and non-parametric models such as Cox proportional hazard (Cox PH) model (Cox and Oakes 1984; David 1972), survival neural network (Faraggi and Simon 1995), survival tree (Bou-Hamad et al. 2011; LeBlanc and Crowley 1995), regularized Cox PH model (Fan and Li 2002), regularized accelerated failure time (AFT) model (Huang et al. 2006), supervised principal components based survival models (Li and Li 2004) have been proposed.

The past two decades have seen various survival ensembles with parametric and/or non-parametric base models and combining techniques. These techniques include bagging (Hothorn et al. 2004, 2006), boosting (Binder and Schumacher 2008; Binder et al. 2009; Hothorn and Bühlmann 2006; Li and Luan 2005; Ma and Huang 2007; Ridgeway 1999; Wang and Wang 2010), random survival forest (RSF) (Ishwaran et al. 2010, 2011) and the recently proposed rotation survival forest (RotSF) (Zhou et al. 2015). Bagging stochastically changes the distribution of the training data by constructing a base survival model based on different bootstrap samples (Hothorn et al. 2004). Boosting based approaches adaptively change the distribution of the training data according to the

performance of previously trained base models and usually either use all covariates to fit the gradients in each step for low-dimensional data (Ridgeway 1999) or update only one estimate of parameters corresponding to only one covariate in a componentwise manner in case of high-dimensional data (Binder et al. 2009). Random survival forest (RSF) (Ishwaran et al. 2008) extends random forest (RF) (Breiman 2001) to right-censored time-to-event data using the same principles underlying RF and enjoys all RF's important properties. In RSF, tree node splits are designed to maximize survival differences between subtree nodes. A so-called ensemble cumulative hazard function (CHF) can be estimated by aggregating Nelson–Aalen estimators for all “in-bag” data samples. All these survival ensembles have demonstrated their usefulness compared to previous single algorithms.

Rotation survival forest (RotSF) (Zhou et al. 2015) is newly proposed survival ensemble based on rotation forest (RotF) (Rodriguez et al. 2006), in which the training data for each base model is formed by applying PCA transformation to rotate the original covariates axes. In RotSF and other RotF based approaches, ensemble diversity is achieved by covariates transformation for each base model and prediction accuracy is promoted by keeping all principal components in the training data set. However, due to intensive computations during eigenvalue decomposition of data covariance matrix, such approaches often fail when dealing with high-dimensional data.

In view of the fact that dimensionality reduction can be achieved by random subspace (Ho 1998) method which randomly selects a small number of dimensions from a given covariate set in building a base model, we propose a new survival ensemble called random rotation survival forest (RRotSF) for analyzing high-dimensional survival data. The proposed methodology can be viewed as a combination of rotation forest, random subspace and bagging (Breiman 1996). And it extends the RotSF approach from low dimensional to high dimensional time-to-event censored data. In this study, the decision tree algorithm is chosen as the base survival model for our survival ensemble as it is the most popular non-parametric method in analyzing survival data (Bou-Hamad et al. 2011).

## Methods

Given a training dataset:  $D = (\tau_q, \delta_q, \mathbf{X}_q), q = 1, \dots, n$ , where  $\tau_q$  is the survival time for the  $q$ -th sample,  $\delta_q$  is the censored status indicator,  $\mathbf{X}_q$  is a variable set  $\mathbf{V}$  of  $p$  covariates and  $n$  is the number of observed samples, a high-level description of how the proposed RRotSF algorithm train a base survival model  $S_i$  in the ensemble is presented in the following:

1. Randomly select  $r < p$  covariates from the  $p$ -dimensional data set  $D$  and the newly obtained training set  $D_r = (\tau_q, \delta_q, \mathbf{X}_{qi})$  consists of  $r$ -dimensional training samples. Here we set  $r = \lceil \sqrt{p} \rceil$  for simplicity.
2. Generate a bootstrap sample  $D' = (\tau'_q, \delta'_q, \mathbf{X}'_{qi})$  of size  $n$  from  $D_r$  to enhance diversity and for calculating covariate importance.
3. Randomly split variables  $\mathbf{V}$  into  $k = r/M$  equal size subsets  $V_j, j = 1, \dots, k$  and denote the not used covariates (remaining variables) as  $RV$ . Apply PCA to each bagged training subset with  $V_j$  covariates. Retain all derived principal component rotations  $M_j$ s and set rotations of  $RV$  to 0 to inject more randomness.

