Available online at www.SciMedJournal.org

SciMedicine Journal

(ISSN: 2704-9833)

Vol. 3, No. 1, March, 2021

On Fair Performance Comparison between Random Survival Forest and Cox Regression: An Example of Colorectal Cancer Study

Sirin Cetin [1], Ayse Ulgen [2*], Isa Dede [3] Wentian Li [4]

[1] *Department of Biostatistics, Faculty of Medicine, Tokat GaziosmanPasa University, Turkey.*

[2] *Department of Biostatistics, Faculty of Medicine, Girne American University, Karmi, Cyprus.*

[3] *Medical Oncology, Faculty of Medicine, Mustafa Kemal University, Antakya, Turkey.*

[4] *The Robert S. Boas Center for Genomics and Human Genetics, The Feinstein Institutes for Medical Research, Northwell Health, Manhasset, NY, United States.*

Received 12 December 2020; Revised 20 February 2021; Accepted 24 February 2021; Published 01 March 2021

Abstract

Random Forest (RF), a mostly model-free and robust machine learning method, has been successfully applied to right-censored survival data, under the name of Random Survival Forest (RSF). However, RF/RSF has its distinct strategies in classification and prediction. First, it is an ensemble classifier and its performance is an average of multiple rounds of data fitting. Second, the training set is a bootstrap (sampling with replacement) generated set with repeated used of roughly 2/3 of all samples and testing set consists of those not used (out of bag samples). Both features are not intrinsic to Cox regression or other single classifiers. Not considering these two features could potentially lead to a partial comparison between the performance of the two methods. By using a colorectal survival dataset, we illustrate the problems of using k-fold cross-validation, using only one resampling without an ensemble average, and using the whole dataset for both fitting and testing, in Cox regression, when comparing with RSF. We provide a more accessible R code for simple calculation of discordance index (D-index) and unweighted integrated Brier score (IBS) for Cox regression, and unweighted IBS for RSF.

Keywords: Random Survival Forest; Cox Regression; Machine Learning; Brier's Score; Discordance Index; Colorectal Cancer.

1. Introduction

In cancer epidemiology studies, one of the most commonly used analyses is Cox regression, which regresses the right-censored time-to-death data on risk factors. In recently years, new methodologies are introduced to survival analysis. An all-embracing name "machine learning" covers a whole spectra of these new techniques [1], with most of them "model free" making less assumption about the data.

We are particularly interested in the Random Forest (RF) [2] (or Random Survival Forest (RSF) when it is applied to survival analysis), because RF/RSF is easy to explain, easy to code, and easy to apply to data. There are already many articles published about application of RSF to survival data [3, 4] and its comparison to the standard method in survival analysis, i.e., the Cox regression (also called proportional hazards regression).

Even though public software are available making the application of RSF easy, being unfamiliar with the new

* Corresponding author: ayshe.ulgen@global.t-bird.edu

http://dx.doi.org/10.28991/SciMedJ-2021-0301-9

method leads people to use the default parameter values, and unaware that the performance of RSF may change with the parameter value [5-7], as well as not able to compare its performance fairly with standard approaches. The latter problem is partly due to the fact that RSF is an ensemble classifier (data fitter), not the one-time-only run of Cox regression. Performance comparison between the two are often not carried out appropriately. Moreover, RSF has a particular way in choosing training dataset (sample with replacement, or bootstrap) and testing dataset (samples not chosen by the bootstrap).

If the performance of Cox regression is measured from a testing set which is not equivalent to that in RSF, then the estimated performances of RSF and Cox regression are not equivalent. Besides the extremely unfair choice of using the whole dataset as both training and validation, the seemingly fair k-fold cross-validation for Cox regression is also not appropriate, in two different ways. First, it is not equivalent to RSF in that each sample is only used once in training whereas in the bootstrap set, a sample can be used more than once. Second, the typical value for k is 10, meaning the performance is averaged over 10 times, whereas the number of runs in RSF ensemble learning is of the order of hundreds or thousands.

Many good practice known by data analysts may not been made explicit or emphasized enough in the literature. We share our experience in our comparison of Cox regression and RSF on a colorectal cancer survival data. Although there are excellent software packages professionally written, we still found it difficult to find an error comparison program that fits our need. Therefore, we share our simple but hopefully accessible code in public domain.

2. Data

All analyses in this paper are done using n=221 colorectal patients, collected from Mustafa Kemal University Medical Oncology Department, with 7 independent variables: age (mean=60.5 year), gender (91 male, 130 female), cancer type (152 colon cancer, 69 rectal cancer), leukocyte (range: 3220-29440, plus one outlier 93350), neutrophile (range: 1930-27520), lymphocyte (range: 220-3850), and platelet (range: 64000-1145000). The distribution of last four variables (blood test results) all peak around 20%-30% of the maximum value (excluding one outlier of white blood cell count). The study was approved by the Ethics Committee of Mustafa Kemal University Medical Faculty (2020/28).

The dependent variable is the right censored time to death in the unit of days. The status δ_i = 1 if the i'th person is dead at diagonosis-to-death time T_i, and δ_i = 0 if the i'th person is unavailable (censored) at time after diagnosis T_i. The mean diagonosis-to-death time for patients who die is 617 days (1.69 years), whereas the mean time for censored patients is 1128 days (3.09 years). Because the distribution of the time is not normal, mean may not be the best characterization of the distribution. For example, the median, geometric mean, mode, of diagonosis-to-death time are 401, 280, ~500 days (or 1.1, 0.77, 1.4 years). Thoese for censored patients are 943, 769, 1778 days (or 2.58, 2.1, 4.9 years).

There are other information which is not available for all samples, including: cancer stage information (available on 80% the samples: 9,28,117,24 persons at stage=1,2,3,4), metastasis status (available on 64% of the samples: 61 persons whose cancer is spread, 81 are not), and treatment information (on 111 adjuvan and 49 neoadjuvan). Either because these variable can be too strong predictor of the survival time or because there are too much missing data in them, they are not used as the predicting factors.

