DOI 10.15622/ia.21.5.1

L. Utkin , A. Konstantinov

RANDOM SURVIVAL FORESTS INCORPORATED BY THE NADARAYA-WATSON REGRESSION

Utkin L., Konstantinov A. **Random Survival Forests Incorporated by the Nadaraya-Watson Regression.**

Abstract. An attention-based random survival forest (Att-RSF) is presented in the paper. The first main idea behind this model is to adapt the Nadaraya-Watson kernel regression to the random survival forest so that the regression weights or kernels cannnbsp;be regarded as trainable attention weights under important condition that predictions of the random survival forest are represented in the form of functions, for example, the survival function and the cumulative hazard function. Each trainable weight assigned to a tree and a training or testing example is defined by two factors: by the ability of corresponding tree to predict and by the peculiarity of an example which falls into a leaf of the tree. The second main idea behind Att-RSF is to apply the Huber's contamination model to represent the attention weights as the linear function of the trainable attention parameters. The Harrell's C-index (concordance index) measuring the prediction quality of the random survival forest is used to form the loss function for training the attention weights. The C-index jointly with the contamination model lead to the standard quadratic optimization problem for computing the weights, which has many simple algorithms for its solution. Numerical experiments with real datasets containing survival data illustrate Att-RSF.

Keywords: machine learning, random survival forest, survival analysis, Harrell's C-index, cumulative hazard function, attention mechanism, Huber's contamination model.

1. Introduction. Survival analysis can be regarded as an important and fundamental tool for modelling applications using time-to-event data [1]. In various spheres of life, including, medicine, reliability, safety, finance and economics, we encounter time-to-event data. Therefore, many machine learning models have been proposed to deal with time-to-event data and to solve the corresponding problems in the framework of survival analysis [2–5].

Three types of survival models can be considered [4]. Models of the first type, called parametric models, assume that a probability distribution of time to event is known, but its parameters are unknown and should be estimated. Models of the second type, called semi-parametric models, do not assume any probability time-to-event distribution, but assume that there is some known functional dependence between covariates and the model outcomes. The well-known semi-parametric model is the Cox proportional hazards model [6] which can be regarded as a regression model. Models of the third type, called non-parametric, do not use any information about a probability time-to-event distribution as well as a relationship between covariates and the model outcomes. The well-known non-parametric survival model is the Kaplan-Meier model [4]. An important peculiarity of many survival models is that their outcomes are functions, for example, survival functions, hazard functions, cumulative hazard functions, but not point-valued data.

Following the Cox model, many of its modifications overcoming some disadvantages of the Cox model have been developed, for example, models based on the Lasso method [7], models generalizing the Cox model by using neural networks [3], the support vector machine [8], survival trees [9], random survival forests (RSFs) [10] as an extension of the original random forest (RF) [11]. Due to the small number of tuning parameters, due to the ability to deal with both low and high-dimensional data, due to adaptability to data, RSFs became a popular tool for survival analysis of time-to-event data in many applications. RSFs have demonstrated their efficiency in solving many real problems [12–18].

One of the ways to improve RSF is to replace the standard averaging with the weighted sum of the tree survival functions. Following this idea, the corresponding weighted RSF was proposed in [19]. According to the weighted RSF, every tree is assigned by a weight which is computed by solving an optimization problem maximizing the concordance error rate called C-index [20]. The main disadvantage of the weighted RSF is that it uses weights which do not depend on each example and are defined only by the corresponding survival tree. This fact reduces the RSF accuracy. In order to overcome this difficulty, we propose the attention-based RSF (Att-RSF). The idea behind Att-RSF is to adapt the Nadaraya-Watson regression to the original RSF. In other words, every survival tree jointly with an example, which falls into the tree, is considered as a term in the Nadaraya-Watson regression with a weight which is trained through its trainable parameters. The idea to assign weights to trees in the RF in accordance with the tree importance and with the example importance is not new, and it was proposed in [21] where attention weights are trained by solving the quadratic optimization problem. It turns out that this idea to consider the RF as the Nadaraya-Watson regression can be extended to RSF taking into account the RSF peculiarities which differ RSF from the RF. In particular, we propose to optimize the model parameters in accordance with the C-index as a measure of the RSF accuracy instead of the simple difference between predicted values and true labels used in the RF. This leads to a quite different optimization problem.

