A comparison of tree-based methods for survival data in precision medicine

Jungmi Park

The Graduate School
Yonsei University
Department of Biostatistics and Computing

A comparison of tree-based methods for survival data in precision medicine

A master's thesis
submitted to the Department of Biostatistics and Computing
and the Graduate School of Yonsei University
in partial fulfillment of the
requirements for the degree of
Master of Science

Jungmi Park

December 2021

This certifies that the master's thesis of ***Jung mi Park*** is approved.

Inkyung Jung: Thesis Supervisor

Chung Mo Nam: Thesis Committee Member #1

Hyung Seok Park: Thesis Committee Member #2

The Graduate School

Yonsei University

December 2021

Contents

List of Tables

List of Figures

Abstract

A comparison of tree-based methods

for survival data in precision medicine

Precision medicine has been addressed in several clinical trials. Studies done to identify effective cancer treatments are mostly conducted on groups and do not account for individuals' characteristics. However, it is essential to determine how subgroups are differentially affected by interventions. There are multiple tree-based methods for analyzing data with right-censored survival outcomes related to precision medicine. This study compared four tree-based methods for identifying subgroups: the latest method, the depth importance in precision medicine; double-weighted trees; model-based partitioning; and simple Cox split trees methods.

Simulations were performed to compare the performance of these four methods in various scenarios where data were generated by different survival time distributions. The accuracy of each method was measured as the proportion of correctly selecting the most

important predictor at the first node among the total number of simulation runs. As the first two well-performed methods, MOB and DIPM methods were fit using the Surveillance, Epidemiology, and End Results database. This study's results can be used to help researchers and clinicians to choose the optimal tree-based method for analyzing right-censored data in precision medicine.

Keywords: precision medicine, variable importance, subgroup identification, random forest, survival data

1. Introduction

Precision medicine, also referred to as personalized medicine, has grown in popularity as a cancer treatment. Its treatment outcomes can be reduced by identifying patient subgroups according to their genes, other characteristics, and environments. There has been a deluge of machine learning algorithms that enable the development of precision medicines based on clinical data. These machine learning algorithms identify possible target-based medicines for treating various diseases, such as breast cancer, COVID-19, and coronary artery disease. Tree-based methods are some of the most commonly used machine learning tools and they produce simple and easily interpretable decision rules. However, they are prone to overfitting, when the model is particularly deep and too specific to the dataset on which they are trained. This study compared four tree-based methods that used ensemble, bagging, and random forest models to determine which performed the best in various situations.

In this study, the performance of the following four tree-based methods was tested using right-censoring data: simple Cox splits (Su and Tsai, 2005), the model-based recursive partitioning (MOB) method (Seibold et al., 2016), the double-weighted tree method (Zhu et al., 2017), and the depth importance in precision medicine (DIPM) method (Chen and Zhang, 2020).

All four of these methods are based on recursive partitioning, but they use different techniques to find the underlying predictors in the data. For example, when fitting

ensemble tree model, the weighted method randomly generates candidate splitting variables and cut-points while the other three methods consider a broader pool of candidate splits. Therefore, this study was conducted to determine how precisely the methods detected the significant predictors of differences of mean survival estimates between two treatments.

The rest of the paper is organized as follows. Section 2 describes the methods analyzed in this study. Section 3 discusses the results of simulations assessing the methods' performances. Section 4 tests the methods on real data. Finally, Section 5 discusses this study's results and provides conclusions.

2. Tree-based methods

Tree-based methods are also known as recursive partitioning methods and are used for segmentation, classification, and prediction. This section explains how tree-structured analyses can be applied to right-censored survival data and identify homogenous subgroups.

2-1. Depth importance precision medicine

The DIPM algorithm is composed of the importance score method developed by Chen et al. (2007) and the tree-based method developed by Zhu et al. (2017). The DIPM method outperforms other methods because it uses the importance score method in all candidate variables. The DIPM method utilizes depth variable importance scores to select the best candidate variable to split each node. The score integrates the depth of the node within the tree and the magnitude of the relevant effect. The closer the node in question is to the root node, the more important the variables are. A Cox model is used to generate the z^2 statistic of the effect of the split and treatment interaction term. The magnitude of the effect of splitting nodes is also accounted for by setting it equal to this z^2 statistic.

