BART-Survival: A Bayesian machine learning approach to survival analyses in Python

08 August 2024

Summary

BART-Survival is a Python package that allows time-to-event (survival) analyses in discrete-time using the non-parametric machine learning algorithm, Bayesian Additive Regression Trees (BART). BART-Survival combines the performance of the BART algorithm with the complementary data and model structural formatting required to complete the survival analyses. The library contains a convenient application programming interface (API) that allows a simple approach when using the library for survival analyses, while maintaining capabilities for added complexity when desired. The package is intended for analysts exploring use of flexible non-parametric alternatives to traditional (semi-)parametric survival analyses.

Statement of need

Survival analyses are a cornerstone of public health and clinical research in such diverse fields as cancer, cardiovascular disease, and infectious diseases [@altman1998; @bradburn2003]. Traditional parametric and semi-parametric statistical methods, such as the Cox proportional hazards model, are commonly employed for survival analyses [@cox1972]. However, these methods have several limitations, particularly when applied to complex data. One major issue is the need for restrictive assumptions, such as proportional hazards and predefined functional forms, which may not hold true in complex, real-world healthcare data [@ishwaran2008; @harrell2015]. Additionally, these methods often struggle with high-dimensional datasets, leading to problems with overfitting, multicollinearity, and dealing with complex interactions [@ishwaran2008; @joffe2013].

More recently, non-parametric machine learning approaches have been introduced to address these limitations by reducing the need for restrictive assumptions and providing increased capabilities for more accurately modeling underlying distributions and complex interactions [@ishwaran2008; @harrell2015]. BART is one such machine learning method that has demonstrated utility in the survival setting through its performance in identifying underlying statistical distributions [@chipman2010; @sparapani2021]. BART offers flexibility in modeling complex relationships and interactions within the data without requiring the specification of a particular functional form [@sparapani2016].

Currently, the only BART survival algorithm readily available exists as part of the `BART` R package, which contains a library of various BART-based approaches in addition to a BART survival analysis application [@sparapani2016; @sparapani2021]. BART-Survival package described here combines the survival analysis approach outlined in the `BART` R package with the foundational Python-based probabilistic programming language library, `PyMC`, and the accompanying BART algorithm from the `PyMC-BART` library. Our aim in developing `BART-Survival` is to provide accessibility to the BART survival algorithm within the Python programing language. This contribution is beneficial for analysts when Python is the preferred programming language, the analytic workflow is Python-based, or when the R language is unavailable for analyses. Additionally, `BART-Survival` package abstracts away the complexities of the `PyMC` and `PyMC-BART` libraries through use of a pre-specified core model and generalized functionality that can accommodate analyses across various survival settings. The `BART-Survival` package is intended for public health and clinical professionals and students who are looking for non-parametric alternatives to traditional (semi-)parametric survival analysis, especially for use in large, complex healthcare data and machine learning applications.

Methods

The following sections provides details on the methods employed by the BART-Survival library, focusing specifically on the discrete-time Survival algorithm and the produced estimates. For review of the BART algorithm used we refer to PyMC-BART publication [@quiroga2023].

Background

The `BART-Survival` package provides a discrete-time Survival method which aims to model Survival as a function of the cumulative risk of event occurrences over the series of discrete time intervals.

Using discrete-time intervals provides a convenient approach that allows flexible modeling of the latent probability of an event as a non-parametric function of the distinct time interval and a set of observation covariates. The latent probabilities can then be used for deriving Survival probability or other estimates.

The foundation of the method is simple.
1. Create a sequence of time intervals, denoted as t_j with $(j = 1, ..., k)$, from the range of observed event times. 2. Then for each interval t_j obtain the number of observations with an event, along with the total number of observations at risk

for having an event. 3. Finally, the risk of event occurrence within each interval t_j can naively be derived as:

```
\begin{equation}
P_{t_j} = \frac {\text{n events}_{t_j}} {\text{n at risk}_{t_j}}
\end{equation}
```

and the Survival probability $S(t)$ at a time q, can be derived as:

```
\begin{equation}
S(t_q) = \prod_{j=1}^{q} (1-P_{t_j})
\end{equation}
```

`BART-Survival` builds off this simple foundation by replacing P_t with a probability risk estimate, $p_{t_j|x_i}$ yielded from the BART regression model. The predicted values $p_{t_j|x_i}$ are generated for each observation, at each time interval from the set j. Downstream targets can be further derived from these predicted values with observation-level Survival derived as:

```
\begin{equation}
S(t_q|x_i) = \prod_{j=1}^{q} (1-p_{t_j|x_i})
\end{equation}
```

Statistical estimands can also be estimated using the predicted $p_{t_j|x_i}$ through evaluation of marginal functions of the predicted values.