Similar to the attention-based RF [21], the Huber's ϵ-contamination model [22] is introduced to define the trainable parameters of the attention weights such that these trainable parameters of weights are optimally selected from an arbitrary adversary distribution. The ϵ-contamination model allows us to introduce weights being a linear function of the C-index.

Our contributions can be summarized as follows:

1. A new attention-based RSF model is proposed. According to the model, the trainable attention mechanism is incorporated into RSF to improve the accuracy of obtained predictions.

2. The proposed attention-based RSF can be regarded as an adaptation of the Nadaraya-Watson kernel regression to RSF. Moreover, we extend the Nadaraya-Watson kernel regression to predictions in the form of functions, for example, the cumulative hazard function.

3. Numerical experiments with real datasets are provided to justify Att-RSF, and to compare it with original RSFs [10] and the weighted RSF proposed in [19].

The paper is organized as follows. Related work devoted to machine learning models in survival analysis, the attention mechanism and the weighted RFs can be found in Section 2. Section 3 provides basic definitions of survival analysis, RSFs and the Nadaraya-Watson regression jointly with the attention mechanism. The main ideas of the Att-RSF and algorithms for training optimal attention parameters are considered in Section 4. Numerical experiments with well-known public real data illustrating the proposed Att-RSF model and comparing it with the available survival machine learning models are given in Section 5. Concluding remarks are provided in Section 6.

2. Related work. Machine learning models in survival analysis. Many survival machine learning models dealing with time-to-event data have been developed and investigated to predict survival time or other survival measures. A comprehensive review of the recent survival machine learning models is presented by [4]. The most popular survival model is the semi-parametric Cox proportional hazards model [6] which establishes a linear relationship between the covariates and the distribution of survival times. Tibshirani [7] presented a modification based on the Lasso method. Similar Lasso modifications, for example, the adaptive Lasso, were also proposed by several authors [23, 24] The linear relationship can be viewed in some applications as a disadvantage which can be resolved by relaxing the linear relationship assumption and extending the Cox model to more complex models [2,25,26]. At the present time, survival models can be regarded as extensions of many well-known machine learning models, for example, the Lasso models [23], SVM [8], decision trees [9], neural networks [2,26,27], etc.

We pay attention to random survival forests (RSFs) which can be regarded as one of the most powerful and efficient tools for survival analysis especially when the training data are tabular. Various implementations and modifications of RSFs were considered and studied in [14,15,17,19,28,29]. To improve the available RSF models, it is proposed to incorporate the attention mechanism with trainable parameters into RSFs, which allows us to take into account the importance of trees in RSF as well as the importance of every training or testing example.

Attention mechanism. Many attention-based models have been developed to improve the performance of classification and regression algorithms. Detailed and comprehensive surveys of attention models can be found in [30–35]. It is important to point out that attention models are mainly applied to the natural language processing, including text classification, translation, etc., to the computer vision area, including image-based analysis, visual question answering, etc. However, time-to-event data in many applications have a tabular form. An attempt to incorporate the attention mechanism into the RF was made in [21]. Following this work, we try to incorporate the attention mechanism into RSF by using peculiarities of survival models which include: predictions in the form of functions of time, the model accuracy measure in the form of the C-index, and censored data. The proposed attention mechanism can be also regarded as an extension of the weighted RFs.

Weighted RFs. Various models and methods have been developed to implement the weighted RFs. They can be divided into two groups. The first group consists of models which are based on assigning weights to decision trees in accordance with some criteria to improve the classification and regression models [36–38]. This group contains models using weights of classes to take into account imbalanced datasets [39]. However, the assigned weights in the aforementioned works are not trainable parameters. Therefore, models [40,41] from the second group use trainable weights of trees such that the weights are trained by solving optimization problems in accordance with a certain loss function for the whole RF. The model of the weight assigning in [21] differs from the above models because weights are assigned depending on trees and each example.