First, bootstrap sampling is performed to generate *M* trees with current within-node

data. The best split of a given node in the bootstrap samples is that with the largest z^2 Wald test statistic. The best splits at all nodes are continuously identified until there are fewer than the minimum number of child nodes for every candidate split, which is defined as the minimum number of subjects in nodes. This procedure is repeated until a predefined number of embedded trees are fit and a random forest is constructed using then-current within-node data. A random forest of embedded trees is set to split a node with the best variable. Once it is built, a variable importance score is calculated for each variable p for each tree *N*:

$$Score(N, i) = \sum_{p \in N_i} 2^{-D(p)} M_p$$

where N_i is the set of nodes in tree *N* split by variable *i* and $D(p)$ is the depth of node *p*. Cox hazard model at node *p* as follows:

$$h(p, treatment, split) = h_0(p) * e^{(\beta_1 * treatment + \beta_2 * split + \boldsymbol{\beta_3} * treatment * split)}$$

where $split = \begin{cases} I(x \leq c), & if\ x\ is\ ordinal \\ I(x \in S), & else\ if\ x\ is\ nominal \\ I(x = 0), & else\ if\ x\ is\ binary \end{cases}$

M_p is the z^2 statistics from the Wald test of significance of β_3. By using a discount rate of 0.5 for depth values, deeper variables have smaller scores and so are considered less important. For instance, the first split variable has a value of 0.5 and its daughter nodes have values of 0.25. The magnitude of the effect of splitting a node is the significance of the interaction term in a Cox model. The best split variable is identified with the largest scores averaged across all *M* trees in forest *f* as follows:

$$score(f,i) = \frac{1}{M}\sum_{N \in f} score(N,i)$$

The split that is chosen is the one with the largest z^2 Wald test statistic. These steps are repeated in subsequent nodes until the minimum number of subjects in the nodes of the embedded trees is satisfied at which point growth stops. The best-predicted treatment class was confirmed by comparing the mean survival times of each treatment group, which were defined as the areas under their Kaplan-Meier curves.

2-2. Double-weighted trees

The double-weighted tree algorithm is designed to infer personalized treatment rules with high-dimensional covariates. Given that this algorithm addresses dimensionality by conducting marginal searches as a result of splitting rules, it incorporates the strengths of the outcome-weighted learning framework and the interpretability of the single-tree method. It also constructs embedded trees like the DIPM algorithm and uses an accelerated failure time model to analyze right-censored data.

Double-weighted trees are built in two main steps. The first step is running each pseudo-algorithm of the general subject-weighted ensemble tree model. The second step is running the single-learning tree model for developing optimal treatment rules. The survival time is expressed as:

$$Y = \min(Y^0, C)$$

where Y^0 is the true survival time and C is the censoring time. Random sample sets are expressed as:

$$\{Y_i, \delta_i, X_i\}_{i=1}^{n}$$

where δ is the censoring indicator and X is patients' characteristics. Then, the first Kaplan-Meier weight, u_i, can be generated for each subject i. The Kaplan-Meier weight for the first subject is expressed as:

$$u_1 = \frac{\delta_{(1)}}{n}, u_i = \frac{\delta_{(i)}}{n-i+1}\prod_{j=1}^{i-1}\left(\frac{n-j}{n-j+1}\right)^{\delta_{(i)}}, i = 2, 3, .., n,$$

where $\delta_{(1),}\delta_{(2)}, \dots, \delta_{(n)}$ have the ordered statistics $Y_{(1),}Y_{(2),}, \dots, Y_{(n).}$

Then bootstrap samples are drawn from training data $\{\boldsymbol{X_i}, w_i, Y_i\}_{i=1}^{n}$ for regression and $\{\boldsymbol{X_i}, w_i, A_i\}_{i=1}^{n}$ for classification. Random candidate splitting variables and cut points are generated at each node until the sample size is under the predefined minimum number of samples. Each variable's potential contribution score is calculated by weighting X's variance and Gini impurity (Zhu et al., 2017). An ensemble tree model is built based on the candidate split with the highest score. Then, the mean response function $\hat{m}$ can be estimated by minimizing the loss of the weighted least squares regression of X, which is expressed as:

$$\frac{1}{2}\sum_{i=1}^{n} u_i^*(\log(Y_i) - m(X_i))^2 \ , \ u_i{}^* = u_k \ if \ k = \sum_j I(Y_{(j)} \leq Y_i)$$

Next, using the estimated mean response function, the plug-in subject weight is calculated as follows:

$$w(X_i) = \frac{|Y_i - \hat{m}(X_i)|}{P(A_i|X_i)} \ , \ where \ m(X) = E(Y|X)$$

Last, the estimated subject weights are used to obtain the optimal treatment rule by minimizing the weighted misclassification error as follows:

$$\hat{D}^*(X) = \arg\min \frac{1}{n} \sum_{i=1}^{n} u_i{}^* w(X_i) I(A_i \neq D(X_i))$$

The double-weight tree method's importance score can be set equal to the misclassified treatment classification ratio. Single reinforcement learning trees are traditional in that they select split variables and mute variable noise at each internal node.

