

Supplementary Material 1: Selecting P and d_0 for the lung cancer data

Initially, we did not know which values of P and d_0 are suitable for the data analysis. Thus, we generated a matrix of the c -index all combinations from P= 0.001 to 0.1 and from $d_0 = 0.01$ to 0.1 (below). We then decided to use only P= 0.01 since other values produce the same c -index. We also use only $d_0 = 0.01$ and $d_0 = 0.1$ that cover all the possible values of the c -index.

> cindex	P-value=0.001	P-value=0.002	P-value=0.003	P-value=0.004	P-value=0.005
d0=0.01	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.02	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.03	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.04	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.05	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.06	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.07	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.08	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.09	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.1	0.5210	0.5210	0.5210	0.5210	0.5210
	P-value=0.006	P-value=0.007	P-value=0.008	P-value=0.009	P-value=0.01
d0=0.01	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.02	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.03	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.04	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.05	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.06	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.07	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.08	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.09	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.1	0.5210	0.5210	0.5210	0.5210	0.5210
	P-value=0.011	P-value=0.012	P-value=0.013	P-value=0.014	P-value=0.015
d0=0.01	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.02	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.03	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.04	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.05	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.06	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.07	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.08	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.09	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.1	0.5210	0.5210	0.5210	0.5210	0.5210
	P-value=0.016	P-value=0.017	P-value=0.018	P-value=0.019	P-value=0.02
d0=0.01	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.02	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.03	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.04	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.05	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.06	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.07	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.08	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.09	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.1	0.5210	0.5210	0.5210	0.5210	0.5210
	P-value=0.021	P-value=0.022	P-value=0.023	P-value=0.024	P-value=0.025
d0=0.01	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.02	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.03	0.5229	0.5229	0.5229	0.5229	0.5229
d0=0.04	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.05	0.5210	0.5210	0.5210	0.5210	0.5210
d0=0.06	0.5210	0.5210	0.5210	0.5210	0.5210

[illegible]

[illegible]

[illegible]

Supplementary Material 2:

Selecting the adjusted P-value for *ctree(.)* for the lung cancer data

To decide which input values of “alpha” (the adjusted P-value threshold) are suitable for *ctree(.)*, we generated a matrix of the number of terminal nodes for all combinations from alpha = 0.11 to 0.9 (below). However, the resultant *ctree* had only one or two terminal nodes when “alpha < 0.64”. Slightly larger values of alpha yielded more reasonable trees with 3 terminal nodes ($0.64 \leq P^{Adj} \leq 0.67$, Figure 7a) and 4 terminal nodes ($0.67 < P^{Adj} \leq 0.90$, Figure 7b).

```
> Terminal_node_size
P-threshold=0.11 P-threshold=0.12 P-threshold=0.13 P-threshold=0.14
1 1 1 1
P-threshold=0.15 P-threshold=0.16 P-threshold=0.17 P-threshold=0.18
1 1 1 1
P-threshold=0.19 P-threshold=0.2 P-threshold=0.21 P-threshold=0.22
1 1 1 1
P-threshold=0.23 P-threshold=0.24 P-threshold=0.25 P-threshold=0.26
1 1 1 1
P-threshold=0.27 P-threshold=0.28 P-threshold=0.29 P-threshold=0.3
1 1 1 1
P-threshold=0.31 P-threshold=0.32 P-threshold=0.33 P-threshold=0.34
1 1 1 1
P-threshold=0.35 P-threshold=0.36 P-threshold=0.37 P-threshold=0.38
1 1 1 1
P-threshold=0.39 P-threshold=0.4 P-threshold=0.41 P-threshold=0.42
1 1 1 1
P-threshold=0.43 P-threshold=0.44 P-threshold=0.45 P-threshold=0.46
1 1 1 1
P-threshold=0.47 P-threshold=0.48 P-threshold=0.49 P-threshold=0.5
1 1 1 1
P-threshold=0.51 P-threshold=0.52 P-threshold=0.53 P-threshold=0.54
2 2 2 2
P-threshold=0.55 P-threshold=0.56 P-threshold=0.57 P-threshold=0.58
2 2 2 2
P-threshold=0.59 P-threshold=0.6 P-threshold=0.61 P-threshold=0.62
2 2 2 2
P-threshold=0.63 P-threshold=0.64 P-threshold=0.65 P-threshold=0.66
2 3 3 3
P-threshold=0.67 P-threshold=0.68 P-threshold=0.69 P-threshold=0.7
3 4 4 4
P-threshold=0.71 P-threshold=0.72 P-threshold=0.73 P-threshold=0.74
4 4 4 4
P-threshold=0.75 P-threshold=0.76 P-threshold=0.77 P-threshold=0.78
4 4 4 4
P-threshold=0.79 P-threshold=0.8 P-threshold=0.81 P-threshold=0.82
4 4 4 4
P-threshold=0.83 P-threshold=0.84 P-threshold=0.85 P-threshold=0.86
4 4 4 4
P-threshold=0.87 P-threshold=0.88 P-threshold=0.89 P-threshold=0.9
4 4 4 4
```

