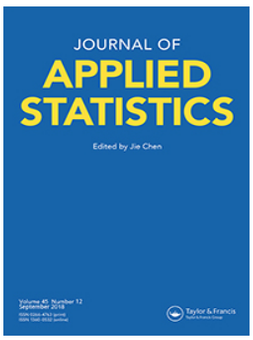

Journal of Applied Statistics

ISSN: 0266-4763 (Print) 1360-0532 (Online) Journal homepage: https://www.tandfonline.com/loi/cjas20

Predicting risk for adverse health events using random forest

Guy Cafri, Luo Li, Elizabeth W. Paxton & Juanjuan Fan

To cite this article: Guy Cafri, Luo Li, Elizabeth W. Paxton & Juanjuan Fan (2018) Predicting risk for adverse health events using random forest, Journal of Applied Statistics, 45:12, 2279-2294, DOI: 10.1080/02664763.2017.1414166

To link to this article: https://doi.org/10.1080/02664763.2017.1414166

JOURNAL OF APPLIED STATISTICS, 2018
VOL. 45, NO. 12, 2279–2294
https://doi.org/10.1080/02664763.2017.1414166

Taylor & Francis
Taylor & Francis Group

Predicting risk for adverse health events using random forest

Guy Cafri[a], Luo Li[b,c], Elizabeth W. Paxton[a] and Juanjuan Fan[c]

[a]Surgical Outcomes and Analysis, Kaiser Permanente, San Diego, CA, USA; [b]Computational Science Research Center, San Diego State University, San Diego, CA, USA; [c]Department of Mathematics and Statistics, San Diego State University, San Diego, CA, USA

ABSTRACT

Estimation of person-specific risk for adverse health events in medicine has been approached almost exclusively using parametric statistical methods. Random forest is a machine learning method based on tree ensembles that is completely nonparametric and for this reason may be better suited for risk prediction. An introduction to a random forest is provided with a focus on its application to risk prediction. Using data from a total joint replacement registry, we illustrate risk prediction for the binary outcome of 90-day mortality following implantation, as well as time to device failure for aseptic reasons with the competing risk of mortality. Using the methods described in this paper, the random forest could be applied to risk prediction in a wide variety of medical fields. Issues related to implementation are discussed.

ARTICLE HISTORY
Received 22 March 2017
Accepted 1 December 2017

KEYWORDS
Machine learning; nonparametric; random forest; survival; competing risk; total knee replacement

1. Introduction

Prediction of a person's risk for adverse health events, such as mortality and cardiovascular disease, has become increasingly popular. Systematic reviews have identified 110 different risk-scoring methods for cardiovascular disease [2] and 145 risk models for type II diabetes [25] alone. The implementation of risk-scoring algorithms is critically important in disseminating information to patients and clinicians. Among the most well-known risk prediction tools are those related to heart disease [2,30] and breast cancer [13,14], although risk prediction is applied in many fields. Many methods are available for constructing risk estimates, although the majority are based on fairly restrictive statistical models. The challenge is that conventional parametric statistical models require explicit specification of how the variable inputs or predictors relate to the outcome of interest. By contrast, machine learning methods are nonparametric in nature and this flexibility can lead to improved prediction accuracy. One impediment to the adoption of these methods among statisticians may be a lack of familiarity, therefore in this paper we describe and illustrate one method, random forest [4], which has been shown to have excellent predictive accuracy when compared to a large number of machine learning methods [7,8].

To better appreciate the strength of the method being proposed, it is important to consider some of the limitations associated with past approaches. Parametric approaches

CONTACT Guy Cafri guycafri@gmail.com

typically make strict assumptions about the data. Consider the multiple linear regression model as an example: $Y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} + \varepsilon_i$ for $i = 1, 2, \ldots, N$. Here $x_{i1}, \ldots, x_{ik}$ are the values on the k explanatory variables for the ith observation, $\beta_0, \ldots, \beta_k$ are the parameters to be estimated, and $\boldsymbol{\varepsilon_i}$ is a random error typically assumed to be distributed as $N(0, \sigma^2)$. A critical assumption of this model is that the variables are constrained to have linear and additive effects on the response. While a regression model can incorporate nonlinear and interactive effects, the specified form of such effects can be restrictive, and with a growing number of variables accurate model specification becomes increasingly difficult. By comparison, a nonparametric approach makes no assumptions about the relationship between the predictors and response, $Y_i = f(x_{1i}, \ldots, x_{ki}) + \varepsilon_i$ for $i = 1, 2, \ldots, N$ where $f(x_{1i}, \ldots, x_{ki})$ is a function determined entirely by the data without any imposed structure. This feature enables a nonparametric approach to better capture intricate relationships between adverse outcomes and their predictors. For instance, Body Mass Index (BMI) is a commonly used predictor in various mortality risk calculations. However, the relationship between BMI and mortality may be complex. People who are underweight and obese are both associated with increased risk of mortality [1]. Additionally, smoking decreases BMI, but increases mortality risk from other causes [1,11]. The possible nonlinear effects of BMI and interactions with other predictors, such as smoking, would need to be specified in a parametric regression model, but would be incorporated automatically into constructing predictions when using random forest.

Current prediction tools are most commonly created using logistic regression or parametric/semi-parametric survival models [2,28]. These methods rely on correct specification of measured variables, and the prediction accuracy may be compromised when model assumptions are violated. Random forest is an appealing alternative because it does not rely on a model to be specified. Moreover, random forest can mitigate overfitting and provide unbiased estimates of prediction accuracy through its use of the bootstrap and corresponding left-out samples. Random forest also provides a means to reduce dimensionality and quantify relative prediction strength of individual variables using variable importance measures. Lastly, random forest may be particularly appealing for rare event data because it avoids problems with convergence, high variance, and overfitting that can be encountered with parametric/semi-parametric models.

While the authors are not aware of the use of random forest for risk prediction in medical applications, a growing literature has used random forest for estimating causal effects in medicine [9,12,29]. The purpose of this article is to introduce and illustrate the application of RF for risk calculation. Illustrations are provided for mortality and time to device failure following primary elective total knee replacement.