2-3. Model-based recursive partitioning

Model-based recursive partitioning is used to detect patient subgroups that are identifiable by predictive factors. A Weibull or Cox model is used to analyze right-censored survival times. However, unlike the double weighted and DIPM methods, the MOB produces a segmented model with treatment parameters that differ by patient subgroup. The outcome is $Y = \min(Y^0, C)$ where Y^0 is the true survival time and C is the censoring time. Observed samples are expressed as $\{Y_i, \delta_i, X_i\}_{i=1}^{n}$ where δ is the censoring indicator and X is the set of patients' features. The model $M(y, \delta, x)$ describes the conditional distribution of survival probability as a function of the intervention and patient characteristics. The parameters are noted as the intercept α and treatment effect β. The estimators are solved by minimizing the negative-log-likelihood of X, which is expressed as follows:

$$\partial(\alpha, \beta) = \arg min \sum_{i=1}^{N} \Psi(y, \delta, x)$$

A treatment effect is estimated when a parameter corresponding to that treatment is applicable to all patients. However, the effect may not be consistent across subgroups. Thus, a generalized M-fluctuation test should be conducted to identify parameter

instability and so determine whether splitting is necessary. The splitting node Z_j has the highest parameter instability, which is equivalent to the minimum p-value. When the variable is statistically significant, patients are split into $b = 1, \ldots, B$ subgroups and the subgroup-specific model parameters are $\partial(b)$, which explains both $\alpha(b)$ and $\beta(b)$. Thus, segmented objective functions are generated using the following method:

$$\sum_{b=1}^{B} \sum_{i \in I_b} \Psi(y_i, \delta_i, x_i)$$

Partitioned model parameters are estimated by minimizing the segmented objective function, which is:

$$(\hat{\vartheta}(b))_{b=1,..,B} = \arg min \sum_{i=1}^{N} \sum_{b=1}^{B} 1(z_i \in B_b) \Psi((y, x), \vartheta(b))$$

The best split points at each node are calculated by locally optimizing Ψ. Once there is no instability in the parameters, recursive partitioning stops and the final tree is built. The Weibull model achieves better fits because Martingale residuals are defined as the

derivative of the log-likelihood with respect to the intercept whereas the Cox model treats the intercept as a nuisance parameter that is omitted in partial likelihood analysis. The Martingale residuals are used to check whether there is a general difference in the endpoints of different patients. Variables and cut-points are not normally distributed, so the Bonferroni-adjusted permutation tests are used to partition variables and select cut-points.

2-4. Simple Cox splits

The simple Cox splits method combines Cox proportional hazards (PH) regression model with a tree-based model. The Cox PH model is defined as:

$$\lambda(t|z) = \lambda_0(t)\exp\{\beta' z^0\}$$

However, the Cox PH model cannot handle nonlinear effects or complex interactions. The simple Cox splits model is produced by adding an augmentation tree structure to the Cox PH model to check the accuracy of the predictor survival probability it identifies. The hybrid simple Cox splits model is expressed as:

$$\lambda(t|z) = \lambda_0(t)\exp\left\{\beta'^{Z^0} + \gamma'^{Z^{(T)}}\right\}$$

where $Z^{(T)}$ is a dummy vector by augmentation *T*. If the ith observation belongs to the jth terminal node of T, ${Z_{ij}}^{(T)}$ =1 and 0 otherwise. β'^{Z^0} is the nonlinearity, thresholds, and interactions in augmentation tree *T*. The adequacy of this tree is assessed by testing the hypothesis $H_0{:}\gamma = 0$. Other techniques, such as growing larger trees, pruning, and selecting tree sizes following the classification and regression tree (Breiman et al., 1984). Tree growth is expressed as follows:

$$\lambda(t|z) = \lambda_0(t)\exp\left\{\beta'^{Z^0} + v1_{\{Z_j \le c \text{ or } Z_j \in S\}}\right\}$$

where term $1_{\{Z_j \le c\}}$ equals 1 when Z_j is a continuous variable that is split by cut-point c or it is a categorical variable included within *S*. By using the covariate-adjusted log-rank test, this method solves the partial log-likelihood of the hybrid model and finds the best split. Then Bayesian information criterion is utilized to select the best tree size.

3. Simulations

Simulations were conducted using 9 different scenarios to determine the methods' accuracies over different time distributions, sample sizes, and numbers of variables. Each scenario was simulated 500 times. The accuracy was estimated by the percentage of 500 simulations in which the important variables were correctly selected as the first split or first three splits.

3-1. Scenarios

Scenarios 1–5 had either 15 or 30 candidate split variables and 250 or 1,000 pieces of data. Datasets with 15 candidate split variables consisted of 9 ordinal, normally distributed variables and 6 binary variables. The continuous data were sampled from the normal distribution N(0,1) and the binary data were sampled from the Discrete Uniform[0,1]. Datasets with 30 candidate split variables were generated using the same distributions and had 18 ordinal, normally distributed variables and 12 binary variables.

Scenario 1: The first scenario was an exponential survival time model containing the treatment and one important continuous variable.