3. Method

3.1. Random Survival Forest

We re-introduce Random Survival Forest in more detail because of its importance in understanding the points we are making in this paper. We have the right-censored data where where the dependent variable being (T_i, δ_i) $(i = 1,2,\cdots n$ is the index for persons), δ_i is the status (1 if dead, 0 if unknown at the end time of data collection), and T_i is either the time to the $\delta_i = 1$ event or the end time of data collection, for $\delta_i = 0$ samples. The independent variable is $X_i = (x_{1i}, x_{2i}, \cdots x_{ki} \cdots)$, where $k = 1,2,\cdots K$ is the index for variables.

1. A randomly-sampling-with-replacement (bootstrap) dataset is produced. This dataset contains n items (same number as the original dataset), but two or more items can be identical because of the "with replacement" requirement. In general, $1 - e^{-1} = 0.632 = 63.2\%$ independent samples from the original dataset are chosen.

2. These n items are used to produce a decision tree, i.e., a series of node at which samples are classified (split into two branches) according to some independent variables, so that there is a maximum discriminative power between the two branches:

   (a) At each node, a randomly selected $K_0 \leq K$ independent variables are used to split the samples;

   (b) Each branch should contain at least an average of n_{min} samples;

(c) Besides #(b), there are other stopping criteria if the discriminative power between the two branches is lower than a threshold.

3. The cumulative hazard function for each person is estimated from the terminal nodes.

4. At each iteration, the error is calculated on those samples that are not chosen by the bootstrap (roughly $e^{-1} = 0.368 = 36.8\%$ of the samples) – called out of bag (OOB) samples.

5. Steps #1 - #4 are iterated N_{rep} times, and the overall error of RSF is an average of N_{rep} errors calculated above.

The first lesson from the above re-introduction is that RSF is an ensemble learner/classifier/data-fitter, each iteration contributes to the overall performance, and the overall error is an average of that in all iterations. Second lesson is that RSF's strategy in training and validation samples partition is different from that in k-fold cross-validation. In bootstrapped training set, some samples are used multiple times, whereas in k-fold cross-validation, one training sample is used only once. The third point to note is that there are several parameters controlling the tree construction process, and it is taken as granted that their default setting lead to the optimal performance.

Let us review parameters described above and their correspondence in the function *rfsrc* in the R package *randomForestSRC*: number of rounds (number of trees), N_{rep} (*ntree* in rfsrc, with default value of 1000; minimum (after averaging over all nodes) number of samples per node before stopping, n_{min} (*nodesize* in rfsrc, with the deault value of 15).

3.2. Programs Used

All survival analyses are carried out by two R statistical packages (*www.r-project.org*), including *survival* [8], *randomForestSRC* [4]. Some analyses in the Discussion section use the R packages *pec* [9] and *ipred* [10].

3.3. Measure of Error: Integrated Brier Score

For binary outcomes, the Brier score [11] (at given time point) is simply the mean square difference between the predicted survival probability $p_i(t)$ and the actual survival situation at that time, if known (index i for samples: $i = 1,2,\cdots n$):

$$BS(t) = \frac{1}{n}\sum_{i=1}^{n} \begin{cases} (p_i(t)-1)^2 & if\ t < t_i \\ p_i(t)^2 & if\ t \geq t_i\ ,\ \delta_i = 1 \\ NA & if\ t \geq t_i\ ,\ \delta_i = 0 \end{cases} \quad (1)$$

Note that when time t passed the censored time t_i, we do not know whether the person survives or not, thus the prediction of the survival of that sample can not be checked (shown as *NA* – not available). The integrated Brier score can be defined as an average of $\mathrm{BS}(t)$ at all available time points in the dataset, T_j ($j = 1,2,\cdots,n_T$), and the integral is approximated by a summation:

$$IBS \approx \frac{1}{n_T}\sum_j BS(T_j) \quad (2)$$

Equation 2 is the "unweighted" IBS: we do not use the weighted IBS because we want to make the calculation as simple as possible.

For *randomForestSRC* R package [4], the object produced by *rfsrc* function contains the predicted survival probability for both the OOB samples (*rfsrc()$survival.oob*) and the "in the bag" training samples (*rfsrc()$survival*), in a single array of length $n \times n_T$ (one sample of length n_T followed by the second sample, etc). The available time points $\{T_j\}$ can be obtained by *rfsrc()$time.interest*

The *survfit* function from *survival* R package [8] can be used to obtain the predicted survival probability. The *surfit(coxph(), newdata=...)$surv* produces a matrix with n-row (n = number of samples) and n_T-column (n_T = number of time points). The *surfit(coxph(), newdata=...)$time* is the list of observed time points. Using the predicted survival probability from the output *survfit/coxph* (from the *survival* package), we wrote our own R function to calculate IBS for Cox regression [12]. The R function and data that support the findings of this study are openly available in github at *github.com/wlicol/coxrsf*.

3.4. Measure of Error: D-index or One Minus the Concordance Index (C-index)

Concordance index used in survival data is defined as the proportion of correct prediction on survival by the model in all pair of samples. It can also be called Harrell's C-index named after the author of the measurement [13]. More specifically, for any two samples where at least one of them has status=1 (e.g. dead), if the survival time for the person whose model-predicted risk for death is higher, then the sample pair is said to be concordant. Percentage of concordant pairs is the C-index. One minus the C-index, or proportion of discordance pairs, is called C-error by Ishwaran et al. (2008) [4], can also be called D-index for discordance rate.

3.5. Parameter Tuning and Setting in RSF

Following the discussions in (e.g.) [5], we examined the impact of the number of trees (N_{rep}, *ntree* in *rfsrc* of the R package *randomForestSRC*), the minimum number of samples (on average) per node before stopping the growth of the tree n_{min}, *nodesize* in *rfsrc* of the R package *randomForestSRC*), and number of variables used to split a tree K_0 (or *mtry* in *rfsrc*). We did not see obvious impact of mtry on performance probably because the number of independent variables (seven) is too small [7]. The impact of ntree and nodesize on RSF performance is shown in Figure 1.