Data Preparation

To properly model $p_{t|x}$ the data requires an transformation from a standard Survival dataset to a *augmented* dataset. Survival data is typically given as a paired (**event status**, **event time**) outcome, along with a set of covariates for each observation. In this setup **event status** is typically a binary variable (1=event; 0=censored) and **event time** is some continuous representation of time.

The *augmented* dataset transforms the generic dataset from a single, paired (**event status**, **event time**) outcome per observation, to a sequence of single (**event status**) outcomes over the series of discrete-time intervals, up to the given **event time** for the observation.

For example if the unique set of a dataset's **event times** is **{4,6,7,8,12,14}**, and a single observation's paired outcome is (**event status** = 1, **event times** = 12), then the single observation will be represented in the *augmented* dataset as the sequence of rows:

```
\begin{matrix}
\text{event status} &\text{time} \\
 --- &---\\
    0 & 4\\
    0 & 6 \\
    0 & 7 \\
    0 & 8 \\
    1 & 12 \\
\end{matrix}
```

Each row in the *augmented* dataset can then be treated as an independent observation, with **event status** as the outcome Y and the **event time** T as an added covariate. Now each of the original observations are represented by j rows ($j = 1, ..., i_{\text{event time}}$) and the corresponding variables can be denoted as (y_{ij}, t_j, x_{ij}).

Model

Using the new *augmented* dataset, the model is simplified to a probit regression of y_{ij} on time t_j and covariates x_{ij}, which yields a latent value p_{ij} corresponding to $P(y_{ij} = 1)$. Explicitly the model is defined as:

```
\begin{align*}
    y_{ij} | p_{ij} \sim Bernoulli(p_{ij}) \\
    p_{ij} | \mu_{ij} = \Phi(\mu_{ij})\\
    \mu_{ij} \sim \text{BART}(j,x_{i})\\
\end{align*}
```

where Φ is the CDF of the Normal distribution.

A trained `BART-Survival` model can then be used to yield the p_{ij} predictions, which can be used to derive Survival as described above.

Inference

A common goal of Survival regression models is to derive a statistical estimate for the adjusted effect of a variable on the outcome. A classic example of these estimates are Hazard Ratios derived from the coefficients of Cox Proportional Hazard Models.

With `BART-Survival`, the underlying BART algorithm does not rely on a linear equation and therefore does not produce coefficient that can be treated as conditional measures of effects. Instead, variable effects are summarized as marginal effect estimates which can derived through use of partial dependence functions.

The partial dependence function method is relatively simple. It involves generating predictions of p from a trained `BART-Survival` model using a *augmented partial dependence* (*APD*) dataset as input.

The *APD* dataset is generated so that a specific variable $x_{[I]}$ is deterministically set to a specific value for all observations while the other covariates ‘$x_{[O]_i}$‘ remain as observed. Each observation is then expanded over the time-intervals

$1, ..., j_{T_{max}}$ to create the discrete-time datasets. Here ‘$j_{T_{max}}$‘ is the maximum time across all **event times**.

To estimate the effect of an variable on the outcome, multiple *APD* datasets are created. Each *APD* dataset varies the value of the specific variable of interest (i.e. ‘$x_{[I]_1}$‘, ‘$x_{[I]_2}$‘), allowing evaluation of the outcome under different conditions of that variable. The p_{ij} values from each predicted dataset ($p_{[1]}$, $p_{[2]}$), can be used to obtain the marginal estimates of a specific outcome.

Common marginal effect estimates derived from these predicted values include:

- Marginal difference is Survival probability at time j:

```
\begin{equation}
\text{Surv Diff}_{marg} = E_{i}[S_{p_{[2]}}(t_j)] - E_{i}[S_{p_{[1]}}(t_j)]
\end{equation}
```

- Marginal Risk Ratio at time j:

```
\text{RR}_{marg} = \frac {E_{i}[p_{[2]_{j}}]} {E_{i}[p_{[1]_{j}}]}
```

- Marginal Hazard Ratio (assuming constant hazard rates):

```
\text{HR}_{marg} = \frac {E_{ij}[p_{[2]}]} {E_{ij}[p_{[1]}]}
```

For users familiar with causal inference literature, these partial dependence functions are similar to the g-estimation and counterfactual outcomes methods common to the causal inference field.

In addition to providing point estimates, the `BART-Survival` naturally generates Bayesian credible intervals as a product of the posterior predictive distribution. The credible intervals can provide useful measure of uncertainty and allows for bayesian variants of hypothesis testing and statistical inference to be made on the estimates.

Summary The `BART-Survival` package provide the algorithms necessary to complete the 3 major steps of the Survival analysis.

1. Generate augmented dataset
2. Train-predict-transform the BART model and estimates
3. Generate *augmented partial dependece* datasets and generating marginal estimates.

Acknowledgements

We thank Oscar Rincón-Guevara for helpful suggestions and review. We also thank Tegan Boehmer, Sachin Agnihotri, and Matt Ritchey for supporting the project throughout its development.