$Y_0 \sim Exp(e^{\mu})$

$C \sim Exp\left(e^{0.3(U_1+U_2)}\right)$

$Y = \min(Y_0, C)$

$Censoring\ rate = 50\%$

$\mu = -0.6I_B - 0.7X_1 + 0.5I_BX_1 - 0.7X_5X_7 + 0.2X_4 + 0.5X_3$

Scenario 2: The second scenario was a Weibull survival time model containing the treatment and one important continuous variable.

$Y_0 \sim Weibull(e^{\mu}, 2)$

$C \sim Exp\left(e^{-0.3(U_1+U_2)}\right)$

$Y = \min(Y_0, C)$

$Censoring\ rate = 47\%$

$\mu = -0.6I_B - 0.7X_1 + 0.5I_BX_1 - 0.7X_5X_7 + 0.2X_4 + 0.5X_3$

Scenario 3: The third scenario was an underlying tree model containing the treatment and one important binary variable.

$Y_0 \sim Weibull(e^{\mu}, 2)$

$C \sim Exp(0.8e^{-\mu})$

$Y = \min(Y_0, C)$

$Censoring\ rate\ = 45\%$

$\mu = I_{(X_1 \leq 0)}(5.0I_A + 3.8I_B) + I_{(X_1 > 0)}(3.6I_A + 4.0I_B)\ -0.7X_5X_7 + 0.2X_4 + 0.5X_3$

Scenario 4: The fourth scenario was an underlying tree model containing the treatment and three important binary variables.

$Y_0 \sim Weibull(e^{\mu}, 2)$

$C \sim Exp(0.8e^{-\mu})$

$Y = \min(Y_0, C)$

$Censoring\ rate\ = 46\%$

$$\mu = I_{(X_1 \leq 0 \cap X_2 \leq 0)}(5.0I_A + 1.6I_B) + I_{(X_1 \leq 0 \cap X_2 > 0)}(3.6I_A + 4.0I_B)$$
$$+I_{(X_1 > 0 \cap X_3 \leq 0)}(4.0I_A + 3.6I_B) + I_{(X_1 > 0 \cap X_3 > 0)}(1.6I_A + 5.0I_B)$$

Scenario 5: The fifth scenario was an underlying Weibull survival time model that violated the proportional hazards assumption, differed by treatment group, and contained one important continuous variable.

$$Y_0 \sim Weibull(1, e^{\mu})$$

$$C \sim Exp\left(e^{-0.3(U_1+U_2)}\right)$$

$$Y = \min(Y_0, C)$$

$$Censoring\ rate\ = 56\%$$

$$\begin{cases} \mu = 3.6 - 0.6X_1^2 + 0.05X_5, & when\ the\ treatment\ equals\ 0 \\ \mu = 0.5 + 0.1X_1, & when\ the\ treatment\ equals\ 1 \end{cases}$$

Scenarios 6–9 were conducted with three sets of variables and sample sizes of 250. Each set of variables included 10 variables that were all ordinal and normally distributed, $X \sim N(0, \Sigma)$, and $\Sigma_{i,j} = \rho^{|i-j|}$ where $\rho = 0.2$. Then either 1, 5, or 20 variables that were correlated with the important variable X_1 were added.

In scenarios 6–8, $Z = 0.8X_1 + 0.1N_1 + 0.1N_2 + N_3$ where N_1 and N_2 are both N(0,1) and N_3 is N(0,0.2). In scenario 9, $Z = 0.8X_1 + 0.1X_2 + 0.1X_3 + N_4$ where N_4 is N(0,0.4).

Scenario 6: The sixth scenario was an exponential survival model containing the treatment and one important continuous variable.

$Y_0 \sim Exp(e^{\mu})$

$C \sim Exp\left(e^{0.3(U_1+U_2)}\right)$

$Y = \min(Y_0, C)$

$Censoring\ rate = 49\%$

$\mu = -0.2I_B - 1.1X_1 + 1.2I_BX_1$

Scenario 7: The seventh scenario was a Weibull survival time model with the treatment and one important continuous variable.

$Y_0 \sim Weibull(e^{\mu}, 2)$

$C \sim Exp\left(e^{-0.3(U_1+U_2)}\right)$

$Y = \min(Y_0, C)$

$Censoring\ rate = 51\%$

$\mu = -0.2I_B - 1.1X_1 + 1.2I_BX_1$

Scenario 8: The eighth scenario was a Weibull survival time model with the treatment and one important binary variable.

$Y_0 \sim Weibull(e^{\mu}, 2)$

$C \sim Exp(0.8e^{-\mu})$

$Y = \min(Y_0, C)$

$Censoring\ rate = 47\%$

$\mu = I_{(X_1 \leq 0)}(5.0I_A + 1.6I_B) + I_{(X_1 > 0)}(3.6I_A + 4.0I_B)$

Scenario 9: The ninth scenario was a Weibull survival time model with the treatment and three important binary variables.