4. Arrange all PCA rotations to match variable order in  $V$  and obtain rotation matrix  $R_i^a$ .
5. Use the newly transformed data  $D_t = (\tau'_q, \delta'_q, \mathbf{X}'_{qi}R_i^a)$  as the training set to train a base survival model  $S_i$ .

The major difference between RotF (also RotSF) and the proposed RRotSF lies in that the former transforms the whole training set via PCA while the latter transforms only a random subspace of the whole training set which in turn greatly reduces the computational complexity caused by eigenvalue decomposition of high-dimensional covariance matrix.

The pseudo-code of the proposed RRotSF algorithm is presented in Algorithm 1:

---

**Algorithm 1** The Random Rotation Survival Forest Algorithm

---

```

1: INPUT:
2:   Training data:  $D = (\tau_q, \delta_q, \mathbf{X}_q), q = 1, \dots, n$ , where  $\mathbf{X}$  is a variable set  $\mathbf{V}$  of  $p$  covariates
3:    $M$ : Number of variables within a subset
4:    $L$ : Ensemble size
5:    $S_i$ : Survival tree base learning algorithm
6: OUTPUT:
7:   Survival ensemble  $S$ 
8: procedure RROTFSF( $D, M, L$ )
9:   while  $i$  in  $1 : L$  do
10:    Prepare the random rotation matrix  $R_i^a$ :
11:    Randomly select  $r$  covariates from variable set  $V$  where  $r = \lceil \sqrt{p} \rceil$  and the newly obtained
    training set  $D_r = (\tau_q, \delta_q, \mathbf{X}_{qi})$ 
12:    Generate a bootstrap sample  $D' = (\tau'_q, \delta'_q, \mathbf{X}'_{qi})$  of size  $n$  from  $D_r$ 
13:    Randomly split variables  $\mathbf{V}$  into  $k = r/M$  equal size subsets  $V_{i,j}, j = 1, \dots, k$ 
14:    The not chosen variables are denoted by  $RV$ 
15:    while  $j$  in  $1 : k$  do
16:      Generate a bootstrap sample  $D''$  of size  $n$  from  $D'$ 
17:      Let  $X'_{i,j}$  be subset of  $D''$  with variable set  $V_{i,j}$ 
18:      Apply PCA on  $X'_{i,j}$  to get the rotation matrix  $M_{i,j}$ 
19:    end while
20:    Set the corresponding rotations of  $RV$  variables to 0
21:    Rearrange all  $M_{i,j}$ s according to the variable order in  $V$  and obtain rotation matrix  $R_i^a$ 
22:    Train a survival tree base learner  $S_i$  using  $D_t = (\tau'_q, \delta'_q, \mathbf{X}'_{qi}R_i^a)$  as the training set
23:  end while
24:  return The survival ensemble  $S$ 
25: end procedure
26: In prediction, the average hazard  $h$  for new data  $\mathcal{X}$ :

```

$$h = \frac{1}{L} \sum_{S_i \in S} S_i(\mathcal{X}R_i^a)$$


---

Some parameters should be specified before applying RRotSF. Similar to other ensemble methods, ensemble size which specifies the number of base survival models can be tuned by the users. Parameters  $M$  which controls the number within a feature subset is set to 2 as is done in RotSF.

**Results and discussion**

In the experiments, we perform five replications of two fold cross-validation as suggested by Dietterich (1998). In fivefold to twofold cross-validation, the dataset is randomly divided into two halves, the first half is used for training and the other half for testing and vice versa. This process is repeated five times for each dataset.

## Datasets

In order to carry out empirical comparisons, we want to test the proposed algorithm on five real high-dimensional benchmark datasets. In the following datasets, when distant metastasis-free survival (DMFS) time values are available, DMFS values are used as the primary survival end-points, otherwise relapse-free or overall survival time values are applied.