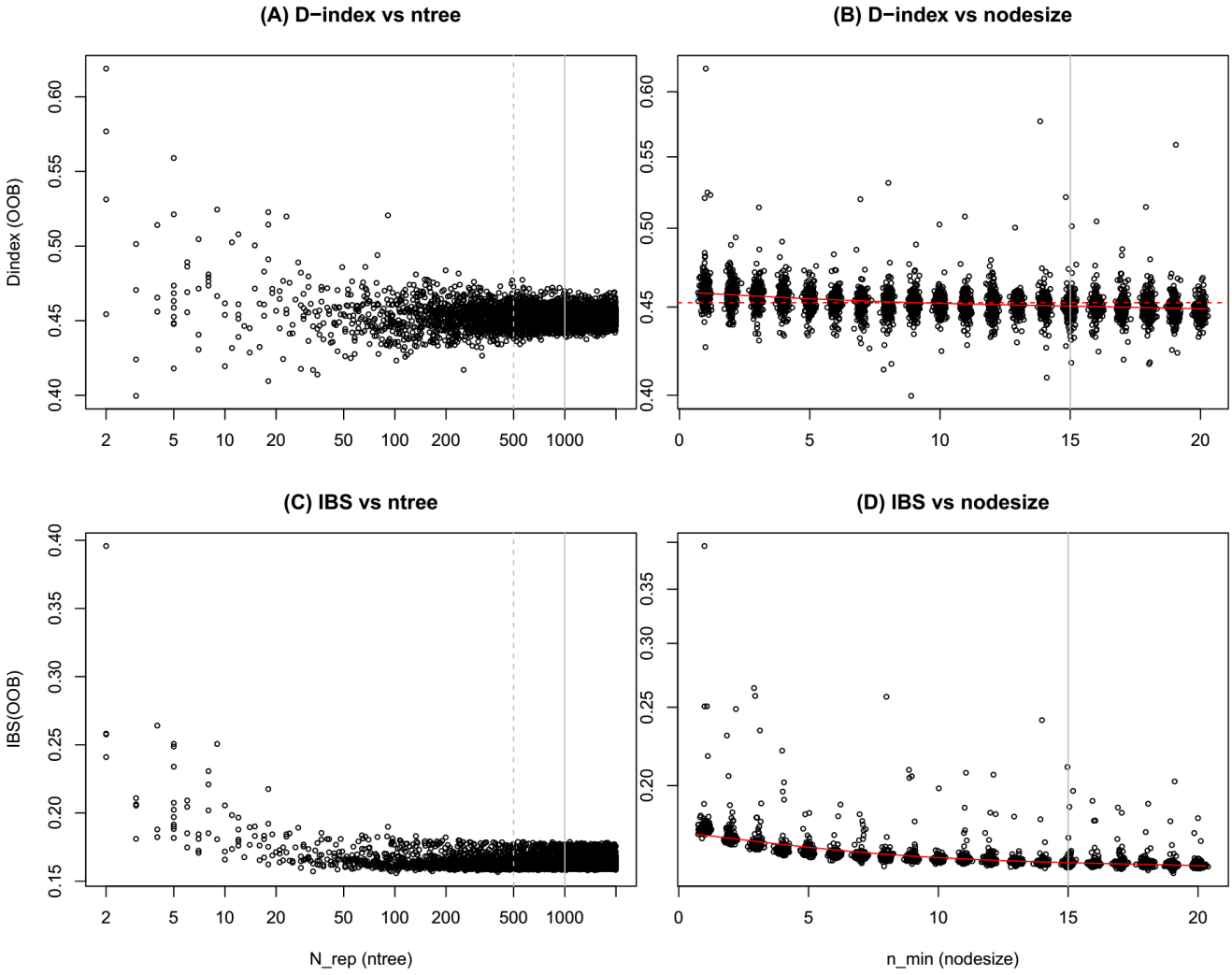

Figure 1. OOB D-index for RSF as a function of (A) number of trees ($ntree$); (B) minimum (after averaging over all nodes) number of samples per node before stopping ($nodesize$); OOB IBS for RSF as a function of (C) $ntree$; and (D) $nodesize$.

Figure 1(A) and (C) shows OOB D-index and IBS as a function of ntree in our colorectal survival data with 7 independent variables. At the default value of ntree=1000, the variation of the error calculation seems to be minimum. At 500 trees, the variation is about the same level. However, when the number of tree is too low (e.g. < 100), we may either get a very small error or a very large one for D-index, and larger error for IBS, by chance.

Figure 1(B) and (D) shows OOB D-index and IBS as a function of *nodesize*. Both are not flat, and there is a gradual decrease of error when $nodesize$ is increased. A larger *nodesize* means a lesser grown tree, and it might prevent tree overfitting. However, if *nodesize* as a percentage of the total sample size n is too large, it may have a chance to fit the data. With the guidence from Figure 1, we chose *ntree*=500 and *nodesize*=15.

Although traditionally, the training set selected is by bootstrap (subsampling with replacement), the default setting in *rfsrc* of *randomForestSRC* is *samptype = "swor"* or sampling without replacement. We changed the setting to swr (sampling with replacement) to be consistent with the original literature. Our conclusion is not affected by using *samptype = "swor"*, which is recommended by some papers because of the availibility of theoretical results [5], but the Cox regression fitting will need to be carried out on the sampling without replacement training set also.

4. Results

4.1. Always Compare Performance on Equivalent Sets

We run RSF 100 times on the colorectal dataset with 7 independent variables and calculate IBS for both OOB samples and IB (in-bag) samples, both directly provided by the *rfsrc* function (see Methods). The number of trees is fixed at 500, and nodesize value is randomly selected from (10-20). Figure 2(A) shows the sorted OOB IBS (black) and IB IBS (blue) from small to large. It is well known that OOB errors is larger than IB errors.