$Y_0 \sim Weibull(e^{\mu}, 2)$

$C \sim Exp(0.8e^{-\mu})$

$Y = \min(Y_0, C)$

$Censoring\ rate = 51\%$

$$\mu = I_{(X_1 \leq 0 \cap X_2 \leq 0)}(5.0I_A + 1.6I_B) + I_{(X_1 \leq 0 \cap X_2 > 0)}(3.6I_A + 4.0I_B)$$
$$+ I_{(X_1 > 0 \cap X_3 \leq 0)}(4.0I_A + 3.6I_B) + I_{(X_1 > 0 \cap X_3 > 0)}(1.6I_A + 5.0I_B)$$

3-2. Results

The simulation results are presented in Tables 1 and 2. In scenarios 1–5 in which the amount of data increased, all methods' performance was negatively correlated with the number of candidate split variables and positively correlated with sample size. The reason for the positive correlation was that data provides information, so the more data there is, the more information there was for the methods to identify patterns. However, the MOB method tended to perform best in scenarios 1, 2, and 5. followed in order of decreasing performance by the DIPM, simple Cox, and double weighted tree methods. However, the MOB method performed second-best after the DIPM method in scenarios 3 and 4 and when there were 30 variables. In scenario 5, all methods were confirmed to perform the worst in which survival time models differed by treatment group.

Although the weighted classification tree method was designed to perform well with high-dimensional covariates, it only worked as well as the other methods in scenarios 3 and 4 in which the tree depth was greater than 1. It performed worst in scenarios 1, 2, and 5 in which there were non-tree survival time distributions. Also, the simple Cox tree tended to outperform the weighted method, likely because each candidate split variable contributed only one random split to the pool of candidate splits at each node.

For scenarios 6–9, all methods' performances were negatively correlated with the number of correlated variables. As in scenarios 1–5, the DIPM and MOB methods outperformed the other two methods. The MOB method slightly outperformed the DIPM for non-tree time distributions while the DIPM method outperformed the MOB method

for the tree time distributions in scenarios 8 and 9. Tree model depth was positively correlated with the differences in the methods' performances. The weighted method considered a narrower pool of candidate splits than the other three methods, which caused it to perform worse. For this same reason, the weighted method was affected by additional correlation. Collectively, these results show that the sample size and the number of variables affect tree-based method performance at predicting survival outcomes.

Table 1. Percentage of simulations in which X_1 was correctly selected as the first split in scenarios 1–3 and 5 and X_1, X_2, and X_3 were correctly selected as the first three splits in scenario 4.

		N = 250		N = 1,000	
Scenario number and description	**Method**	**P = 15**	**P = 30**	**P = 15**	**P = 30**
1. Non-tree, exponential	**Weighted**	15.6	12.6	17.4	10.0
	Simple	51.2	29.0	63.2	31.8
	MOB	61.4	47.4	72.0	52.4
	DIPM	56.4	30.0	63.6	32.2
2. Non-tree, Weibull	**Weighted**	17.8	13.4	19.6	11.2
	Simple	56.8	24.2	68.2	36.4
	MOB	60.0	25.6	78.2	35.2
	DIPM	61.2	46.8	72.0	56.6
3. Tree of depth 2	**Weighted**	68.0	49.2	82.0	58.8
	Simple	72.6	48.8	76.6	49.8
	MOB	73.8	51.2	72.6	56.4
	DIPM	77.4	63.6	89.8	70.0
4. Tree of depth 3	**Weighted**	74.0	68.0	84.6	66.8
	Simple	76.0	77.4	82.2	61.2
	MOB	82.0	74.6	84.0	73.8
	DIPM	83.6	77.4	91.0	87.0
5. Non-tree, non-PH	**Weighted**	10.0	1.8	17.2	3.0
	Simple	11.0	9.8	21.2	13.4
	MOB	22.8	13.4	25.6	19.8
	DIPM	20.4	11.0	21.6	12.6

Table 2. Percentage of simulations in which X_1 was correctly selected as the first split in scenarios 6–8 and X_1, X_2, and X_3 were correctly selected as the first three splits in scenario 9.

Scenario number and description	No. of Z Vars.	Weighted	Simple	MOB	DIPM
6. Non-tree, exponential	**1**	52.0	54.0	60.8	59.8
	5	7.4	18.4	20.6	17.2
	20	1.2	1.6	9.6	7.6
7. Non-tree, Weibull	**1**	53.2	62.0	73.2	77.4
	5	15.2	21.6	25.8	29.0
	20	6.0	5.8	15.4	13.0
8. Tree of depth 2	**1**	79.6	86.2	98.4	99.6
	5	61.8	82.0	90.2	95.2
	20	53.6	63.8	77.8	80.0
9. Tree of depth 3	**1**	70.0	61.0	71.2	82.6
	5	27.8	30.2	46.6	51.6
	20	18.0	12.4	25.4	35.0

4. Applications

The methods were used to analyze Surveillance, Epidemiology, and End Results (SEER) data. This dataset reflects cancer incidence rates in population-based cancer registries covering 27.8% of the US population. Data were generated using SEER*Stat version 8.2.2. The sample contained data about 485,245 breast cancer patients who received nipple-sparing mastectomies (NSM) and 14,770 breast cancer patients who received total mastectomy (TM). The outcome variable was time to death attributable to breast cancer. The candidate covariates were age, sex, race, tumor size, tumor grade, estrogen receptor (ER) status, progesterone receptor (PR) status, Human epidermal growth factor receptor 2 (HER2), and breast cancer staging provided by the American Joint Committee on Cancer (AJCC). The interventions were NSM and TM.