Supplementary Material 3:

Additional simulation results under continuous covariates

The simulations of the main text were based on discrete covariates, namely, $x_{ij} = 1, 2, 3$, or 4 . This section considers additional simulations under “continuous” covariates.

We generated continuous covariates $x_{ij} \sim Unif(-1.5, 1.5)$ from the R function *X.pathway*(., $\rho_1 = 0.8$, $\rho_2 = 0.8$). Given the covariates, we generated T_i from a model $h(t|\mathbf{x}'_i) = h_0(t) \exp(\mathbf{x}'_i \boldsymbol{\beta})$ with $h_0(u) = 1$. Except for T_i and $\mathbf{x}'_i$, all other settings are unchanged.

To deal with continuous covariate by the proposed method, one needs to choose a sequence $c_1 < c_2 < \dots < c_m$ to discretize covariate values. We transform the continuous covariates into quantile levels, namely, 1, 2, 3, or 4 (~25th, 25th~50th, 50th~75th, or 75th~ percentile). We term this setting as “balanced” covariates. We also consider “unbalanced” covariates defined as 1, 2, 3, or 4 (~5th, 5th~50th, 50th~95th, or 95th~ percentile). Note that the ctrees is applied without any discretization.

Tables S1-S2 compare the performance of the proposed tree, the logrank tree, and the ctrees. If the covariates are balanced, the proposed tree and the logrank tree give almost identical performance in terms of the classification ability (LR test) and prognostic ability (c -index). In addition, the proposed tree produces better classification and prognostic ability than the ctrees. Thus, there is little loss of information by discretization.

If the covariates are unbalanced, the performance of all the proposed tree gets slightly lower due to the unbalanced discretization. In the presence of unbalanced covariates, the proposed tree exhibits modest advantage over the logrank tree in terms of the LR test and the c -index. The improved results come from the fact that the proposed tree avoids false nodes by stabilizing the test results for unbalanced samples. This is exactly what we expected. However, the improvement under non-sparse setting is weaker than that under the sparse setting (Table S1 vs. Table S2).

Overall, we conclude that the balanced discretization produces little effect on the performance of the proposed tree. Thus, one can safely discretize continuous covariates into quantile levels before applying the proposed tree. In addition, we conclude that the proposed tree remains advantageous over the logrank tree due to the stabilization of the unbalanced discretization. This feature is similar to that found in the simulation results of the main article.

Table S1. Simulation results comparing the four methods under the sparse setting (100 replications). The sample size is $n = 100$ and the censoring percentage is around 50%~56%. The score tests are stabilized by $d_0 = 0.01$ and 0.1 .

Split criterion: P-value<0.01			Logrank	Score 0.01	Score 0.1	ctree
$\beta = 1$	Balanced covariates	No. terminal nodes	19.02	18.48	13.24	3.67
		Precision%	0.405	0.422	0.528	-
		LR test: log ₁₀ (P)	-11.61	-11.61	-12.35	-11.17
		c-index	0.795	0.795	0.794	0.764
	Unbalanced covariates	No. terminal nodes	24.48	23.45	15.00	3.67
		Precision%	0.292	0.311	0.419	-
		LR test: log ₁₀ (P)	-10.30	-10.70	-12.07	-11.17
		c-index	0.778	0.782	0.793	0.764
$\beta = 0.5$	Balanced covariates	No. terminal nodes	18.63	18.11	12.83	3.33
		Precision%	0.358	0.366	0.445	-
		LR test: log ₁₀ (P)	-8.47	-8.36	-8.94	-8.56
		c-index	0.773	0.772	0.770	0.736
	Unbalanced covariates	No. terminal nodes	24.15	23.45	14.52	3.33
		Precision%	0.267	0.278	0.378	-
		LR test: log ₁₀ (P)	-7.35	-7.40	-8.51	-8.56
		c-index	0.753	0.754	0.763	0.736
Split criterion: P-value<0.005			Logrank	Score 0.01	Score 0.1	ctree
$\beta = 1$	Balanced covariates	No. terminal nodes	17.12	16.64	10.93	3.47
		Precision%	0.435	0.456	0.616	-
		LR test: log ₁₀ (P)	-11.81	-11.86	-12.39	-10.82
		c-index	0.794	0.794	0.791	0.759
	Unbalanced covariates	No. terminal nodes	22.86	21.62	12.55	3.47
		Precision%	0.300	0.324	0.481	-
		LR test: log ₁₀ (P)	-10.37	-10.77	-12.39	-10.82
		c-index	0.777	0.781	0.790	0.759
$\beta = 0.5$	Balanced covariates	No. terminal nodes	16.65	16.02	10.19	3.13
		Precision%	0.381	0.391	0.541	-
		LR test: log ₁₀ (P)	-8.55	-8.43	-9.28	-8.17
		c-index	0.772	0.771	0.768	0.730
	Unbalanced covariates	No. terminal nodes	22.44	21.43	11.99	3.13
		Precision%	0.282	0.297	0.436	-
		LR test: log ₁₀ (P)	-7.380	-7.510	-8.70	-8.17
		c-index	0.753	0.754	0.760	0.730