2. Methods

2.1. Introduction to random forest

Random forest (RF) [4] is a nonparametric approach based on decision trees. When more than one tree is used, it is called a tree ensemble or forest. Depending on the nature of the outcome, classification, regression, or survival trees are used. The motivating factor for using a forest as opposed to a single tree is improved prediction accuracy [18]. Prediction

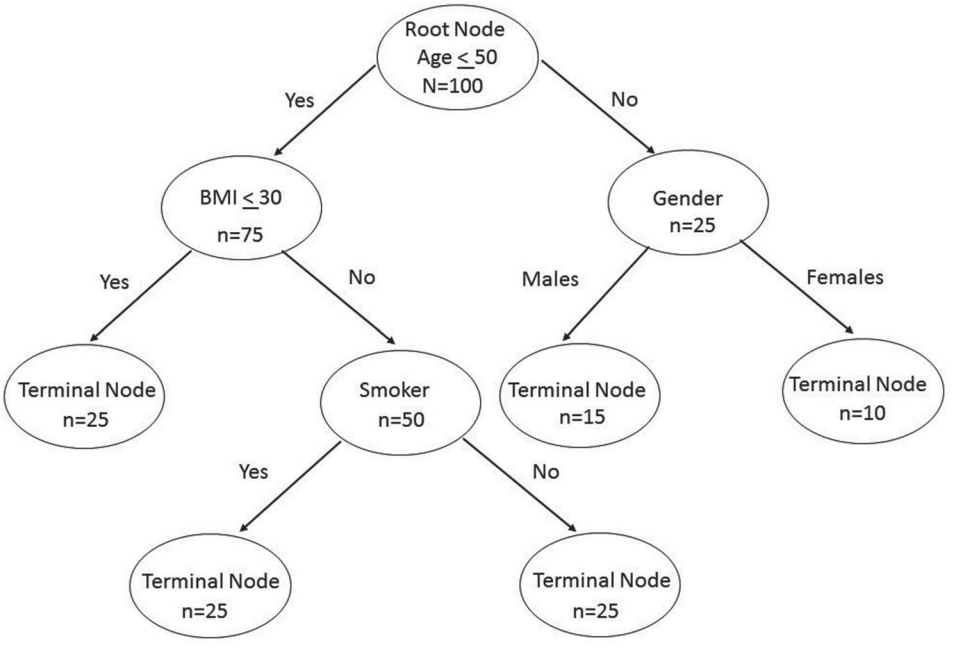

Figure 1. Hypothetical tree.

accuracy is determined by both bias and variance of an estimator. Individual trees have low bias but high variance, the latter mitigated by averaging predictions of trees grown on different bootstrap samples of the data [3]. In addition to bootstrapping, at each split of the data a random subset of predictors is considered as possible candidates for the splitting variable, which further reduces the variance by de-correlating the trees [4].

The tree growing process for any one tree in a forest is based on recursively partitioning the data using the best binary split until some criteria are met. The approach was originally described by Breiman *et al.* [5] for classification and regression trees. Specifically, a tree is grown by splitting the data at the root node into two child nodes that maximize between-node heterogeneity or minimize within-node impurity (see Figure 1). The same procedure is repeated for each child node until reaching a point where further splitting no longer decreases the impurity or a predetermined stopping rule is reached. A node that cannot be split any further is called a *terminal node*. Each terminal node is a distinct partition of the sample based on the input variables. In other words, each terminal node is characterized by a unique combination of the attributes of an observation or patient characteristics. Given a set of patient characteristics, predictions can be based on a summary of the outcome (e.g. mean) in each terminal node. Below we provide an overview of the RF algorithm, followed by a description of how RF uses its collection of trees to construct predictions.

The RF algorithm is based on the following:

(1) Sample with replacement the size of the original data (bootstrap). On average 63% of the original sample will be included in the bootstrap sample and 37% will be left out.
(2) A tree is grown on the bootstrapped data.

(a) At each node, randomly select a subset m of the total k predictors to consider splitting the data on.
(b) Among the m predictors, select the split that minimizes the within-node error or maximizes the between-node heterogeneity.
(c) Repeat steps a and b until between-node heterogeneity/within-node impurity ceases to improve or a stopping rule is reached (e.g. minimum node sample size needed to partition the data).

(3) Repeat steps 1 and 2 for as many bootstrap samples/trees as desired.

Once an RF is constructed using the above steps, predictions are based on averaging the predicted values from each tree in the forest. RF has a built in tool for evaluating prediction accuracy that avoids overly optimistic estimates of accuracy because the data used to build the RF is separate from the data used to evaluate its accuracy. Specifically, each tree in an RF is constructed from a subsample of the data (due to bootstrapping) known as 'in-bag' data, the left over observations known as 'out-of-bag' data (not used to construct each tree) are used to evaluate prediction accuracy.

2.2. Random forest for binary endpoints

For binary outcomes, prediction is based on the probability of experiencing the event of interest. For this problem, we propose using regression RF as opposed to classification RF, given the way most classification RF algorithms construct predictions from terminal nodes. For classification RF, predictions in terminal nodes are based on majority vote. That is, the majority of patients in the terminal node determine if all patients in the node are predicted to have an event or not. The predicted probability of a new patient experiencing an event is calculated as an average of the 0/1 classifications from all trees in the forest. In contrast, with regression RF predictions in an individual tree are not coarsened into a 0/1, rather they average the responses in the terminal node, resulting in predictions that can range from 0 to 1 [24] (see also Appendix 1 for limited simulations supporting use of regression over classification RF). The predicted probability of a new patient experiencing an event is estimated by the average of the probabilities from all trees in the forest.

The extent of impurity or variability in a regression RF node (T) can be characterized by its variance:

$$\hat{\sigma}_T^2 = \frac{1}{N}\sum_{i\in T}(y_i - \bar{y})^2, \tag{1}$$

where observed values on the response are denoted by y_i $(i = 1 \ldots N)$ and $\bar{y} = (1/N)\sum_{i=1}^{N} y_i$.