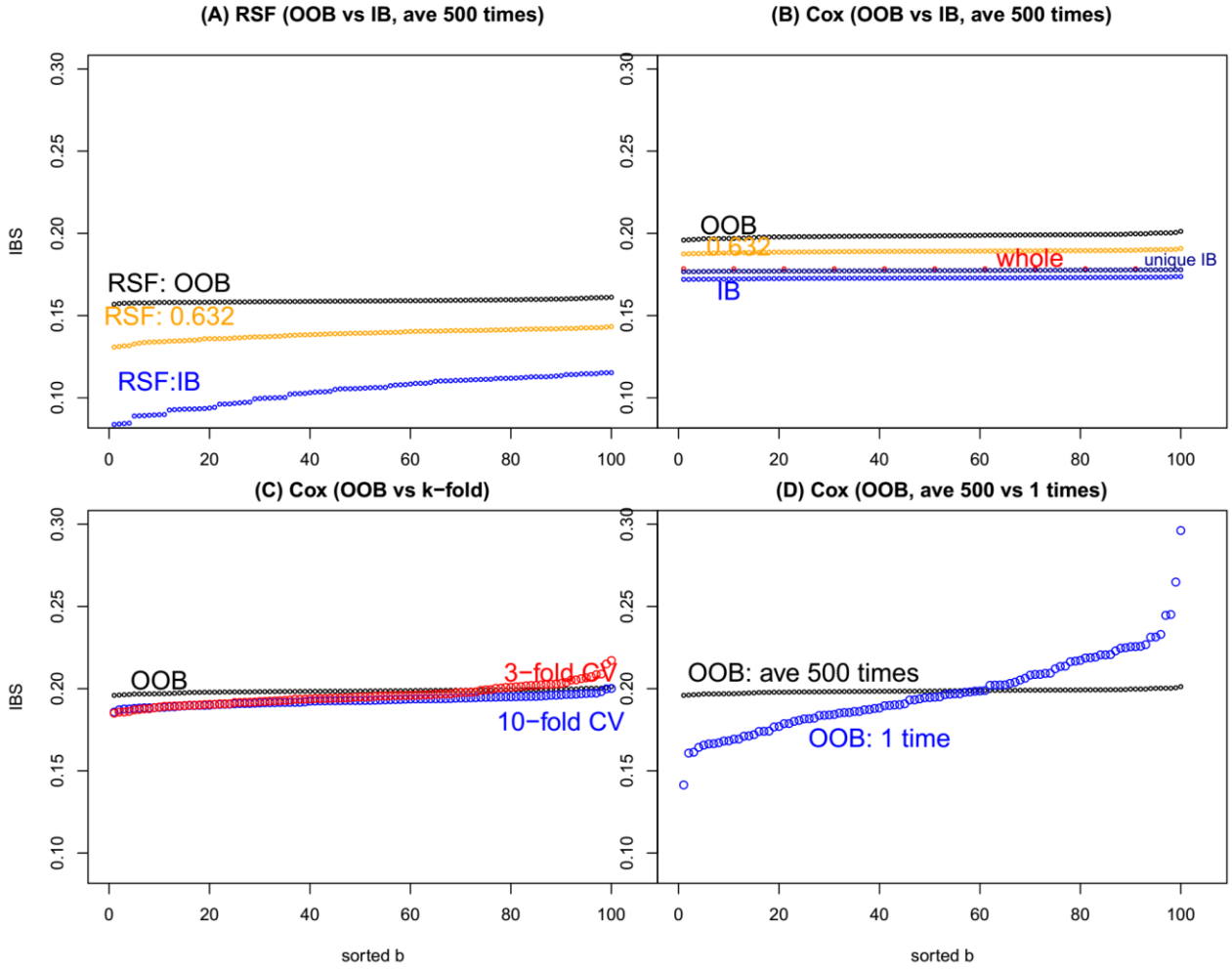

Figure 2. IBS (integrated Brier score) by both Random Survival Forest (RSF) and Cox regression for the colorectal survival dataset in various validation set selections and choice between ensemble or single classifier. (A) RSF: for OOB (black), IB (blue), and weighted of both (0.632 of OOB and 0.368 for IB) (orange); (B) Cox regression: for OOB (black), IB (blue), and 0.632-weight (orange), whole dataset as both training and testing (red), unique IB (dark blue); (C) Cox regression: 3-fold CV(red), 10-fold CV(blue); (D) Cox regression: single classifier OOB (blue). The Cox OOB IBS in (B) is reproduced in (C) and (D) for comparison.

Similarly, we run Cox regression 100 × 500 times, where in each group of 500 runs, a random bootstrap set is used for fitting the regression, which is then applied to the OOB samples to calculate IBS (using our custom R code, see Methods); these 500 IBSs are averaged to get one IBS similar to that for RSF. Again, Figure 2(B) shows that IB (blue) has lower errors than OOB (black).

It was suggested Efron (1983) [14] that the testing error rates can be weighted down by a factor of $1 - e^{-1} = 0.632$, with the rest (weight = $e^{-1} = 0.368$) replaced by training error rates. These weighted down errors when it is done on the forest level for RSF and on individual run level (equivalent to tree level) for Cox regression are shown in Figure 2(A) and (B) (orange). If we compare the IBS between RSF and Cox regression, that on OOB set in RSF should be compared to that on OOB set in Cox regression. Similarly, IB should be compared to IB, and 0.632-weighted error should be compared to another 0.632-weighted error.

Two more types of error are shown in Figure 2(B) for Cox regression. One is to use all samples as both training and testing set (red). As there is no random factor involved, there is only one value. The red line is simply a repetition of the single value 10 times. Another type (we called it “unique IB") is to use each person/sample, if they are sampled

multiple times in IB due to sample-with-replacement procedure, only once towards the error calculation (dark blue). Both are similar to IB error, though slightly higher.