After removing data about patients with missing outcomes, multiple imputation by chained equations was implemented to generate missing values for candidate covariates. The censoring rate of the final dataset was 12%. The data was then divided in half, one half of which was used for training and the other half of which was used for testing. Final trees were built using the DIPM and MOB methods. MOB analysis identified the AJCC stage ($\leq$ 2 vs. $>$ 2) at the first split and recursively identified AJCC stage ($\leq$ 1 vs. >1) and ER (negative vs. positive) at the next two splits (Figure 1). Next, a final tree was generated using the DIPM method and the training dataset that identified tumor grade at the first split node and age and PR at the next two split nodes. The cut point for tumor

grade was < 3 versus 3, ≤ 52 versus > 52 for age, and negative versus positive for PR (Figure 2). Kaplan-Meier curves and log-rank p-values were calculated to identify statistically significant subgroups and optimal treatments for each of them. All splits identified by the MOB and DIPM methods were statistically significant. Although each MOB and DIPM method identified different splitting values and candidate variables, they identified statistically significant treatments for clinically meaningful subgroups. The Kaplan-Meier curves and the log-rank p-values of the testing data were similar to those for the training data. Mean survival probabilities within identified subgroups were different between all treatments and an optimal treatment was identified for each subgroup. The effect of NSM was as good as TM for most subgroups. However, NSM produced better outcomes for patients whose cancer stage was over 2 and ER was negative than TM.

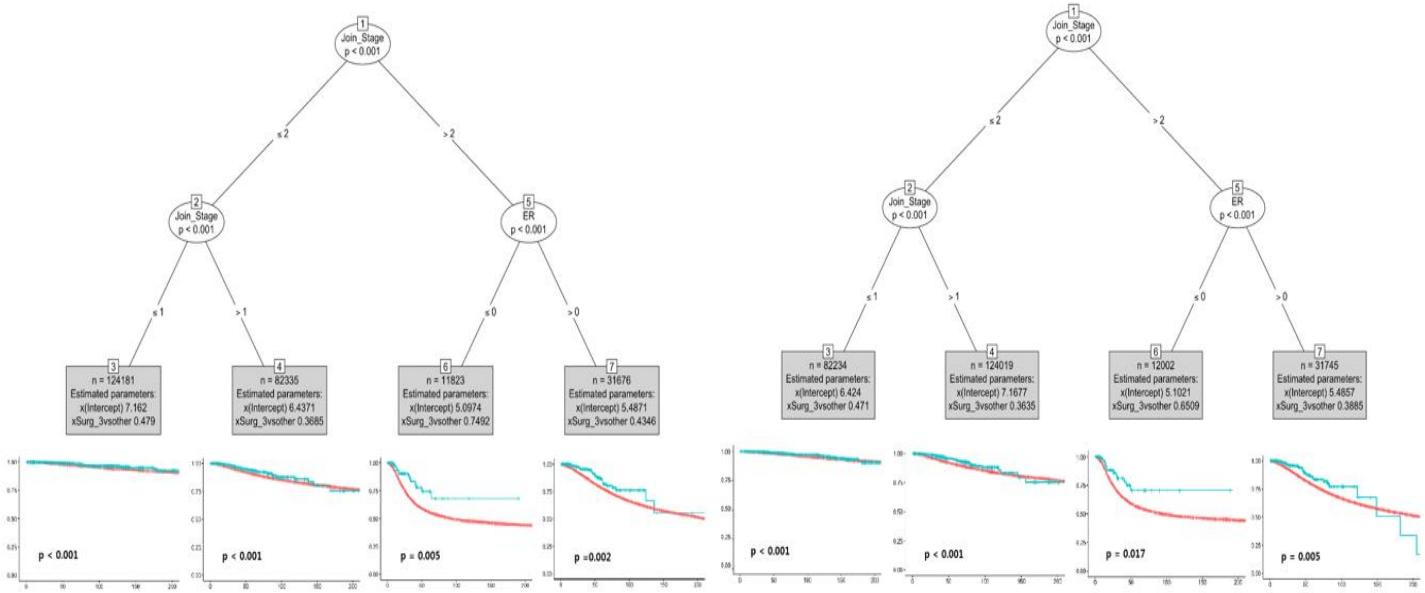

Figure 1. The final tree and corresponding Kaplan-Meier curves for each subgroup identified in SEER data. Trees fit by the MOB method on (left) training data and (right) test data.