A short introduction of the benchmark datasets are given below.

### *UPP dataset*

The UPP dataset contains transcript profiles of 251 p53-sequenced primary breast tumors published by Miller et al. (2005). In each patient sample, 44,928 gene features and 21 clinical covariates are provided. The data can be obtained from the R package “breastCancerUPP” of “Bioconductor”.

### *MAINZ dataset*

The MAINZ breast cancer dataset provided by Schmidt et al. (2008) contains the gene expression patterns of 200 tumors of patients who were not treated by systemic therapy after surgery using a discovery approach. Each patient sample contains 22,283 gene features and 21 clinical covariates. The dataset is available from the R package “breastCancerMAINZ” of “Bioconductor”.

### *TransBig dataset*

This breast cancer dataset contains gene expression and clinical data published in Desmedt et al. (2007). The data contains 198 samples to independently validate a 76-gene prognostic breast cancer signature as part of the TransBig project. In the data, 22,283 gene features and 21 clinical covariates are provided for each sample. The dataset can be obtained through the R package “breastCancerTRANSBIG” of “Bioconductor”.

### *VDX dataset*

The Veridex (VDX) dataset which contains 344 patients with primary breast cancer was published in Wang et al. (2005). In the data, 22,283 gene features and 21 clinical covariates are provided for each sample. The dataset can be obtained through the R package “breastCancerVDX” of “Bioconductor”.

### *TCGA dataset*

This dataset is provided by The Cancer Genome Atlas (TCGA) and presented in Fang and Gough (2014). It contains both clinical covariates and gene expression information of 3096 cancer patients covering 12 major types of cancers. In each sample, 19,420 gene state information and 5 clinical covariates are provided. The data is available from the R package “dnet” of “CRAN”.

Summary information including gene features, clinical covariates and the number of samples of all datasets can be found in the following Table 1.

**Table 1 Summary of five benchmark datasets used**

	Gene features	Clinical covariates	Samples
UPP	44,928	21	251
MAINZ	22,283	21	200
TransBig	22,283	21	198
VDX	22,283	21	344
TCGA	19,420	5	3096

### Performance metrics and statistical tests

In survival analysis, we are much concerned with the relative risks between patients with different covariates information. Hence, as suggested by Ishwaran et al. (2008), we adopt Harrell's concordance index (C-index, CI) (Harrell et al. 1996) to evaluate the accuracy of such relative risks in our later experiments and rank their performance on all datasets.

CI can be calculated in the following steps:

- Create all pairs of observed survival times.
- For all valid survival time pairs, namely, pairs where one survival time  $T_{j1}$  is greater than the other  $T_{j2}$ , test whether the corresponding predictions are concordant, i.e.,  $\eta_{j1} > \eta_{j2}$ . If so, add 1 to the running sum  $s$ ; If  $\eta_{j1} = \eta_{j2}$ , add 0.5 to the sum  $s$ ; If  $\eta_{j1} < \eta_{j2}$ , add 0 to the sum  $s$ .
- Count the number  $n$  of valid survival time pairs. Divide the total sum  $s$  by the number of valid survival time pairs  $n$  and we obtain  $CI = s/n$ .

Similar to AUC used in classification, CI usually lies between 0.5 and 1. When  $CI = 1$ , it means that the model has a perfect prediction accuracy and when  $CI = 0.5$ , it implies that the model is just like random guessing.

The results obtained in experiments are further validated by some proper statistical tests. As suggested by Demšar (2006) and was done in Zhou et al. (2015), we use the non-parametric Friedman rank sum test (Demšar 2006) to test the statistical significance of various survival models. If the value Friedman test is large enough, the null hypothesis that there is no significant difference among the different survival models can be rejected and some post-hoc such as Nemenyi test can be applied to find where the differences lie. If the differences are not significant according to the Nemenyi statistics, we use a two-sample Wilcoxon test to check whether the difference between pairs is significant.