4.2. Performance Determined by OOB Samples and by k-fold Cross Validation may not be the Same for Cox Regression

Most people use a k-fold cross-validation to evaluate the performance of Cox regression. In k-fold cross-validation, the whole dataset is partitioned into k groups: rotating each k-1 groups as training then applied to the kth group for error calculation, the final error is the average of these k errors. The $k = 3$ choice leads to a situation similar to RSF in that 66% of the samples are used for training whereas 33% of the samples for validation. However, there are still two differences: one being that the training set in RSF contains n examples instead of $(k-1)n/k$ for k-fold validation, another being that the final error in RSF is an average of N_{rep} (*ntree*) runs and that in k-fold validation in Cox regression is an average of k runs; and typically $k \ll N_{rep}$.

Figure 2(C) shows (sorted from small to large) IBS of 100 runs for 3-fold validation and 10-fold validation. Both tend to have a lower error than those calculated from OOB samples (average of the 100 runs: 19.86% for OOB, 19.58% for 3-fold CV, 19.28% for 10-fold CV). There have been several publications addressing related issues, though not necessarily for survival data [15, 16]. Whether the choice of OOB overestimates the error or choice of k-fold cross-validation underestimates the error, the message from Figure 2(C) is that when two models are compared, the same validation data selection scheme should be used.

4.3. Only the Ensemble Classifier Version of Cox Regression should be used in a Fair Comparison, not the Single Classifier Version

Since RSF is an ensemble classifier whereas Cox regression can be considered as a single classifier [17-19], our proposal for a fair comparison between the two is to convert Cox regression to an ensemble classifier. It is done by repeated random sampling subset and using the OOB samples for error calculation, then averaging errors from these individual runs.

What if Cox regression remains to be a single classifier? In other words, what if the repeated bootstrap, OOB error calculation is reduced to only one run? Figure 2(D) shows the OOB error for Cox regression without multiple runs. It is clear that without the averaging of multiple (e.g. 500 times) runs, the errors fluctuate widely (standard deviation (sd) for ensemble classifier version of the Cox regression: 0.001, whereas the sd for the single classifier: 0.024). If one use the single classifier version of Cox regression to compare with with RSF, sometimes the conclusion can be Cox regression outperforms RSF, and other times RSF outperforms Cox regression.

4.4. Similar Conclusions using D-index

The conclusions from Figure 3 can be similarly reached by the error calculation of D-index (see Method section). Figure 3(A) shows that the D-index for Cox regression with OOB, IB, unique-IB, as well as using the whole dataset as both training and testing, and D-index for RSF using OOB as testing set. All Cox regression errors as measured by D-index using some samples in both training and testing set are lower than RSF-OOB, but those using OOB have higher errors. A fair comparison would lead to the conclusion that RSF performs better than Cox regression.

Similar to IBS, D-index also shows that k-fold cross-validation for Cox regression leads to lower error than using OOB as testing set. The 10-fold CV has, on average, lower D-index error than 3-fold CV. The conclusion based on a fair comparison, i.e., using OOB error in both RSF and Cox regression, clearly favors RSF over Cox. However, when 10-fold CV error for Cox regression is used to compare the OOB error for RSF, the two two D-index errors are very close.

Finally, same as IBS, the D-index calculated by one-time OOB set for Cox fluctuates wildly from run to run. Single classifier has higher variance for error rate than ensemble classifiers, including both (e.g. 500-time) OOB and 10-fold CV, 3-fold CV.