Our aim is to incorporate the attention weights with trainable parameters into RSF and to propose simple algorithms for training the parameters.

3. Preliminaries.

3.1. Survival analysis. The i-th patient in survival analysis is represented by a triplet $(\mathbf{x}_i, \delta_i, T_i)$, where $\mathbf{x}_i \in \mathbb{R}^m$ is the feature vector characterizing the patient; T_i is the time to an event of interest; δ_i is the indicator of event observation, in particular, $\delta_i = 1$ if the corresponding event is observed (an uncensored observation), $\delta_i = 0$ if the event is not observed and the corresponding time to event is greater than T_i (a censored observation). Survival analysis aims to estimate time T to the event for a new patient having feature vector $\mathbf{x}$ on the basis of a training set D consisting of n triplets $(\mathbf{x}_i, \delta_i, T_i)$,

$i = 1, ..., n$. We will use the term "patient" to represent arbitrary subjects or objects.

It is important to point out that only a part of the patients will experience the event of interest during the course of the experiment. Other patients will not experience the event of interest after the expiration of the study. Therefore, observation is said to be censored in survival analysis when information on time to the corresponding event of interest is not available, i.e., we have some information about the patient survival time, but we do not know the survival time exactly. One should distinguish between right censoring, left censoring and interval censoring. Right censoring occurs when the study of a patient ends before the event has occurred. Left censoring is when the event of interest has already occurred before studying. This is a very rare case. Interval censoring is a combination of left and right censoring. It should be noted that right censoring is the most common type of censoring, therefore, we consider only this type. Parameter δ_i indicates whether the i-th event is censored or not.

Important concepts in survival analysis are the survival function (SF) and the cumulative hazard function (CHF). The SF $S(t|\mathbf{x})$ is a function of time t defined as the probability of surviving up to time t, i.e.: $S(t|\mathbf{x}) = \Pr\{T > t|\mathbf{x}\}$. The CHF $H(t|\mathbf{x})$ is also a function of time defined through the SF as follows:

$$H(t|\mathbf{x}) = -\ln S(t|\mathbf{x}). \tag{1}$$

Many survival machine learning models have been developed in the last decades. In order to compare the models, special measures are used differently from the standard accuracy measures accepted in machine learning classification and regression models. The most popular measure in survival analysis is Harrell's C-index (concordance index) [20]. It estimates the probability that, in a randomly selected pair of patients, the patient that fails first had the worst predicted outcome. In fact, this is the probability that the event times of a pair of patients are correctly ranked. C-index does not depend on choosing a fixed time for evaluation of the model and takes into account censoring of patients [42].

Let us consider the training set D consisting of n triplets $(\mathbf{x}_i, \delta_i, T_i)$. We consider possible or admissible pairs $\{(\mathbf{x}_i, \delta_i, T_i), (\mathbf{x}_j, \delta_j, T_j)\}$ for $i \leqslant j$. Then the C-index is calculated as the ratio of the number of pairs correctly ordered by the model to the total number of admissible pairs. A pair is not admissible if the events are both right-censored or if the earliest time in the pair is censored. If the C-index is equal to 1, then the corresponding survival model is supposed to be perfect. If the C-index is 0.5, then the model is not better than random guessing.

Let $t_1, ..., t_N$ denote predefined time points of the corresponding N distinct event times. If the output of a survival model is the predicted SF $S(t)$, then the C-index is formally calculated as [4]:

$$C = \frac{1}{M} \sum_{i:\delta_i=1} \sum_{j:t_i<t_j} \mathbf{1}\left[S(t_i|\mathbf{x}_i) - S(t_j|\mathbf{x}_j) > 0\right]. \tag{2}$$

Here M is the number of all comparable or admissible pairs; $\mathbf{1}[\cdot]$ is the indicator function taking value 1 if its argument is true, and 0 if the argument is false.

3.2. Random survival forests. In spite of the efficiency of deep neural networks, RSFs can be regarded as one of the best models for survival analysis due to their properties especially when the tabular training data are used. Therefore, we modify RSFs to improve their prediction capacity.