### Comparison results

Here, we compare RRotSF with five popular survival models. The first two methods are random survival forest (RSF) with different splitting rules, namely RSF-Logrank (RSF1) and RSF-Logrankscore (RSFs); the third and fourth methods are regularized Cox proportional hazard models, i.e. Cox-Lasso and Cox-Ridge; the fifth method is fast cocktail Cox method (CockTail). For the ease of notation, RRotSF, RSF1, RSFs, Cox-Lasso, Cox-Ridge and CockTail are denoted by A, B, C, D, E and F respectively when necessary. Comparisons with five models are conducted with corresponding "glmnet" (Simon et al. 2011), "randomForestSRC" (Ishwaran et al. 2008), and "fastcox" (Yang and Zou 2012) packages

in R. Default settings are adopted for all models. For ensemble methods, i.e. RRotSF, RSFI and RSFs, 500 trees are built.

The corresponding experiment results are listed in the following Table 2. In this table, the numerics in each entry are the average of CI values on fivefold to twofold cross-validation. The best performance in each column (on each dataset) is highlighted by the italic font.

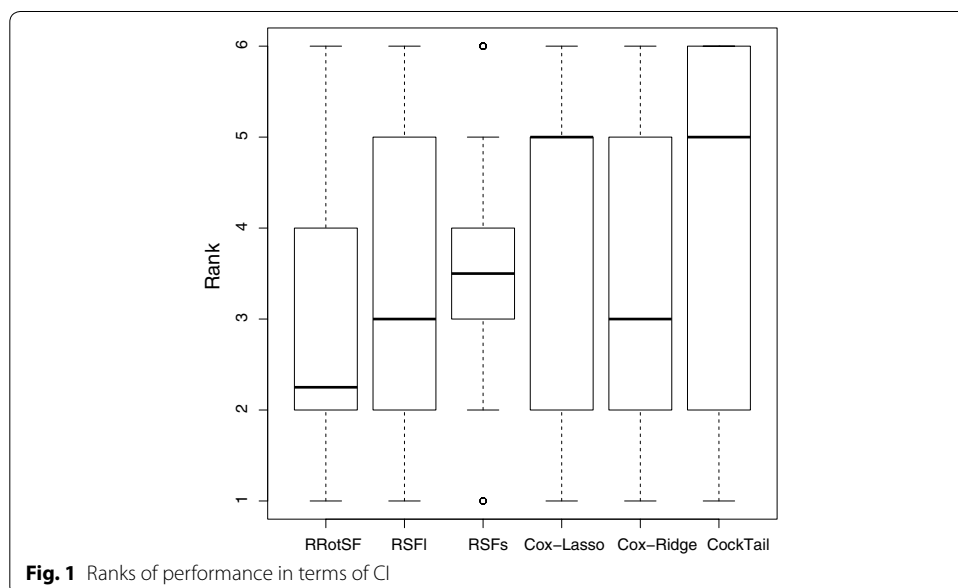
According to Table 2, the proposed RRotSF takes the first place once, takes the second place three times and also takes a fourth place. Though RSFI takes the first place twice, its performance on other tree datasets are rather poor: it takes one fourth, one fifth and one last place respectively. From Table 2, whether RRotSF beats RSFI is not clear at this time but we can safely say that RRotSF outperforms other models, namely RSFIs, Cox-Lasso, Cox-Ridge and CockTail in most cases in terms of averaged CI.

To further evaluate the performance of all compared models, we have ranked each model on every run on these benchmark datasets. This allows us to compare performance of all models in a consistent and nonparametric way. Figure 1 presents the box-plot of ranks of six models in all runs of the experiments.

From the above, we can observe that RRotSF excels, followed by Cox-Ridge and RSFI models. The worst performer on these datasets is CockTail. In spite of the ranks, we also want to contrast these statements with some statistical tests.