The change in the variance in the node as a result of a binary split can be expressed as the difference in the variance in the original node (T) and its left (TL) and right child nodes (TR):

$$\hat{\Delta} = \hat{\sigma}_T^2 - (\hat{\sigma}_{TL}^2 + \hat{\sigma}_{TR}^2), \tag{2}$$

where $\hat{\sigma}_{TL}^2 = (1/N_L)\sum_{i\in TL}(y_i - \bar{y}_{TL})^2, \hat{\sigma}_{TR}^2 = (1/N_R)\sum_{i\in TR}(y_i - \bar{y}_{TR})^2$

A split that minimizes $\hat{\sigma}^2_{TL} + \hat{\sigma}^2_{TR}$, or equivalently maximizes $\hat{\Delta}$, is ideal because it leads to nodes with less error variability. Our implementation of a splitting rule is a weighted version of this difference, such that the variance in the left and right child nodes are each weighed by the proportion of observations they contain, which performs better than the unweighted version [19]:

$$\hat{\Delta}_w = \hat{\sigma}^2_T - \left(\frac{N_L}{N}\hat{\sigma}^2_{TL} + \frac{N_R}{N}\hat{\sigma}^2_{TR}\right). \tag{3}$$

When considering the accuracy of RF predictions, the Brier score and area under the curve are considered. The Brier score is the average-squared difference between the observed and predicted values of the outcome:

$$\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{4}$$

where observations in the data are denoted by y_i ($i = 1 \ldots N$). Prediction for a single regression tree is based on the terminal node mean, in this case the proportion of 1's, a value that is averaged over trees to obtain $\hat{y}_i$. Notably, predictions are based on using trees grown without y_i (i.e. out-of-bag), but using the input values of an observation to calculate a prediction for that case in each tree. We use the out-of-bag data so that the estimate of prediction accuracy is not positively biased as a result of both building a 'model' and evaluating its performance on the same data. Another measure that can be used to evaluate accuracy is area under the receiver operating characteristic (ROC) curve. The ROC curve displays the rate of true positives (x-axis) and false positives (y-axis) as the cutpoint for making a prediction about the absence or presence of an event is altered. The area under this curve represents the extent to which for a randomly chosen pair of individuals, one with an event and one without, the RF assigns a higher probability of experiencing the event to the individual that experienced the event. As with the Brier score, our calculation of the area under the curve involves only predictions based on out-of-bag data.

2.3. Random forest for survival endpoints

For time-to-event outcomes subject to right censoring survival RF is used [21]. In this setting trees are grown using logrank splits. Let $t_1 < t_2 < \cdots < t_J$ be the J ordered failure times ($j = 1, \ldots, J$) in the parent node h, with d_{jc} and Y_{jc} denoting the number of failures and number at risk, respectively, at time t_j in child nodes $c = 1,2$. Define $d_j = d_{j1} + d_{j2}$ and $Y_j = Y_{j1} + Y_{j2}$. The logrank test is then:

$$\frac{\left[\sum_{j=1}^{J}\left(d_{j1} - \frac{d_j}{Y_j}Y_{j1}\right)\right]^2}{\sum_{j=1}^{J}\frac{Y_{j1}}{Y_j}\left(1 - \frac{Y_{j1}}{Y_j}\right)\left(\frac{Y_j - d_j}{Y_j - 1}\right)d_j}. \tag{5}$$

At each internal node, the best split maximizes this logrank statistic among all splits considered based on the m randomly selected predictors. In some research contexts, the presence of an intervening or competing event may preclude the event of interest from taking place.

For instance, if the endpoint of interest is time to knee replacement failure, death would represent a competing event. Particularly when the objective is to aid in patient decision making, incorporating the effect of the competing risk of death is relevant [16]. Further, a modification to the logrank test that accommodates competing risks is possible by adapting Y_j and Y_{j1} in Equation (5) [20]. When evaluating survival endpoints it is commonplace to report the probability of not experiencing an event past a fixed time or its complement. In contrast, with competing risks it is conventional to report the cumulative incidence function, which is the probability of experiencing a failure by a fixed time [16]. In either case, predictions are based on averaging across terminal nodes of trees in the forest. When considering the accuracy of tree predictions, 100*(1-concordance index) is reported [17]. The concordance index has an interpretation similar to area under the ROC curve previously described, accuracy of ranking pairs of randomly selected individuals in terms of their survival on the outcome of interest in the out-of-bag data [20].

2.4. Tuning a random forest

An important aspect of random forest is the need to specify values for tuning parameters of the algorithm, as this can impact the prediction accuracy. Typically, these include the number of variables considered at each split and the minimum number of observations and/or number of events in a terminal node. Trees tend to favor splits on variables with a greater number of response options simply because more splits are being considered [23]. To minimize this type of bias, the effect of variables can be curtailed by setting a limit to the number of random split points considered for each variable [15]. This adds another tuning parameter that might be considered. Moreover, this adds a level of randomization into the tree growing process, which further de-correlates the trees and thereby improve prediction accuracy.

2.5. Variable selection using random forest

There are two important reasons to consider variable selection when developing risk predictions. First, limiting the number of inputs a user has to supply may increase utilization of a prediction tool. Second, elimination of some variables that are not predictive may improve prediction accuracy. There are two popular ways of doing variable selection with random forest, permuted importance and minimal depth. Permuted importance for a particular variable is calculated by comparing the difference in the prediction accuracy of the out-of-bag data to the prediction accuracy when the variable is noised up by randomly permuting its values [4]. For each variable, these values are averaged over all trees in a forest. The larger the importance value, the higher the predictive power of the variable. A small or negative variable importance value indicates low or no predictive power. Variables are sorted from high to low according to their permuted importance score and those with the highest scores are selected. An alternative approach for variable selection is minimal depth, which assesses the predictive power of a variable using in-bag data based on the distance from the root node of a tree to the first split on that variable [22]. For each variable, the values are averaged over all trees in a forest. Smaller minimal depth indicates a variable is more predictive. Variables are sorted from low to high according to their minimal depth score and those with the lowest scores are selected.