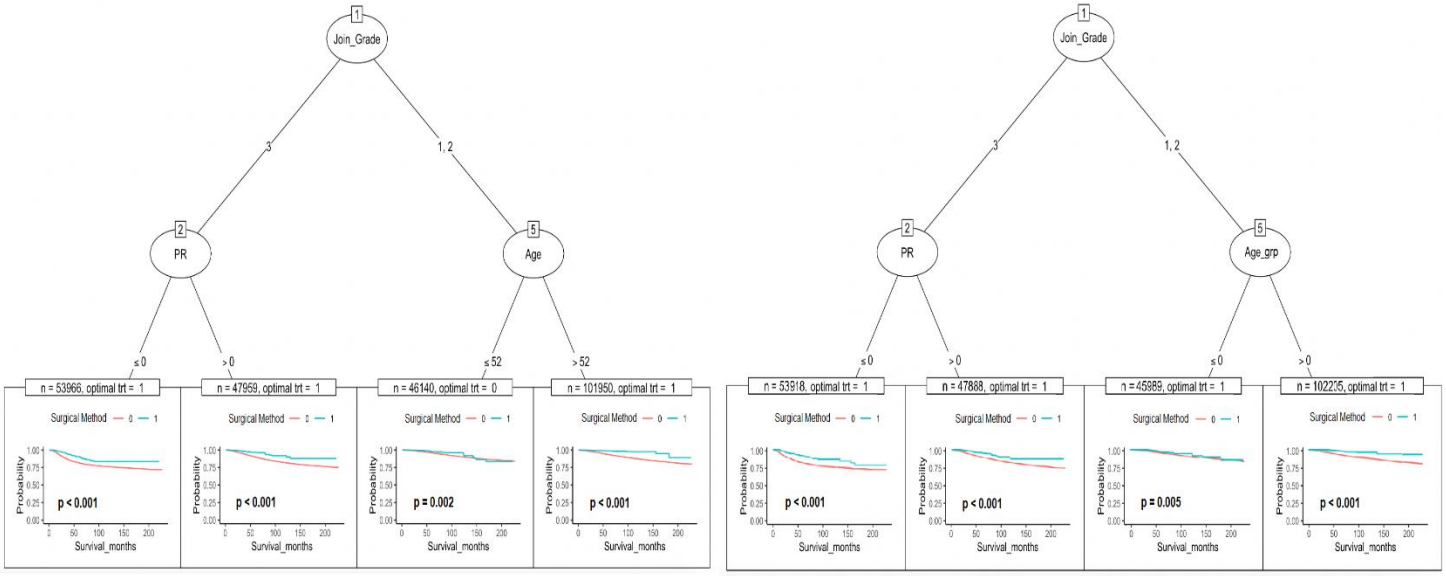

Figure 2. The final tree and corresponding Kaplan-Meier curves for each subgroup identified in SEER data. Trees fit by the DIPM method on (left) training data and (right) test data.

5. Conclusion and discussion

Tree-based models are beneficial to precision medicine due to the fact that they are good at handling complex and non-linear relationships. Among four tree models, a simulation study which included various settings of time distribution showed the MOB and DIPM methods presented good performance overall.

The MOB method for analyzing right-censored survival outcomes was as effective as the DIPM method because both produced significant results in both simulations and when applied to real-world data. In simulations, the MOB method was preferred over the DIPM method under non-tree structured scenarios, whereas the DIPM method had higher accuracies than the MOB method in tree-structured scenarios. In the real-world data analysis, two trees maximized the differences in predicted survival probabilities between treatments within identified subgroups. This result was likely a product of the fact that both trees generate a random forest of embedded trees and select the most appropriate variable for splitting each node. They also resolve the recursive partitioning issue, which is an inherent part of X and the reason that the tree structure varies from sample to sample.

The MOB method took 99% less time to compute than the DIPM method. The MOB method produced scenario results within 5 mins while the DIPM method took at least 10 hrs. Therefore, the MOB method can be used as an alternative to the DIPM method, considering computation time and performance. This study showed that there was a

tradeoff between accuracy and computation time, which was similar to the findings of Chen and Zhang (2020).

There are improvements to be required in future research. One is to validate with any clinical datasets. The final models fit on the 50% training dataset were evaluated by 50% of test dataset, which were held-out samples. Although the results by given models performed best, external validation should be carried out. In addition to this, methods can be applied to data containing multiple treatments and varying censoring rates in future studies.

References

Victoria Chen and Heping Zhang. (2020), Depth importance in precision medicine (DIPM): a tree- and forest-based method for right-censored survival outcomes. Biostatistics, kxaa021

Ruoqing Zhu et al. (2017), Greedy Outcome Weighted Tree Learning of Optimal Personalized Treatment Rules. Biometrics, 73(2): 391–400.