**Table 2 Performance in terms of averaged CI**

	UPP	MAINZ	TransBig	VDX	TCGA
RRotSF	0.6210	0.6997	0.5540	<i>0.6248</i>	0.6287
RSFI	<i>0.6461</i>	<i>0.7069</i>	0.5177	0.5630	0.5740
RSFIs	0.5813	0.6234	0.5375	0.5950	0.6569
Cox-Lasso	0.5763	0.6375	0.5482	0.5327	0.7032
Cox-Ridge	0.6149	0.6802	<i>0.5702</i>	0.6234	0.5516
CockTail	0.5906	0.6298	0.5383	0.5227	<i>0.7051</i>



The Friedman rank sum test outputs a p-value of 0.0001136 which reject the null hypothesis that there is no significant difference among these models and a post-hoc Nemenyi test is applied. Using RRotSF (A) as the control, we obtain the p-values of Nemenyi test for different pairs:  $p_{BA} = 0.35512, p_{CA} = 0.04505, p_{DA} = 0.00184, p_{EA} = 0.61393$  and  $p_{FA} = 0.00022$ . It can be seen that there exist significant differences between RRotSF and RSFs, Cox-Lasso or CockTail.

The differences between RRotSF and RSF1 or Cox-Ridge are not significantly different according to the Nemenyi test. However, a pairwise comparison using the Wilcoxon test rejects the hypothesis of equivalence with low p-values ( $p_{BA} = 0.02189$  and  $p_{EA} = 0.02129$ ). This also indicates RRotSF is also superior to RSF1 and Cox-Ridge on these benchmark datasets.

Therefore, in terms of C-index metric, RRotSF outperforms state-of-the-art survival models such as Random Survival Forest, regularized Cox proportional hazard models on these benchmark datasets. It is clear that other methods (ensembles and not) are available but the goal here is to illustrate some key features of RRotSF and not to provide an exhaustive comparison across methods.

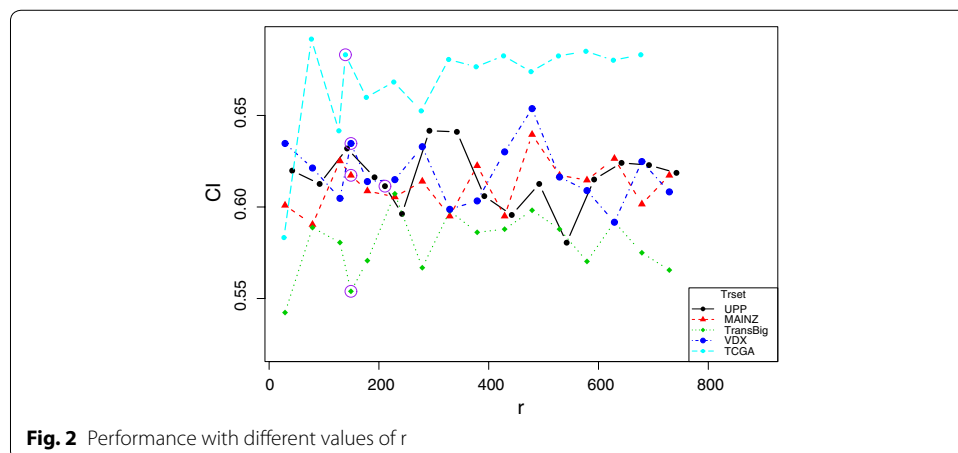
**Parameter sensitivity analysis**

In addition to the above experiments, we also want to examine the sensitivity of RRotSF to the choice of parameters in the underlying survival models.

First, we want to test the performance of RRotSF with different subspace values (the number of variable with each variable subset)  $r$ . In view of the fact that  $r < \sqrt{p}/5$  may result in a less accurate base survival tree and  $r > 5\sqrt{p}$  may cause RRotSF cease to work due to memory overflow problems as all the datasets here are high-dimensional ones, we only test RRotSF with  $r$  values ranging from  $\sqrt{p}/5$  and  $5\sqrt{p}$  in the experiments.

Figure 2 shows the performance of RRotSF with different values of  $r$  on all five benchmark datasets. Performance of the default value( $r = \lceil \sqrt{p} \rceil$ ) of  $r$  on each datasets is indicated by a purple circle.

From Fig. 2, one may observe that except for the values at the very beginning on TransBig and TCGA datasets, RRotSF seems insensitive to changes of  $r$  values. This is very encouraging result, as it demonstrates that RRotSF is robust with respect to  $r$ , even if non-default values are chosen.