2.6. Missing data imputation using random forest

Often predictors of an outcome will have missing data. Missing data on the predictors can be handled using iterative adaptive tree imputation [21]. For a single tree, this approach takes a random draw from the in-bag data whenever an observation with a missing value on a variable is encountered in a candidate split (however, split statistics are based on the non-imputed data only). After splitting the data based on the best identified split, the imputed data are reset to missing and the aforementioned process is repeated whenever missing data are encountered in subsequent splits. The process is repeated for each tree in a forest. To obtain the imputed value for the in-bag data, an average is calculated across terminal nodes in the forest for continuous variables and the most frequently occurring value for nominal variables. The process can be iterated, with further details described elsewhere [21].

3. Data example: predicting risk of device failure and mortality following knee replacement

3.1. Data source

Data are from the Kaiser Permanente's Total Joint Replacement Registry (KPTJRR). KP is an integrated health-care system in eight regions of the United States. KPTJRR uses both paper and electronic data collection processes to capture patient characteristics, implant and surgical information [26,27]. Patients undergoing elective primary total knee replacements (TKA) are used. The outcomes of interest are 90-day mortality and time to device failure for aseptic reasons. All primary TKAs ($N = 74{,}665$) registered from 01/01/2007 to 12/30/2013 were included in the study for the risk of mortality within 90 days. All primary TKAs registered from 04/01/2001 to 12/30/2013 ($N = 110{,}796$) were included in the study for the risk of revision. During the study period, .26% of patients died within 90 days after TKA (197 out of 74,665), and 1.56% (1728 out of 110,796) experienced a device failure with 6.5% (7229 out of 110,796) experiencing a death (competing risk) prior to device failure. Thirty-five risk factors were considered for predicting 90-day mortality, these include common patient factors (e.g. age), as well as many relevant comorbidities (e.g. depression) [10] (Table 1). Thirty candidate risk factors for predicting device failure were considered and included common patient factors, surgical factors, and implant characteristics (Table 2).

3.2. Statistical analysis

For several of the predictor variables, there were observations with missing values. Therefore, an initial step was missing data imputation. The number of missing data iterations was set to five. The number of trees in the RF used for imputation was 500, but otherwise 1000 trees per RF were used. For each tree grown we considered the conventional fixed numbers of predictors at each split, $k/3$ (regression RF) or $\sqrt{k}$ (survival RF). For variables with more than two response options, we limited the number of random split points considered for each variable to two. With respect to tuning the RF, we focused on manipulating the minimum number of observations in a terminal node, given the rare nature of events larger terminal nodes might be expected to yield more accurate predictions. We also examined the effect of varying the number of variables considered at each

Table 1. Minimal depth and permuted importance for all variables predicting 90-day mortality.

	Minimal depth	Permuted importance
Age	2.331	0.091
Fluid and electrolyte disorders	3.606	0.010
Congestive heart failure	3.739	0.012
Peripheral vascular disease	4.072	0.007
ASA score	4.349	0.014
Gender	5.059	0.034
Deficiency anemias	5.343	0.005
Type of procedure	5.460	0.016
Weight loss	5.841	0.006
BMI	5.854	0.018
Pulmonary circulation disease	6.326	0.005
Tumor	6.574	0.003
Race	6.984	−0.002
Hypothyroidism	7.414	0.005
Chronic pulmonary disease	7.875	0.002
Diabetes (w/out complications)	8.206	0.001
Diabetes (w/complications)	8.248	−0.001
Hypertension	8.260	−0.004
Renal failure	8.772	0.000
Other neurological disorders	8.870	0.000
Paralysis	9.401	0.000
Depression	9.993	0.001
Psychoses	10.184	0.001
Drug abuse	10.248	0.000
Chronic blood loss anemia	10.301	0.000
Tobacco use	10.327	−0.010
Valvular disease	10.581	0.001
Liver disease	11.714	0.008
Rheumatoid arthritis/Collagen vascular disease	12.531	−0.005
Alcohol abuse	12.893	0.001
Metastatic cancer	14.645	−0.002
Coagulopthy	15.552	0.000
Lymphoma	18.546	0.001
Peptic ulcer disease (excluding bleeding)	20.661	0.000
Aids/HIV	20.661	0.000

Note: Permuted Importance values are multiplied by 10,000. Type of procedure = unilateral, staged bilateral, simultaneous bilateral. ASA = American Society of Anesthesiologist, BMI = Body Mass Index.

split. After identifying a forest with optimal tuning parameters, the objective was to identify the most relevant variables and to use only those in a separate RF, while achieving comparable or better prediction accuracy. In a final step, person-specific probabilities of mortality within 90 days of implantation and device failure at specified time points were predicted by entering the patient's information into the RF grown with the most important inputs. The inclusion of a more limited number of predictors in the final RF is important so that the ensemble may be utilized by physicians and patients to make predictions in a time efficient manner.

All analyses were performed using the randomForestSRC package in R. R code demonstrating analyses on simulated data are provided in Appendix 2.

3.3. Results

3.3.1. Mortality within 90 days of TKA

Two regression RFs were grown on the imputed data to compare the impact of the terminal node size. We compared terminal node sizes of 25 and 746 (1% of sample). Both

Table 2. Minimal depth and permuted importance for all variables predicting time to device failure.