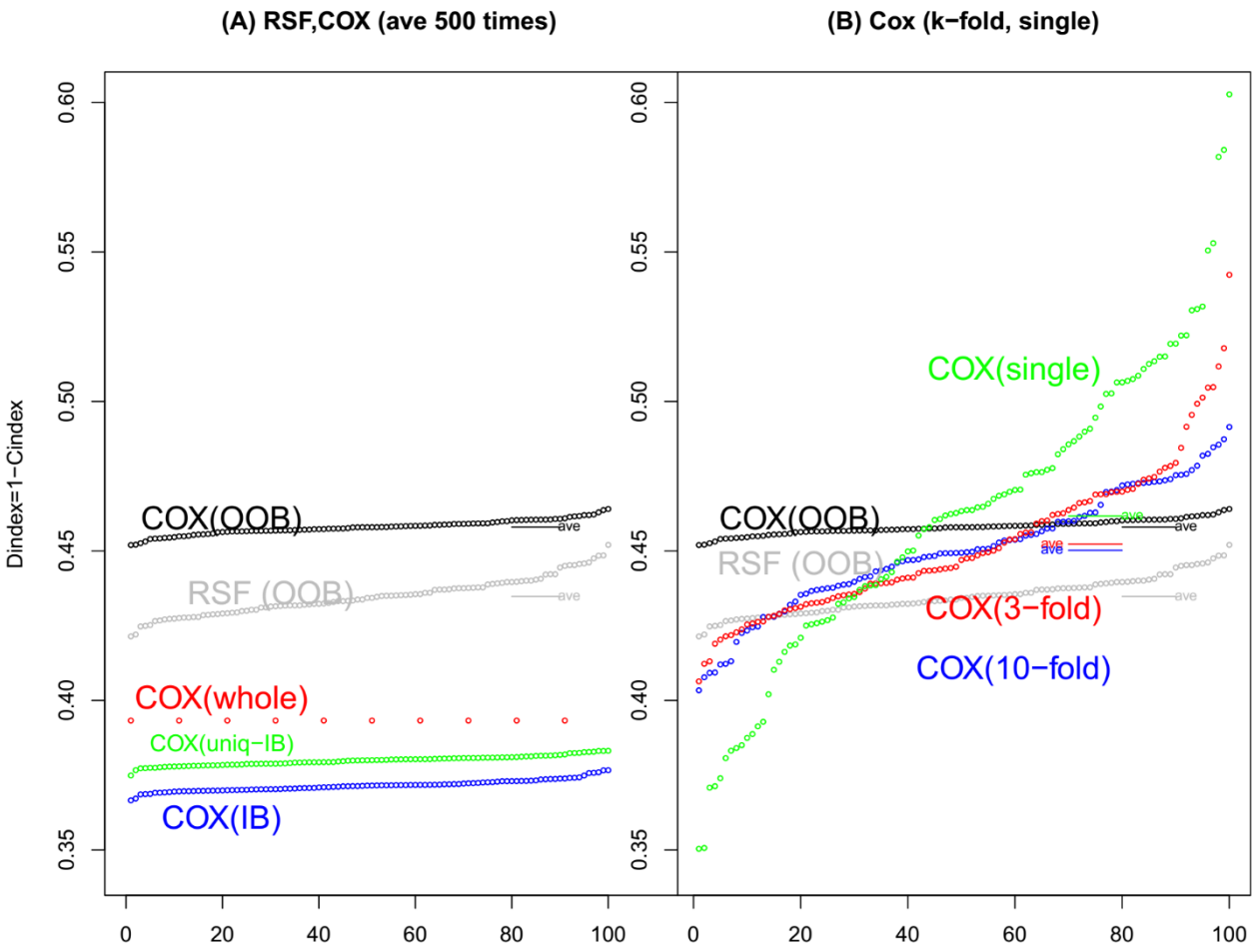

Figure 3. D-index (discordance index, or 1-Cindex, or C-error) by both Random Survival Forest (RSF) and Cox regression for the colorectal survival dataset in various validation set selections and choice between ensemble or single classifier. (A) RSF-OOB (grey), Cox-OOB (black), Cox-IB (blue), Cox-unique-IB (green), Cox-whole (red). The mean of RSF-OOB (grey) and mean of Cox-OOB (black) are shown by two short horizontal bars; (B) Cox-10-fold-CV (blue), Cox-3-fold-CV (red), Cox-OOB-single-classifier (green). The RSF-OOB, Cox-OOB are reproduced in (B) as a comparison. The means of Cox-10-fold-CV (blue), Cox-3-fold-CV (red), and Cox-OOB-single-classifier (green) are shown by short horizontal bars.

5. Discussion

Random survival forest has several advantages over traditional survival analysis method, such as being unlikely to overfit the data with the complete set of independent variables, its easiness in judging the importance of variables for the purpose of variable selection, and its ability to incorporate nonlinear, interactive roles of multiple variables. These are not the topic addressed in the paper.

The motivation of this paper is an observation we have made on existing literature: that when RSF is compared to Cox regression, many papers conclude that the two have similar performance [20, 21], while detail concerning the error calculation (on the Cox regression side) is not provided [22-26]. Some comparisons are even hard to judge because k-fold CV is introduced to RSF [27], or models with different number of independent variables are compared [28]. Steele et al. (2018) and Zhang et al. (2019) [29, 30], k-fold CV is used in evaluating other models including Cox regression, which may be another example of unfair comparison.

On the other hand, when the performance is fairly compared, i.e., when the error is calculated over OOB ensemble for Cox regression, RSF usually is the winner over Cox regression [31, 32]. The practice in Appendix C of Myte (2013) [33] is almost fair except the Cox regression is treated as a single predictor instead of an ensemble classifier. Unfortunately, we found more unfair or potentially unfair comparisons of error in the literature than the correct ones, with the latter more likely carried out in thesis, manual pages, or computer scientists and statisticians than applied researchers. This justifies the need to highlight this issue, i.e., the test set where the error rate or performance is calculated, should be equivalent when comparing two different data analysis techniques.

Although there is a professional written R packages *pec* [9] for evaluating IBS error in different situations, the aim of the package is to be as generic and wide as possible. To make the program to work as desired and to understand most detail may require a steep learning curve for practitioners whose main expertise is in a non-mathematical/non-

statistical field. We try to fill this void by providing a simply written R codes for calculating only IBS/RSF (OOB), IBS/Cox (new samples), and D-index/Cox (new samples). The D-index/RSF (OOB) is the default output of error rate in *randomForestSRC* package). The R code is in *github.com/wlicol/coxrsf*.

Our codes only require a list of predicted survival probability of each person at sequence of time of interest [34]. For RSF, we use the *survival.oob* part of a rfsrc object to obtain the survival probability. For Cox regression, we use the *survfit* function from the *survival* R package (the *surv* part) to obtain the survivial probability.

To double check our program, Figure 4 shows a comparison of Brier score, estimated by using the fitted RSF on the whole dataset itself (not OOB), as a function of time, between our code and two other programs, *pec* and *sbrier/ipred*. We found that (1) the *pec* and *sbrier/ipred* results are very similar, as they both use the inverse probability of censoring (IPC) weights [3, 4]. (2) Our simple code without using any weights lead to very similar result as the weighted BS's. (3) Our program has an extra option to print out BS(t) contributed from deceased samples (status=1) and cencored ones (status=0) separately. Figure 4 shows a huge difference between the two.

It was pointed out that when the independent variables contain both discrete and continuous variables, continuous variables may have an advantage to be chosen more often as the variable to split trees in RF (not RSF specific) [5]. This may rank some continuous variable ahead of categorical variables, and if the categorical are actually more important, underestimate the classifier performance [35]. This issue does not affect the analysis in this paper because our categorial variables (cancer type and gender) are not actually important. Also, this bias potentially underestimates the performance of RSF, so correcting the bias would make RSF even better than Cox regression.