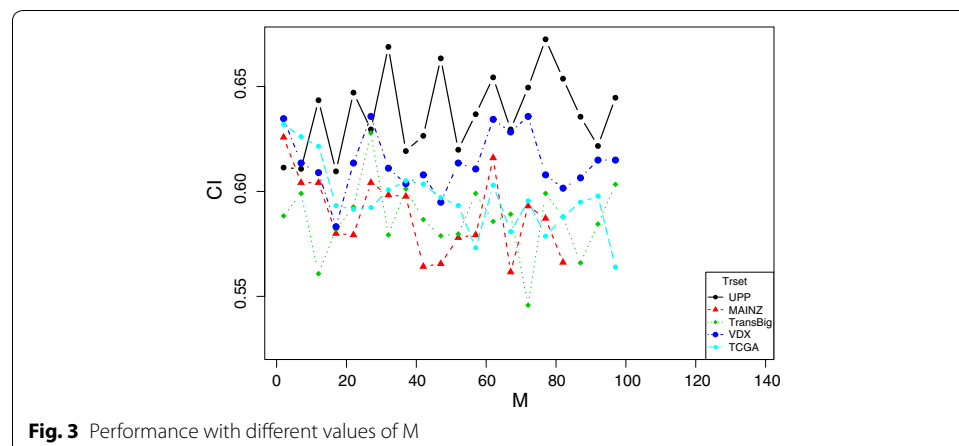
Next, we want to test the performance of RRotSF with number of variable with a subset  $M$ . If  $M = 1$ , then any projection reduces to rescaling of the variable axes. If  $M = p$ , there is only one variable set, i.e. all the variables are used for PCA transformation. In both cases, the ensemble diversity are degraded and prediction errors are larger than those with values in between (Kuncheva and Rodríguez 2007). To see how the choice of  $M$  may influence RRotSF's performance, we test  $M$  with values between 2 and 100. Figure 3 shows the performance of RRotSF with different values of  $M$  on all five benchmark datasets.

The results shown in Fig. 3 agree with the results obtained for RotF in the classification context (Kuncheva and Rodríguez 2007), i.e., there is no consistent pattern or regularity for  $M$  with small values.

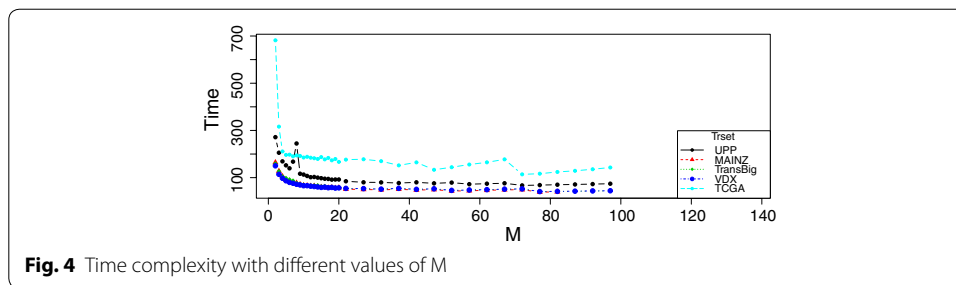
We also consider the time efficiency of RRotSF for different values of  $M$ . From Algorithm 1, one may observe that the major time complexity lies with PCA operations in transforming  $k$  group of variable subsets. If  $M$  is small, each PCA operations will be faster but as the number of variable subsets could be greater, we have to do more PCA transformations. If  $M$  is large, each PCA operation will take a longer time but the the number of variable subsets and hence the number of PCA operations could be less. Figure 4 shows the running times (in seconds) of RRotSF on benchmark datasets with different combinations of  $M$  and  $k$ .

From Fig. 4, we notice a sharp decrease in RRotSF's running time when  $M$  increases if  $2 \leq M \leq 5$ , and a very slow decrease when  $M$  increases if  $5 < M \leq 20$ . When  $M > 20$ , the values of  $M$  have no direct influence on RRotSF's time efficiency as RRotSF's running time remain almost steady on all five benchmark datasets. If we focus only on RRotSF's time efficiency, we should choose a larger  $M$  ( $M < 20$ ). However, to make RRotSF also work for some low-dimensional datasets,  $M$  should be set to a small value to ensure that there is enough diversity among the survival ensemble. Hence, to make a tradeoff between time efficiency and prediction accuracy, the  $M$  value can be set to 2 or 3 for simplicity, though it is not a optimal value in most cases.