Variable	Minimal depth	Permuted importance device failure
Type of procedure	2.203	0.023
Age	2.220	0.049
ASA score	2.239	0.000
Gender	2.625	0.004
BMI	3.368	0.002
Race	4.119	0.006
Bearing surface	4.599	0.001
Mobile bearing design	4.796	0.002
High-flex design	5.254	0.002
Osteonecrosis/avascular necrosis	5.348	0.000
Minimally invasive procedure	5.351	0.002
Osteoarthritis	5.662	0.000
Hinged design	5.695	0.000
Posterior stabilized	5.921	0.002
Rheumatoid arthritis	6.179	0.000
Patella resurfacing	6.307	0.002
Cement fixation	6.492	0.001
Sub-vastus	6.747	0.000
Monoblock all poly tibia	6.767	0.003
Trivector	6.865	0.000
Other surgical techniques	7.383	0.000
Gender-specific implant	7.399	0.002
Parapatellar	7.647	0.002
Constrained design	7.752	0.000
Mid-vastus	7.933	0.001
Inflamatory arthritis	7.961	0.000
Computer-assisted surgery	8.145	0.000
Post-traumatic arthritis	8.457	0.000
Quadricep release	9.334	0.000
Tubercle osteotomy	21.711	0.000

Note: Type of procedure = unilateral, staged bilateral, simultaneous bilateral. ASA = American Society of Anesthesiologist, BMI = Body Mass Index.

had identical Brier scores of .0026, but the larger terminal node size had a better AUC, .75 vs. .73. The larger terminal node size was chosen given modest improvements in performance and computational efficiency. We also considered varying the candidate number of predictors at each split above and below the default of $k/3$. For $k/2$ and $k/4$, the performance metrics were virtually identical, therefore, the default of $k/3$ was retained.

In order to make risk calculation practical, it was necessary to reduce the number of inputs a clinician would be required to enter, therefore, the next step involved variable reduction. A listing of variables sorted by minimal depth, along with their corresponding permuted importance is presented in Table 1. Smaller values for minimal depth indicate greater predictive influence of variables with respect to mortality, while larger values of permuted importance are indicative of variables with greater predictive value. One method of choosing the number of variables to select is based on the minimal depth distribution [22]. Using the minimal depth threshold would have resulted in retaining a total of 20 variables. However, in this context, 10 variables represent the maximum amount of information a clinician is typically willing to enter for any one patient to obtain a prediction, therefore selection was based on this practical consideration. Re-growing a forest based on the 10 variables with the lowest minimal depth values resulted in an equivalent Brier score and slightly improved AUC (.76) in comparison to a forest with all of the original variables. Using this reduced RF, a 75 year-old female undergoing unilateral knee replacement,

30 BMI, American Society of Anesthesiologist score of 2, and having none of the other attributes used to construct the forest, her probability of mortality within 90 days of a TKA is predicted to be .003.

3.3.2. Device failures for aseptic reasons following TKA

Two competing risk survival forests were grown on the imputed data to compare the effect of the terminal node size. We compared terminal node sizes of 25 and 1108 (1% of sample) on the prediction error for device failure based on 100*(1-concordance index), resulting in 38.4% and 34.4%, respectively. These results suggest improvements when going from small to larger terminal nodes and thus the larger node sizes were used for all subsequent forests. We also considered varying the candidate number of predictors at each split above and below the default of $k^{1/2}$. For $k^{2/3}$and $k^{1/3}$, the performance metrics were virtually identical, therefore, the default of $k^{1/2}$ was retained.

The next step involved variable reduction in order to improve the practical implementation of risk prediction. A listing of the variables sorted by minimal depth, along with permuted importance are presented in Table 2. Smaller values for minimal depth indicate greater predictive influence of variables with respect to device failure, while larger values of permuted importance are indicative of variables with greater predictive value. Using a threshold based on a minimal depth distribution would have resulted in retaining all variables. As before, 10 variables represents the maximum amount of information a clinician would typically be willing to enter for any one patient, therefore, variable selection was based on this practical consideration. Based on the results in Table 2, the 10 variables with the lowest minimal depth were selected. Growing a forest based on this subset of variables resulted in a prediction error of 36.0%, slightly higher than the prediction error based on a forest with all of the original variables. For a 65-year-old white male undergoing unilateral knee replacement (metal on conventional polyethylene bearing), with a diagnosis of osteoarthritis, 32 BMI, American Society of Anesthesiologist score of 3, and none of the other attributes used to construct the reduced RF, his probability of device failure is approximately .01 at year one and .03 at year five (Figure 2).

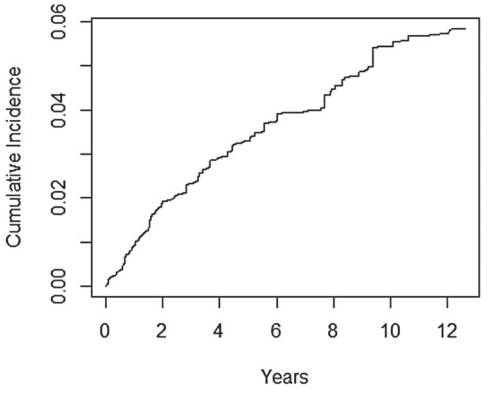

Figure 2. Probability of device failure over time for a hypothetical patient.

4. Discussion

Although risk prediction for adverse health events has been developed and widely applied, they have been based almost exclusively on parametric or semi-parametric statistical methods, which rely on restrictive model assumptions. In particular, nonlinear variable effects and interactions are often either ignored or included in a restricted fashion in existing risk prediction approaches. In this article we proposed the use of random forest, a top-notch machine learning method in terms of prediction accuracy [7,8] that is also completely nonparametric, to address these issues. In addition to describing random forest for risk prediction, the methodology was illustrated on outcomes following total knee replacement. We focused on a select number of issues in order to demonstrate the core features of the proposed method. Certainly, this could be expanded, such as examining the effect of modifying other tuning parameters on prediction accuracy and attempting to further reduce the final number of predictors selected for prediction.