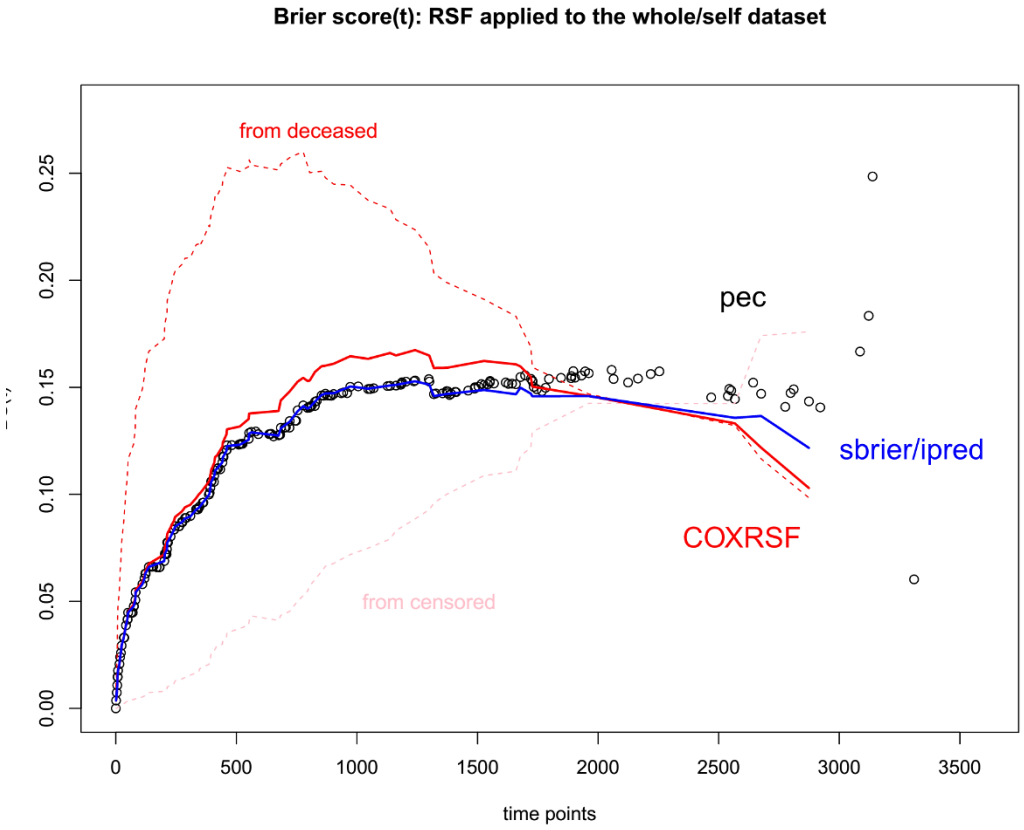

Figure 4. Brier score as a function of time for colorectal data calculated by two public domain programs and our simple R code. This is using the predicted survival function from the trained RSF to the whole dataset (not OOB). (black) from the *pec* program in the *pec* R package; (blue) from the *sbrier* program of the *ipred* R package; (red solid) from our own simple R codes, where the contribution from the deceased samples (status=1) is in red dashed lines, and contribution from the censored samples (status=0) is in pink dashed line.

6. Conclusion

In conclusion, in the current literature of application of random survival forest to real data, the majority of them may not compare the performance of RSF and Cox regression fairly, when the error (D-index or IBS) is not calculated on OOB samples, and/or without enough number of repeated trainings/testings. The k-fold CV may increase the performance of Cox regression over OOB, and if k is too small, the error rate may vary from run-to-run. We provide simple R code to aid the practive of fair comparison between RSF and Cox regression.

7. Abbreviations

BS:	Brier Score	IB:	In-bag (samples)
C-index:	Concordance index	IBS:	Integrated Brier Score
CV:	Cross-validation	OOB:	Out-of-bag (samples),
D-index:	Discordance index	RSF:	Random Survival Forest

8. Declarations

8.1. Author Contributions

S.C., A.U. and W.L.: Conception of the idea, analysis of the results, writing the draft; I.D.: Data collection and interpretation of clinical results. All authors have read the final version of the article and approved.

8.2. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

8.3. Acknowledgements

Wentian Li acknowledges the support from the Robert Boas Center for Genomics and Human Genetics.

8.4. Ethical Approval

Ethical approval was obtained from the Hatay Mustafa Kemal University, Faculty of Medicine on July 6, 2020 with the protocol number 32/28.

8.5. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

8.6. Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

9. References

[1] Wang, P., Li, Y., & Reddy, C. K. (2019). Machine Learning for Survival Analysis. ACM Computing Surveys, 51(6), 1–36. doi:10.1145/3214306.

[2] Breiman L (2001). Random forests, Machine Learning, 45, 5-32. doi:10.1023/A:1010933404324.

[3] Hothorn T., Bühlmann P., Dudoit S., Molinaro A., Van Der Laan M.J. (2006). Survival ensembles, Biostatistics, 7(3), 355–373. doi:10.1093/biostatistics/kxj011.

[4] Ishwaran, H., Kogalur, U. B., Blackstone, E. H., & Lauer, M. S. (2008). Random survival forests. The Annals of Applied Statistics, 2(3). doi:10.1214/08-aoas169.

[5] Boulesteix, A.-L., Janitza, S., Kruppa, J., & König, I. R. (2012). Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2(6), 493–507. doi:10.1002/widm.1072.

[6] Scornet, E. (2017). Tuning parameters in random forests. ESAIM: Proceedings and Surveys, 60, 144–162. doi:10.1051/proc/201760144.

[7] Probst, P., Wright, M. N., & Boulesteix, A. (2019). Hyperparameters and tuning strategies for random forest. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(e1301). doi:10.1002/widm.1301.

[8] Therneau, T. M., & Grambsch, P. M. (2000). The Cox Model. Modeling Survival Data: Extending the Cox Model, 39–77. doi:10.1007/978-1-4757-3294-8_3.

[9] Mogensen, U. B., Ishwaran, H., & Gerds, T. A. (2012). Evaluating Random Forests for Survival Analysis Using Prediction Error Curves. Journal of Statistical Software, 50(11). doi:10.18637/jss.v050.i11.

[10] Peters, A., Hothorn, T., & Lausen, B. (2002). ipred: Improved predictors. R News 2 (2): 33–36.

[11] Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. Monthly weather review, 78(1), 1-3. doi:10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

[12] Cetin S, Ulgen A, Dede I, Li W. (2020) COXRSF: R function to calculate IBS or D-index for Cox regression and random survival forest. Available online: http://github.com/wlicol/coxrsf. (accessed on 20 March 2021).

