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

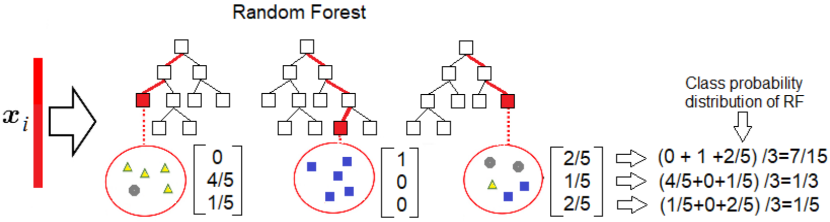
Anna Meldo

Frank Coolen

Classification problem statement

- Given N training examples $S = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)\}$, $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \{1, ..., C\}$
- We aim to construct an accurate classifier $c : \mathbb{R}^m \to \{1, ..., C\}$
- An ensemble-based classifier is the Random Forest (RF)

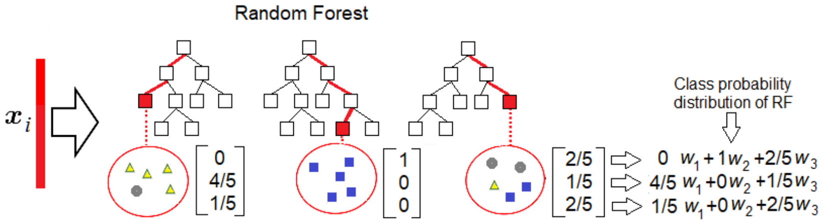
Random Forest



- 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

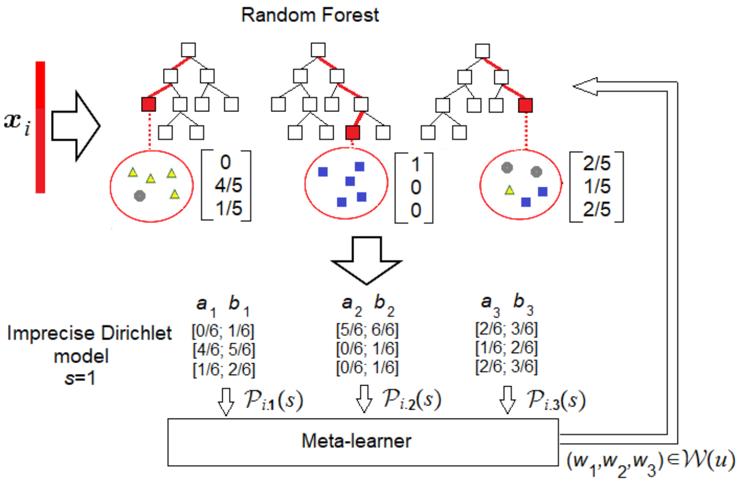
Two ways for solving the problem

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



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:

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

- A new way:

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

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

A simplified quadratic optimization problem

- The quadratic optimization problem for computing optimal weights

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

- $p_{y_i}^{*}(i,t)$ 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 $S = \{(\mathbf{x}_1, \delta_1, D_1), ..., (\mathbf{x}_n, \delta_n, D_n)\}$, $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \{1, ..., C\}$
- 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

Interval-valued hazard estimate

- 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 $\mathcal{B}_k = [\underline{H}_k(t), \overline{H}_k(t)]$

Measure of the model quality

- 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

$$C = \frac{1}{M} \sum_{i:\delta_i=1} \sum_{j:t_i<t_j} \mathbf{1}\left[S(t_i^*|\mathbf{x}_i) > S(t_j^*|\mathbf{x}_j)\right] \to \max$$

Optimization problem with a modified C-index

- Optimization problem with the standard C-index

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

- A modified C-index:

$$C_{\text{new}} = \frac{1}{M}\sum_{(i,j)\in J}\left(H_q(t_j^*|\mathbf{x}_j) - H_q(t_i^*|\mathbf{x}_i)\right)$$

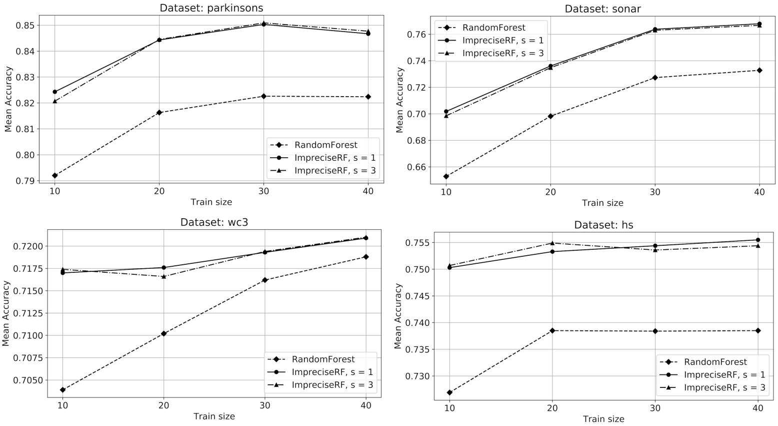
- Optimization problem with the modified C-index

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

- Finally:

$$\min_{\mathbf{w}\in\mathcal{W}(u)}\left(\sum_{q=1}^{T} w_q B_q^* + \lambda\left\|\mathbf{w}\right\|^2\right)$$

Some numerical experiments



Questions

Thank you for your attention

?