Heidi Seibold et al. (2016), Model-Based Recursive Partitioning for Subgroup Analyses. The International Journal of Biostatistics, 12(1): 45-63

Zeileis, A. et al. (2018), Model-based recursive partitioning. Journal of Computational and Graphical Statistics, 17: 492–514.

Zhang, H. and Singer, B. (2010), Recursive Partitioning and Applications. New York: Springer.

Su. X. et al. (2008), Interaction trees with censored survival data. International Journal of Biostatistics, 4: 1–26.

Su. X. and Tsai CL. (2005), Tree-augmented Cox proportional hazards models. Biostatistics, Volume 6, Issue 3: 486–499

Mi Du et al. (2020), Comparison of the Tree-Based Machine Learning Algorithms to Cox Regression in Predicting the Survival of Oral and Pharyngeal Cancers: Analyses Based on SEER Database. Cancers, 12: 2802

L.L. Doove et al. (2014), A comparison of five recursive partitioning methods to find person subgroups involved in meaningful treatment-subgroup interactions. Advances in Data Analysis and Classification, 8: 403-425

Aniek S. et al. (2017), Comparing four methods for estimating tree-based treatment regimes. The international journal of biostatistics, 13(1): pages

Marc D. Ryser et al. (2019), Incidence of Ductal Carcinoma in Situ in the United States, 2000-2014. Cancer Epidemiol Biomarkers Prev. 28(8): 1316–1323

Negassa, A. et al. (2005), Tree-structured subgroup analysis for censored survival data: validation of computationally inexpensive model selection criteria. Statistics and Computing, 15: 231–239.

Wei-Yin Loh et al. (2019), Subgroup identification for precision medicine: A comparative review of 13 methods. WIREs Data Mining Knowl Discov., 9(5): e1326

Zeileis, A. et al. (2006), Evaluating Model-based Trees in Practice. Research Report Series, 32.

Su X, et al. (2009), Subgroup analysis via recursive partitioning. Journal of Machine Learning Research, 10:141–158

Torsten H., et al. (2015), Partykit: a modular toolkit for recursive portioning in R. Journal of Machine Learning Research, 16(1): 3905-3909

Terry M. et al. (1997), An introduction to recursive partitioning using the rpart routine. Technical Report 61, Section of Biostatistics, Mayo Clinic, Rochester, URL http://www.mayo.edu/hsr/techrpt/61.pdf

국 문 요 약

정밀의료를 위한 생존자료에서의

트리 기반 방법들 간의 비교

최근 각종 질병의 진단과 치료에 있어 환자들 마다 서로 다른 치료방법이 필요성이 나타난다. 이에 따라, 유전체, 환경 및 생활습관, 임상 정보 등을 토대로 질병을 예방, 치료하는 목적을 가진 의료 패러다임이 부각된다. 이는 정밀 의료 시대가 도래 됨으로써 표적 치료로의 변화, 바이오마커를 기반으로 한 맞춤형 치료 전략 등 새로운 패러다임을 가져온다. 이러한 개인 맞춤형 의료 진단 및 치료 서비스 제공을 위한 머신 러닝의 기법들 중 하나로, 생존 자료를 활용한 트리 기반 방법들이 존재한다.

따라서, 본 연구는 치료 방법들 간에 유의적인 생존 확률 차이가 존재하는 하위 그룹을 발굴하기 위해 기존의 Simple Cox split 방법(2005), Model-based recursive partitioning(MOB) 방법(2016), The weighted

classification 방법(2017)과 더불어 최근 개발된 Depth importance in precision medicine (DIPM) 방법(2020)을 비교한다.

연구 결과, DIPM, MOB, Simple Cox, 그리고 The weighted classification 순으로 성능이 좋음을 확인하였다. 특히, MOB 방법은 DIPM 방법과 비슷하거나 특정 non-tree 시나리오에서 더 좋은 정확성을 보여주었다. 이에 대한 결론의 주 요인은 방법들의 알고리즘을 통해 확인해 볼 수 있다. 기존의 방법들은 single tree만을 고려하거나 임의적으로 변수 및 cut-point를 지정하지만, DIPM은 embedded tree를 구현함으로써 각 노드 별로 random forest를 형성하여 candidate variable들을 모두 고려한다. 반면, MOB과 Weighted classification은 bootstrapping 기법을 통해 embedded tree들로 구성된 random forest를 구축한다. 따라서, 시뮬레이션을 통해 DIPM 방법과 MOB 방법에서 시간과 성능 관점의 trade-off가 존재함을 확인하였다. 이를 통해 비슷한 성능을 보이면서 계산 시간이 매우 빠른 MOB 방법을 DIPM 방법의 대안책으로 사용할 수 있을 것을 제안한다.

핵심되는 말: 정밀의료, 변수 중요성, 하위 그룹 확인, 랜덤 포레스트, 생존 자료