[13] Harrell, F. E. (1982). Evaluating the yield of medical tests. JAMA: The Journal of the American Medical Association, 247(18), 2543–2546. doi:10.1001/jama.247.18.2543.

[14] Efron, B. (1983). Estimating the Error Rate of a Prediction Rule: Improvement on Cross-Validation. Journal of the American Statistical Association, 78(382), 316–331. doi:10.1080/01621459.1983.10477973.

[15] Mitchell, M. W. (2011). Bias of the Random Forest Out-of-Bag (OOB) Error for Certain Input Parameters. Open Journal of Statistics, 01(03), 205–211. doi:10.4236/ojs.2011.13024.

[16] Janitza, S., & Hornung, R. (2018). On the overestimation of random forest's out-of-bag error. PLOS ONE, 13(8), e0201904. doi:10.1371/journal.pone.0201904.

[17] Kittler, J. (1998). Combining classifiers: A theoretical framework. Pattern Analysis and Applications, 1(1), 18–27. doi:10.1007/bf01238023.

[18] Kittler, J., Hatef, M., Duin, R. P. W., & Matas, J. (1998). On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(3), 226–239. doi:10.1109/34.667881.

[19] Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. Proc. 1st Int. Workshop on Multiple Classifier Systems (MCS00), Lecture Notes in Computer Science, 1857, 1–15. doi:10.1007/3-540-45014-9_1.

[20] Kurt Omurlu, I., Ture, M., & Tokatli, F. (2009). The comparisons of random survival forests and Cox regression analysis with simulation and an application related to breast cancer. Expert Systems with Applications, 36(4), 8582–8588. doi:10.1016/j.eswa.2008.10.023.

[21] Datema, F. R., Moya, A., Krause, P., Bäck, T., Willmes, L., Langeveld, T., … Blom, H. M. (2011). Novel head and neck cancer survival analysis approach: Random survival forests versus cox proportional hazards regression. Head & Neck, 34(1), 50–58. doi:10.1002/hed.21698.

[22] Kwamboka Mageto, D. (2015). Modelling of Credit Risk: Random Forests versus Cox Proportional Hazard Regression. American Journal of Theoretical and Applied Statistics, 4(4), 247. doi:10.11648/j.ajtas.20150404.13.

[23] Zhou, L., Xu, Q., & Wang, H. (2015). Rotation survival forest for right censored data. PeerJ, 3, e1009. doi:10.7717/peerj.1009.

[24] Saadati, M., & Bagheri, A. (2019). Comparison of Survival Forests in Analyzing First Birth Interval. Jorjani Biomedicine Journal, 7(3), 11–23. doi:10.29252/jorjanibiomedj.7.3.11.

[25] Kim, D. W., Lee, S., Kwon, S., Nam, W., Cha, I.-H., & Kim, H. J. (2019). Deep learning-based survival prediction of oral cancer patients. Scientific Reports, 9(1). doi:10.1038/s41598-019-43372-7.

[26] Ma, B., Geng, Y., Meng, F., Yan, G., & Song, F. (2020). Identification of a Sixteen-gene Prognostic Biomarker for Lung Adenocarcinoma Using a Machine Learning Method. Journal of Cancer, 11(5), 1288–1298. doi:10.7150/jca.34585.

[27] Nicolò, C., Périer, C., Prague, M., Bellera, C., MacGrogan, G., Saut, O., & Benzekry, S. (2019). Machine learning and mechanistic modeling for prediction of metastatic relapse in early-stage breast cancer. doi:10.1101/634428.

[28] Nasejje, J. B., & Mwambi, H. (2017). Application of random survival forests in understanding the determinants of under-five child mortality in Uganda in the presence of covariates that satisfy the proportional and non-proportional hazards assumption. BMC Research Notes, 10(1). doi:10.1186/s13104-017-2775-6.

[29] Steele, A. J., Aylin Cakiroglu, S., Shah, A. D., Denaxas, S. C., Hemingway, H., & Luscombe, N. M. (2018). Machine learning models in electronic health records can outperform conventional survival models for predicting patient mortality in coronary artery disease. doi:10.1101/256008.

[30] Zhang, X., Tang, F., Ji, J., Han, W., & Lu, P. (2019). Risk Prediction of Dyslipidemia for Chinese Han Adults Using Random Forest Survival Model. Clinical Epidemiology, Volume 11, 1047–1055. doi:10.2147/clep.s223694.

[31] Miao, F., Cai, Y.-P., Zhang, Y.-T., & Li, C.-Y. (2015). Is Random Survival Forest an Alternative to Cox Proportional Model on Predicting Cardiovascular Disease? 6th European Conference of the International Federation for Medical and Biological Engineering, 740–743. doi:10.1007/978-3-319-11128-5_184.

[32] Kantidakis G (2018). Prediction Models for Liver Transplantation (Master Thesis, Statistical Science, Universiteit Leiden). Available online: www.universiteitleiden.nl/binaries/content/assets/science/mi/scripties/statscience/2018-2019/2018_10_29_masterthesis_kantidakis.pdf (accessed on 12 Mach 2021).

[33] Myte R (2013). Covariate Selection for Colorectal Cancer Survival Data (Bachelor thesis, Umeå University). Available online: https://www.diva-portal.org/smash/get/diva2:627337/FULLTEXT01.pdf (accessed on 18 Mach 2021).

[34] Wang J (2018). Apply Machine Learning Approaches to Survival Data (project report, Dept of Computing, Imperal College London). Available online: https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/1718-ug-projects/Jingya-Wang-Applying-machine-learning-approaches-to-survival-data.pdf (accessed on 2 April 2021).

[35] Wright, M. N., Dankowski, T., & Ziegler, A. (2017). Unbiased split variable selection for random survival forests using maximally selected rank statistics. Statistics in Medicine, 36(8), 1272–1284. doi:10.1002/sim.7212.