From the above, both the choices  $r = \sqrt{p}$  and  $M = 2$  in the default setting for RRotSF are not the best choice in terms of  $C$ -index and are just serendipitous guesses in this study. As we have shown in the above comparison results, RRotSF has outperformed other popular survival models for these rather unfavourable values, we may conclude







that RRotSF is not sensitive to the choice of  $r$  and  $M$ . As both values work well in the experiments, we propose to use these values as default values in the future. Of course, one can use cross-validation techniques to tune these parameters on some particular cases for a better performance.

## Conclusion

In this study, we have developed a new ensemble learning algorithm, random rotation survival forest, for high-dimensional survival analysis. By studying the famous benchmark datasets, we have found that the proposed method generally outperforms state-of-the-art survival models such as random survival forest, regularized Cox proportional hazard models in terms of C-Index metric. As a non-parametric approach, RRotSF does not impose parametric assumptions on hazard functions, and it extends the well-known rotation forest methodology to high-dimensional data analysis.

The R code and the supplementary material are available at url: <https://github.com/whcsu/RRotSF> and we are working hard to provide an R package for the proposed RRotSF algorithm as soon as possible.

## Authors' contributions

The work presented here was carried out in collaboration between all authors. LFZ, HW and QSX defined the research theme; HW and QSX designed the algorithm; HW carried out the experiments and analyzed the data; LFZ and HW interpreted the results and wrote the paper. All authors read and approved the final manuscript.

## Acknowledgements

This work was supported in part by Social Science Foundation for Young Scholars of Ministry of Education of China Under Grant No. 15YJCZH166. The authors want to thank all reviewers and the editor for their valuable and constructive comments, which greatly improves the quality of this paper.

## Competing interests

The authors declare that they have no competing interests.

Received: 11 June 2016 Accepted: 19 August 2016

Published online: 26 August 2016

## References

- Binder H, Schumacher M (2008) Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models. *BMC Bioinform* 9(1):14
- Binder H, Allignol A, Schumacher M, Beyersmann J (2009) Boosting for high-dimensional time-to-event data with competing risks. *Bioinformatics* 25(7):890–896. doi:10.1093/bioinformatics/btp088
- Bou-Hamad I, Larocque D, Ben-Ameur H (2011) A review of survival trees. *Stat Surv* 5:44–71
- Breiman L (1996) Bagging predictors. *Mach Learn* 24(2):123–140
- Breiman L (2001) Random forests. *Mach Learn* 45(1):5–32
- Cox DR, Oakes D (1984) Analysis of survival data, vol 21. CRC Press, Boca Raton
- David CR (1972) Regression models and life tables (with discussion). *J R Stat Soc* 34:187–220
- Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7:1–30