The proposed approach has several strengths as well as a few potential weaknesses. In addition to the strengths already highlighted, RF becomes increasingly appealing relative to parametric/semi-parametric alternatives as the number of predictors and rarity of events increase, given that parametric approaches in this situation can produce solutions that are non-convergent. For instance, applying a logistic regression model to the mortality outcome data would have resulted in non-convergence because two binary predictors and one level of a five-level predictor variable had too few observations/events. A strength of RF would be that it avoids such problems. With respect to limitations, RF requires examination of the impact of tuning parameters, which parametric/semi-parametric methods do not. Furthermore, unlike parametric/semi-parametric approaches, it is more difficult to achieve a substantive understanding of variable effects in RF, although some works have been done to construct estimators comparable to those obtained in parametric models [6]. RF is also more computationally expensive than parametric/semi-parametric approaches. Lastly, we note that RF represents just one of many possible approaches and will not always produce superior prediction accuracy. For instance, after removing the two binary variables and collapsing categories of the five-level predictor that were responsible for non-convergence, we applied a logistic regression to the mortality outcome data, which resulted in almost identical estimates of prediction accuracy to those obtained from RF.

Future work will include making an interactive web page available that can be used by physicians and patients to evaluate risks for adverse events. In conclusion, RF is a useful tool for evaluating the patient risk that can improve the accuracy of predictions and in turn informed decision making.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

[1] K.F. Adams, A. Schatzkin, T.B. Harris,V. Kipnis, T. Mouw, R. Ballard-Barbash, A. Hollenbeck, and M.F. Leitzmann, *Overweight, obesity, and mortality in a large prospective cohort of persons 50 to 71 years old*, New Engl. J. Med. 355 (2006), pp. 763–778.

[2] A.D. Beswick, P. Brindle, T. Fahey, and S. Ebrahim, *A Systematic Review of Risk Scoring Methods and Clinical Decision Aids Used in the Primary Prevention of Coronary Heart*

Disease (Supplement) [Internet], Royal College of General Practitioners (UK), London, 2008 May. (NICE Clinical Guidelines, No. 67S.) Available at http://www.ncbi.nlm.nih.gov/books/NBK55818/ Accessed September 21, 2012.
[3] L. Breiman, *Bagging predictors*, Mach. Learn. 24 (1996), pp. 123–140.
[4] L. Breiman, *Random forest*, Mach. Learn. 45 (2001), pp. 5–32.
[5] L. Breiman, J.H. Friedman, R.A. Olshen, and C.I. Stone, *Classification and Regression Trees*, Wadsworth, Belmont, CA, 1984.
[6] G. Cafri and B.A. Bailey, *Understanding variable effects from black box prediction: Quantifying effects in tree ensembles using partial dependence*, J. Data Sci. 14 (2016), pp. 67–96.
[7] R. Caruna, and A. Niculescu-Mizil, *An empirical comparison of supervised learning algorithms*, Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, 2006.
[8] R. Caruna and A. Niculescu-Mizil, *An empirical evaluation of supervised learning in high dimensions*, Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 2008.
[9] A. Dasgupta, S. Szymczak, J.H. Moore, J.E. Bailey-Wilson, and J.D. Malley, *Risk estimation using probability machines*, BioData Mining. 7 (2014), pp. 2. doi:10.1186/1756-0381-7-2.
[10] A. Elixhauser, C. Steiner, D.R. Harris, and R.M. Coffey, *Comorbidity measures for use with administrative data*, Med. Care. 36 (1998), pp. 8–27.
[11] K.M. Flegal, B.I. Graubard, D.F. Williamson, and M.H. Gail, *Excess deaths associated with underweight, overweight, and obesity*, JAMA. 293 (2005), pp. 1861–1867.
[12] J.C. Foster, J.M. Taylor, and S.J. Ruberg, *Subgroup identification from randomized clinical trial data*, Stat. Med. 30 (2011), pp. 2867–2880.
[13] M.H. Gail, L.A. Brinton, D.P. Byar, D.K. Corle, S.B. Green, C. Schairer, and J.J. Mulvihill, *Projecting individualized probabilities of developing breast cancer for white females who are being examined annually*, J. Natl Cancer Inst. 81 (1989), pp. 1879–1886.
[14] M.H. Gail and J.P. Costantino, *Validating and improving models for projecting the absolute risk of breast cancer*, J. Natl Cancer Inst. 93 (2001), pp. 334–335.
[15] P. Geurts, D. Ernst, and L. Wehenkel, *Extremely randomized trees*, Mach. Learn. 63 (2006), pp. 3–42.
[16] R.J. Gray, *A class of K-sample tests for comparing the cumulative incidence of a competing risk*, Ann. Stat. 16 (1988), pp. 1141–1154.
[17] F. Harrell, R. Califf, D. Pryor, K.L. Lee, and R.A. Rosati, *Evaluating the yield of medical tests*, J. Am. Med. Assoc. 247 (1982), pp. 2543–2546.
[18] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*, Springer-Verlag, New York, 2009.
[19] H. Ishwaran, *The effect of splitting on random forests*, Mach. Learn. 99 (2015), pp. 75–118.
[20] H. Ishwaran, T.A. Gerds, U.B. Kogalur, R.D. Moore, S.J. Gange, and B.M. Lau, *Random survival forests for competing risks*, Biostatistics 15 (2014), pp. 757–773.
[21] H. Ishwaran, U.B. Kogalur, E. Blackstone, and M.S. Lauer, *Random survival forests*, Ann. Appl. Stat. 2 (2008), pp. 841–860.
[22] H. Ishwaran, U.B. Kogalur, E.Z. Gorodeski, A.J. Minn, and M.S. Lauer, *High-dimensional variable selection for survival data*, J. Am. Stat. Assoc. 105 (2010), pp. 205–217.
[23] W.Y. Loh and Y.S. Shih, *Split selection methods for classification trees*, Stat. Sin. 7 (1997), pp. 815–840.
[24] J. Malley, J. Kruppa, A. Dasgupta, K. Malley, and A. Ziegler, *Probability machines. Consistent probability estimation using nonparametric learning machines*, Methods Inf. Med. 51 (2012), pp. 74–81.
[25] D. Noble, R. Mathur, T. Dent, C. Meads, and T. Greenhalgh, *Risk models and scores for type 2 diabetes: systematic review*, BMJ. 343 (2011), pp. d7163.
[26] E.W. Paxton, M.C.S. Inacio, M. Khatod, E.J. Yue, and R.S. Namba, *Kaiser permanente national total joint replacement registry: Aligning operations with information technology*, Clin. Orthop. Relat. Res. 468 (2010), pp. 2646–2663.