NOTE: No. terminal nodes=the number of terminal nodes in a tree; Precision=the number of the true splits in a tree divided by the number of splits in a tree. LR test: $\log_{10}(P)$ =the logarithmic value of the P-value of the LR test for testing the equality of the terminal nodes: $\log_{10}(0.01) = -2.0$ and $\log_{10}(0.001) = -3.0$.

Table S2. Simulation results comparing the four methods under the non-sparse setting (100 replications). The sample size is $n = 100$ and the censoring percentage is around 50%~56%. The score tests are stabilized by $d_0 = 0.01$ and 0.1 .

Split criterion: P-value<0.01			Logrank	Score 0.01	Score 0.1	ctree
$\beta = 0.1$	Balanced covariates	No. terminal nodes	17.34	16.82	11.96	3.70
		Precision%	0.723	0.740	0.799	-
		LR test: log ₁₀ (P)	-11.15	-11.27	-11.24	-10.24
		c-index	0.789	0.789	0.789	0.763
	Unbalanced covariates	No. terminal nodes	20.12	19.36	12.50	3.70
		Precision%	0.647	0.669	0.761	-
		LR test: log ₁₀ (P)	-10.15	-10.41	-11.22	-10.24
		c-index	0.787	0.789	0.793	0.763
$\beta = 0.05$	Balanced covariates	No. terminal nodes	17.43	16.93	11.32	3.01
		Precision%	0.641	0.659	0.713	-
		LR test: log ₁₀ (P)	-6.30	-6.62	-6.68	-6.09
		c-index	0.742	0.742	0.738	0.699
	Unbalanced covariates	No. terminal nodes	19.96	19.22	12.52	3.01
		Precision%	0.597	0.606	0.674	-
		LR test: log ₁₀ (P)	-6.14	-6.07	-6.57	-6.09
		c-index	0.739	0.739	0.744	0.699
Split criterion: P-value<0.005			Logrank	Score 0.01	Score 0.1	Ctree
$\beta = 0.1$	Balanced covariates	No. terminal nodes	15.50	14.89	10.13	3.51
		Precision%	0.747	0.768	0.855	-
		LR test: log ₁₀ (P)	-11.08	-11.01	-11.34	-9.91
		c-index	0.788	0.788	0.787	0.755
	Unbalanced covariates	No. terminal nodes	18.29	17.23	10.61	3.51
		Precision%	0.663	0.688	0.813	-
		LR test: log ₁₀ (P)	-10.15	-10.39	-11.41	-9.91
		c-index	0.787	0.787	0.790	0.755
$\beta = 0.05$	Balanced covariates	No. terminal nodes	14.81	14.18	8.88	2.90
		Precision%	0.669	0.692	0.788	-
		LR test: log ₁₀ (P)	-6.52	-6.71	-6.73	-5.60
		c-index	0.740	0.740	0.736	0.695
	Unbalanced covariates	No. terminal nodes	17.84	16.90	9.93	2.90
		Precision%	0.606	0.613	0.730	-
		LR test: log ₁₀ (P)	-6.23	-6.18	-6.54	-5.96
		c-index	0.736	0.737	0.741	0.695

NOTE: No. terminal nodes=the number of terminal nodes in a tree; Precision=the number of the true splits in a tree divided by the number of splits in a tree. LR test: $\log_{10}(P)$ =the logarithmic value of the P-value of the LR test for testing the equality of the terminal nodes: $\log_{10}(0.01) = -2.0$ and $\log_{10}(0.001) = -3.0$.