- Desmedt C, Piette F, Loi S, Wang Y, Lallemand F, Haibe-Kains B, Viale G, Delorenzi M, Zhang Y, d'Assignies MS et al (2007) Strong time dependence of the 76-gene prognostic signature for node-negative breast cancer patients in the transbig multicenter independent validation series. *Clin Cancer Res* 13(11):3207–3214
- Dietterich TG (1998) Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput* 10(7):1895–1923
- Fan J, Li R (2002) Variable selection for cox proportional hazards model and frailty model. *Ann Stat* 30(1):74–99. doi:[10.2307/2700003](https://doi.org/10.2307/2700003)
- Fang H, Gough J (2014) The 'dnet' approach promotes emerging research on cancer patient survival. *Genome Med* 6:64. doi:[10.1186/s13073-014-0064-8](https://doi.org/10.1186/s13073-014-0064-8)
- Faraggi D, Simon R (1995) A neural network model for survival data. *Stat Med* 14(1):73–82
- Harrell FE, Lee KL, Mark DB (1996) Tutorial in biostatistics multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Stat Med* 15:361–387
- Ho TK (1998) The random subspace method for constructing decision forests. *IEEE Trans Pattern Anal Mach Intell* 20(8):832–844
- Hothorn T, Bühlmann P (2006) Model-based boosting in high dimensions. *Bioinformatics* 22(22):2828–2829. doi:[10.1093/bioinformatics/btl462](https://doi.org/10.1093/bioinformatics/btl462)
- Hothorn T, Lausen B, Benner A (2004) Bagging survival trees. *Stat Med* 23(1):77–91
- Hothorn T, Bühlmann P, Dudoit S, Molinaro A, Van Der Laan MJ (2006) Survival ensembles. *Biostatistics* 7(3):355–373
- Huang J, Ma S, Xie H (2006) Regularized estimation in the accelerated failure time model with high-dimensional covariates. *Biometrics* 62(3):813–820
- Ishwaran H, Kogalur UB, Blackstone EH, Lauer MS (2008) Random survival forests. *Ann Appl Stat* 2(3):841–860
- Ishwaran H, Kogalur UB, Gorodeski EZ, Minn AJ, Lauer MS (2010) High-dimensional variable selection for survival data. *J Am Stat Assoc* 105(489):205–217
- Ishwaran H, Kogalur UB, Chen X, Minn AJ (2011) Random survival forests for high-dimensional data. *Stat Anal Data Min* 4(1):115–132. doi:[10.1002/sam.10103](https://doi.org/10.1002/sam.10103)
- Kuncheva LI, Rodríguez JJ (2007) An experimental study on rotation forest ensembles. In: Haindl M, Kittler J, Roli F (eds) *Multiple classifier systems*. Springer, New York, pp 459–468
- LeBlanc M, Crowley J (1995) A review of tree-based prognostic models. In: Thall PF (ed) *Recent advances in clinical trial design and analysis*. Springer, New York, pp 113–124
- Li L, Li H (2004) Dimension reduction methods for microarrays with application to censored survival data. *Bioinformatics* 20(18):3406–3412
- Li H, Luan Y (2005) Boosting proportional hazards models using smoothing splines, with applications to high-dimensional microarray data. *Bioinformatics* 21(10):2403–2409. doi:[10.1093/bioinformatics/bti324](https://doi.org/10.1093/bioinformatics/bti324)
- Ma S, Huang J (2007) Clustering threshold gradient descent regularization: with applications to microarray studies. *Bioinformatics* 23(4):466–472
- Miller LD, Smeds J, George J, Vega VB, Vergara L, Ploner A, Pawitan Y, Hall P, Klaar S, Liu ET et al (2005) An expression signature for p53 status in human breast cancer predicts mutation status, transcriptional effects, and patient survival. *Proc Natl Acad Sci USA* 102(38):13550–13555
- Ridgeway G (1999) The state of boosting. *Comput Sci Stat* 31:172–181
- Rodríguez JJ, Kuncheva LI, Alonso CJ (2006) Rotation forest: a new classifier ensemble method. *IEEE Trans Pattern Anal Mach Intell* 28(10):1619–1630
- Schmidt M, Böhm D, von Törne C, Steiner E, Puhl A, Pilch H, Lehr H-A, Hengstler JG, Kölbl H, Gehrman M (2008) The humoral immune system has a key prognostic impact in node-negative breast cancer. *Cancer Res* 68(13):5405–5413
- Simon N, Friedman JH, Hastie T, Tibshirani R (2011) Regularization paths for Cox's proportional hazards model via coordinate descent. *J Stat Softw* 39(5):1–13
- Wang Y, Klijn JG, Zhang Y, Sieuwerts AM, Look MP, Yang F, Talantov D, Timmermans M, van Meijer-Gelder ME, Yu J et al (2005) Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet* 365(9460):671–679
- Wang Z, Wang C (2010) Buckley-James boosting for survival analysis with high-dimensional biomarker data. *Stat Appl Genet Mol Biol* 9(1):24
- Yang Y, Zou H (2012) A cocktail algorithm for solving the elastic net penalized Cox's regression in high dimensions. *Stat Interface* 6(2):167–173
- Zhou L, Xu Q, Wang H (2015) Rotation survival forest for right censored data. *PeerJ* 3:1009