Imprecise Extensions of Random Forests and Random Survival Forests

Lev V. Utkin, Maxim Kovalev, Anna Meldo, Frank Coolen

ISIPTA, Gent, July 2019
Authors are from …

1. Peter the Great St. Petersburg Polytechnic University
2. Saint-Petersburg Clinical Research Center for Special Types of Medical Care (Oncological)
3. Durham University

Lev Utkin  Maxim Kovalev  Anna Meldo  Frank Coolen
Classification problem statement

- Given $N$ training examples $S = \{(x_1, y_1), ..., (x_N, y_N)\}$, $x_i \in \mathbb{R}^m$, $y_i \in \{1, ..., C\}$
- We aim to construct an accurate classifier $c : \mathbb{R}^m \rightarrow \{1, ..., C\}$
- An ensemble-based classifier is the Random Forest (RF)
Class probabilities are defined by numbers of training examples which fall into leaf nodes.

The class probabilities of the RF are computed by averaging probabilities of trees.
Weighted Random Forest

- Weights of trees and weighted averaging

Obstacles:
- The number of training examples which fall into a leaf node may be very small
- Precise class probabilities cannot be expected
Two ways for solving the problem

1. The first way is to change splitting rules for tree building (Abellan et al. 2017, Abellan et al., Mantas-Abellan, 2014)
   - it leads to changing a tree building algorithm
   - it cannot be directly applied to regression or survival analysis

2. The second way is to train a meta-learner which takes into account imprecision of the class probabilities
Meta-learner: Underlying ideas

1. the meta-learner produces weights of trees or corresponding class probabilities
2. imprecision of the tree estimates (class probabilities) is defined by an imprecise statistical inference model, for example, the IDM
3. robust (pessimistic or maximin) strategy should be applied to the meta-learner
4. special loss functions should be proposed to simplify optimization problems for computing optimal weights
An example of the whole classifier

Imprecise Dirichlet model $s=1$

$\mathbf{x}_i$
Loss function

- Weights are taken
  1. to minimize Euclidean distance between a training class vector and the obtained class probabilities
  2. to maximize the distance over “imprecise” sets of tree class probabilities

- A standard way for constructing maximin optimization problem:

\[
\max_{P(i,t) \in \mathcal{P}_{i,t}(s)} \min_{w \in \mathcal{W}(u)} \sum_{i=1}^{N} \sum_{c=1}^{C} (\langle p_{i,c}, w \rangle - l_{c}(y_i))^2 + \lambda \|w\|^2
\]

- A new way:

\[
\max_{P(i,t) \in \mathcal{P}_{i,t}(s)} \min_{w \in \mathcal{W}(u)} \sum_{i=1}^{N} \left\langle w, 1 - \sum_{c=1}^{C} l_{c}(y_i) p_{c}(i, t) \right\rangle + \lambda \|w\|^2
\]

We consider only the class probability corresponding to class \(y_i\) of the \(i\)-th training example and find how it is far from 1
The quadratic optimization problem for computing optimal weights

$$\min_{w \in W(u)} \left( \lambda \| w \|^2 - \sum_{t=1}^{T} w_t \sum_{i=1}^{N} p_{y_i}^*(i, t) \right)$$

$p_{y_i}^*(i, t)$ are smallest values of probabilities from extreme points of $\mathcal{P}_{i.t}(s)$ (from an imprecise statistical model)
Survival analysis (problem statement)

- It is solved by the Random Survival Forest (RSF)
- Given \( N \) training examples \( S = \{(x_1, \delta_1, D_1), ..., (x_n, \delta_n, D_n)\} \), \( x_i \in \mathbb{R}^m \), \( y_i \in \{1, ..., C\} \)
- \( D_i \) indicates time to event of the patient
- If the event of interest is observed, \( \delta_i = 1 \), (an uncensored observation); if the event is not observed, \( \delta_i = 0 \) (a censored observation)
- A specific regression problem, where we compute the cumulative hazard estimate \( H_k(t) \) for every leaf node \( k \) by means of the Nelson-Aalen estimator
How to take into account imprecision of the cumulative hazard estimate $H_k(t)$?

The Nelson–Aalen estimator has a standard $100(1 - \alpha)\%$ confidence interval for $H_k(t)$:

$$H_k(t) \pm z_{1-\alpha/2} \cdot \sigma_k(t)$$

where $z_{1-\alpha/2}$ is the $1 - \alpha/2$ fractile of the standard normal distribution, $\sigma(t)$ is the variance of the Nelson-Aalen estimator.

We get intervals $B_k = [\underline{H}_k(t), \overline{H}_k(t)]$
Now we have a new measure of the model quality

The C-index estimates how good the model is at ranking survival times

It estimates the probability that, in a randomly selected pair of patients, the patient that fails first had a worst predicted outcome

\[ C = \frac{1}{M} \sum_{i: \delta_i = 1} \sum_{j: t_i < t_j} 1 \left[ S(t_i^* | x_i) > S(t_j^* | x_j) \right] \rightarrow \max \]
Optimization problem with a modified C-index

- Optimization problem with the standard C-index

\[
\min_{H_q(t|x) \in \mathcal{B}_q} \max_{w \in \mathcal{W}(u)} \sum_{(i,j) \in J} 1 \left[ \sum_{q=1}^{T} w_q \left( H_q(t^*_j|x_j) - H_q(t^*_i|x_i) \right) \right] \geq 0
\]

- A modified C-index:

\[
C_{new} = \frac{1}{M} \sum_{(i,j) \in J} \left( H_q(t^*_j|x_j) - H_q(t^*_i|x_i) \right)
\]

- Optimization problem with the modified C-index

\[
\max_{H_q(t|x) \in \mathcal{B}_q} \min_{w \in \mathcal{W}(u)} \sum_{q=1}^{T} w_q \sum_{(i,j) \in J} \left( H_q(t^*_i|x_i) - H_q(t^*_j|x_j) \right)
\]

- Finally:

\[
\min_{w \in \mathcal{W}(u)} \left( \sum_{q=1}^{T} w_q B^*_q + \lambda \|w\|^2 \right)
\]
Some numerical experiments
Questions

Thank you for your attention