[27] E.W. Paxton, M.C.S. Inacio, and M. Kiley, *The Kaiser permanente implant registries: effect on patient safety, quality improvement, cost effectiveness, and research opportunities*, Perm. J. 16 (2012), pp. 36–44.
[28] S. van Dieren, J.W. Beulens, A.P. Kengne, L.M. Peelen, G.E. Rutten, M. Woodward, Y.T. van der Schouw, and K.G. Moons, *Prediction models for the risk of cardiovascular disease in patients with type 2 diabetes: a systematic review*, Heart 98 (2012), pp. 360–369.
[29] S. Wager and S. Athey, *Estimation and inference of heterogeneous treatment effects using random forests*, arXiv preprint arXiv:1510.04342.
[30] P.W. Wilson, R.B. D'Agostino, D. Levy, A.M. Belanger, H. Silbershatz, and W.B. Kannel, *Prediction of coronary heart disease using risk factor categories*, Circulation 97 (1998), pp. 1837–1847.

Appendices

Appendix 1 (simulation study)

In the data-generation process, let X_{ik} for $k = 1,2 \ldots ,10$ and $i = 1, 2 \ldots , N$ be Bernoulli distributed independent variables with success probability (s). The true probabilities (p_i) were calculated as

$$z_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \beta_3 X_{i3} + \cdots + \beta_{10} X_{i10},$$

where $\beta_1 = \beta_2 = \log(2)$, $\beta_3 = \cdots = \beta_{10} = 0$, $\beta_0 = 0$or $\beta_0 = -5$, $X_{i1} \ldots X_{i10} \sim$ bernoulli(s), with s being .1 or .5.

$$p_i = \frac{1}{(1 + \exp(-z_i))}.$$

In order to generate a binary outcome, $Y_i \sim$ bernoulli(p_i). The estimated probabilities ($\hat{p}_i$) are based on either a classification RF or a regression RF. We generated 1000 observations per simulated data set, and for each of the four simulation conditions 1000 simulated data sets were generated. To evaluate the accuracy of the two approaches, the predicted OOB probabilities ($\hat{p}_i$) were compared to the true probabilities (p_i) via the root mean squared error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - \hat{p}_i)^2}.$$

The table below presents a summary of the simulation results. The results indicate that regression RF performs better than classification RF in all four simulated conditions.

S	β_0	Regression RF RMSE (SD)	Classification RF RMSE (SD)
0.1	0	0.071 (0.008)	0.117 (0.008)
0.5	0	0.106 (0.006)	0.219 (0.006)
0.1	−5	0.013 (0.003)	0.022 (0.007)
0.5	−5	0.026 (0.004)	0.058 (0.009)

SD = standard deviation

Appendix 2 (R code for conducting analysis)

A2. Classification

A simulated data set was generated with a binary endpoint and 500 observations. A mix of 10 binary and continuous were included, but only two variables, X1 and X6, had an effect and ∼10% of values for X1 and X6 were set to be missing completely at random. The code to generate the data set is based on the following:

```
set.seed(549)
N=500
int<-rep(0,N)
X1<-rbinom(N, 1, .5)
X2<-rbinom(N, 1, .5)
X3<-rbinom(N, 1, .5)
X4<-rbinom(N, 1, .5)
X5<-rbinom(N, 1, .5)
X6<-rnorm(N)
X7<-rnorm(N)
X8<-rnorm(N)
X9<-rnorm(N)
X10<-rnorm(N)
logodds=log(2)
dat<-data.frame(cbind(int,X1,X2, X3, X4, X5, X6, X7, X8, X9, X10))
z=int+logodds*dat$X1+logodds*dat$X6
pr = 1/(1+exp(-z))
response = rbinom(N,1,pr)
dat$Y=response
dat$int<-NULL
mcar=runif(N, min=0, max=1); dat$X1 = ifelse(mcar<.1, NA, dat$X1)
mcar2=runif(N, min=0, max=1); dat$X6 = ifelse(mcar2<.1, NA, dat$X6)
facs <- (names(dat) %in% c("X1","X2","X3","X4","X5"))
dat[facs] <- lapply(dat[facs], as.factor)
str(dat)
```

Next, we load the libraries required for growing random forest and calculating AUC

```
library(randomForestSRC)
library(AUC)
```

Given the presence of missing data we first impute. The process of imputation requires specification of tuning parameters as with any random forest. Here, the number of variables considered at each split point (mtry) is set to the default for regression problems $k/3$, the number of splits considered for any individual variable (nsplit) is set to 2, the terminal node size (nodesize) is set to be at minimum 1% of the data, and the number of iterations (nimpute) is set to 5. The choice for the tuning parameters in this phase are not empirical based.

```
set.seed(549)
imp_dat<- impute.rfsrc(Y ~ ., data = dat, nsplit = 2, mtry=ceiling((ncol(dat)-1)/3),
nodesize=ceiling(0.01*nrow(dat)), nimpute=5)
```

Once the data have been imputed we formally consider the impact of tuning parameter selection on prediction accuracy. For instance, below we compare a minimum terminal node size of .1% to 1%

```
set.seed(549)
v1<- rfsrc(Y ~ ., data = imp_dat, nsplit = 2, mtry=ceiling((ncol(dat)-1)/3), nodesize=ceiling
(0.001*nrow(dat)))
v2<- rfsrc(Y ~ ., data = imp_dat, nsplit = 2, mtry=ceiling((ncol(dat)-1)/3), nodesize=ceiling
(0.01*nrow(dat)))
AUC1=auc(roc(v1$predicted.oob,factor(v1$yvar)))
B_score1=1/nrow(dat)*sum((v1$yvar-v1$predicted.oob)^2)
AUC2=auc(roc(v2$predicted.oob,factor(v2$yvar)))
B_score2=1/nrow(dat)*sum((v2$yvar-v2$predicted.oob)^2)
```

In this case, there appears to be little difference between the two terminal node sizes considered. A similar process can be undertaken to investigate the impact of other tuning parameters, mtry and nsplit, In high-dimensional settings, it is often of interest to reduce the number of inputs. This can be done by calculating minimal depth and permuted importance

```
var.select(object = v2, conservative = "high")
vimp(v2)$importance [order(-vimp(v2)$importance)]
```

Both minimal depth and permuted importance indicate X1 and X6 are the most important predictors. A forest can be refit with just these two variables.

```
refit < -imp_dat[c("Y","X1","X6")]
set.seed(549)
v3 < - rfsrc(Y ~ ., data = refit, nsplit = 2, mtry = ceiling((ncol(dat)-1)/3), nodesize = ceiling(0.
01*nrow(dat))) # 1% terminal node size
AUC3 = auc(roc(v3$predicted.oob,factor(v3$yvar)))
B_score3 = 1/nrow(dat)*sum((v3$yvar-v3$predicted.oob)^2)
```

Using this reduced forest a prediction can be calculated using the following:

```
lis < -list(X1 = c(1), X6 = c(0))
pre = expand.grid(lis)
myfacs < - !(names(pre) %in% c("Y","X6"))
pre[myfacs] < - lapply(pre[myfacs], as.factor)
str(pre)
predic1 = predict(v3,pre,importance = "none")
predic1$predicted
```

A3. Time-to-event

A simulated data set was generated with a survival endpoint (with competing risk) and 500 observations. As with the classification data, a mix of 10 binary and continuous were included, but only two variables, X1 and X6, had an effect and ~10% of values for X1 and X6 were set to be missing completely at random. The code to generate the data set is based on the following:

```
set.seed(623)
N = 500
X1 < -rbinom(N, 1, .5)
X2 < -rbinom(N, 1, .5)
X3 < -rbinom(N, 1, .5)
X4 < -rbinom(N, 1, .5)
X5 < -rbinom(N, 1, .5)
X6 < -rnorm(N)
X7 < -rnorm(N)
X8 < -rnorm(N)
X9 < -rnorm(N)
X10 < -rnorm(N)
logHR = log(2)
dat < -data.frame(cbind(X1,X2, X3, X4, X5, X6, X7, X8, X9, X10))
dat$lambda < -.05*exp(logHR*X1+logHR*X6)
dat$t < -(-log(runif(N))/dat$lambda)
dat$event < -floor(runif(N, min = 0, max = 3))#approximately equal censored (0) events (1) deaths
(2)
#dat$event < -rbinom(N,1,.5)
dat$lambda < -NULL
mcar = runif(N, min = 0, max = 1); dat$X1 = ifelse(mcar < .1, NA, dat$X1)
mcar2 = runif(N, min = 0, max = 1); dat$X6 = ifelse(mcar2 < .1, NA, dat$X6)
facs < - (names(dat) %in% c("X1","X2","X3","X4","X5"))
dat[facs] < - lapply(dat[facs], as.factor)
str(dat)
```

Next, we load the libraries required for growing random forest

```
library(randomForestSRC)
```

Given the presence of missing data we first impute. The process of imputation requires specification of tuning parameters as with any random forest. Here, the number of variables considered at each split point (mtry) is set to the default for survival problems $\sqrt{k}$, the number of splits considered for any individual variable (nsplit) is set to 2, the terminal node size (nodesize) is set to be at minimum 1% of the data, and the number of iterations (nimpute) is set to 5. The choice for the tuning parameters in this phase are not empirically based.

```
set.seed(623)
imp_dat <- impute.rfsrc(Surv(t, event) ~ ., data = dat, nsplit = 2, mtry = ceiling(sqrt(ncol(dat)-
2)), nodesize = ceiling(0.01*nrow(dat)), nimpute = 5)
```

Once the data have been imputed we formally consider the impact of tuning parameter selection on prediction accuracy. For instance, below we compare a minimum terminal node size of .1% to 1%

```
set.seed(623)
v1 <- rfsrc(Surv(t, event) ~ ., data = imp_dat, nsplit = 2, mtry = ceiling(sqrt(ncol(dat)-2)),
nodesize = ceiling(0.001*nrow(dat)))
v2 <- rfsrc(Surv(t, event) ~ ., data = imp_dat, nsplit = 2, mtry = ceiling(sqrt(ncol(dat)-2)),
nodesize = ceiling(0.01*nrow(dat)))
v1
v2
```

In this case, the larger terminal node size appears to be slightly better. A similar process can be undertaken to investigate the impact of other tuning parameters, mtry and nsplit. In high-dimensional settings, it is often of interest to reduce the number of inputs. This can be done by calculating minimal depth and permuted importance

```
var.select(object = v2, conservative = "high")
vimp(v2)$importance[,1][order(-vimp(v2)$importance[,1])]
```

In this case minimal depth assigns the smallest depths to X1 and X6, but permuted importance only indicated X6 was important. Nevertheless, a forest can be refit with just these two variables.

```
refit <-imp_dat[c("t", "event","X1","X6")]
set.seed(623)
v3 <- rfsrc(Surv(t, event) ~ ., data = refit, nsplit = 2, mtry = ceiling(sqrt(ncol(dat)-2)),
nodesize = ceiling(0.01*nrow(dat)))
```

Using this reduced forest a prediction can be calculated using the following:

```
lis <-list(X1 = c(1), X6 = c(2))
pre = expand.grid(lis)
myfacs <- !(names(pre) %in% c("Y","X6"))
pre[myfacs] <- lapply(pre[myfacs], as.factor)
str(pre)
Tz <- v3$time.interest # all distinct times
Tz <-data.frame(Tz)
Tz
k <-seq(from = 1, to = 316, by =30) #used to select a specific set of times from the dataset in
order to get predictions
Time <-Tz[k,]
pre2 = predict(v3,pre,importance = "none")
CIF = data.frame(pre2$cif[,k,1])
matplot(Time, CIF, ylab = "Cumulative Incidence", col = 1, type = "l")
```