A general algorithm for constructing RSFs can be represented as follows [43]:

1. Q subsets of training data are selected to build Q trees in RSF. Each subset excludes on average 37% of the data, is called out-of-bag data (OOB data).

2. Each survival tree is built on the corresponding subset. At each node of the tree, $\sqrt{m}$ candidate variables are randomly selected. The node is split using the candidate variable that maximizes the survival difference between daughter nodes.

3. Each tree is built to full size under the constraint that a terminal node should have no less than $d > 0$ unique events. Here d is a tuning parameter which is chosen to get the best results.

4. CHFs or SFs are calculated for each tree. The ensemble CHF or the ensemble SF are obtained by averaging CHFs or SFs of trees.

5. Using out-of-bag data, prediction errors for the ensemble CHF or the ensemble SF are calculated.

The accuracy of RSF predictions is defined by a splitting rule. A good split maximizes survival difference across the two sets of data [43]. There are several splitting rules used in RSF [4, 43]. We do not consider them because the proposed approach does not depend on a splitting rule.

Before computing the ensemble CHF or the ensemble SF having CHF or SFs of trees, we consider how to compute the CHF for the k-th terminal node of a tree. Let $\{t_{j,k}\}$ be a set of $N(k)$ distinct event times in terminal node k of the q-th tree such that $t_{1,k} < t_{2,k} < ... < t_{N(k),k}$ and $Z_{j,k}$ and $Y_{j,k}$ equal to the number of events and patients at risk at time $t_{j,k}$. The CHF for node k is defined by using the Nelson–Aalen estimator as follows:

$$H_k(t) = \sum_{t_{j,k} \leqslant t} Z_{j,k}/Y_{j,k}. \tag{3}$$

If the i-th patient with features $\mathbf{x}_i$ falls into node k, then one can say that $H(t|\mathbf{x}_i) = H_k(t)$. The ensemble CHF for the i-th patient is obtained by averaging CHFs of all Q trees, i.e.,

$$H_f(t|\mathbf{x}_i) = \frac{1}{Q}\sum_{q=1}^{Q} H_q(t|\mathbf{x}_i). \tag{4}$$

The SF can be obtained from $H_q(t|\mathbf{x}_i)$ as follows:

$$S_q(t|\mathbf{x}_i) = \exp\left(-H_q(t|\mathbf{x}_i)\right). \tag{5}$$

Ishwaran et al. [43] proposed another ensemble estimate using OOB data. Suppose that tree q is built on a set of OBB examples with indices from set O_q. The OOB prediction for each training example $\mathbf{x}_i$ uses only the trees that did not have $\mathbf{x}_i$ in their bootstrap sample. If to denote the indicator function as $\mathbf{1}(i \in O_q)$, then the OOB ensemble CHF for the i-th training example is estimated as:

$$H_f(t|\mathbf{x}_i) = \frac{\sum_{q=1}^{Q} \mathbf{1}(i \in O_q) \cdot H_q(t|\mathbf{x}_i)}{\sum_{q=1}^{Q} \mathbf{1}(i \in O_q)}. \tag{6}$$

3.3. Attention mechanism and the Nadaraya-Watson regression.

The idea of the attention mechanism can clearly be explained by using the Nadaraya-Watson kernel regression model [44,45]. If there is a training set $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$ consisting of n examples, where $\mathbf{x}_i \in \mathbb{R}^m$ is a feature vector and $y_i \in \mathbb{R}$ is the corresponding label, then the regression output prediction z, associated with a new input feature vector $\mathbf{x}$, can be estimated as the weighted average in the form of the Nadaraya-Watson kernel regression model [44,45]:

$$z = \sum_{i=1}^{n} \alpha(\mathbf{x}, \mathbf{x}_i) y_i. \tag{7}$$

Here $\alpha(\mathbf{x}, \mathbf{x}_i)$ is the attention weight which measures how the feature vector $\mathbf{x}$ is far from the feature vector $\mathbf{x}_i$ from the training set. The closer $\mathbf{x}$ to $\mathbf{x}_i$, the greater the corresponding weight $\alpha(\mathbf{x}, \mathbf{x}_i)$. Generally, arbitrary distance functions satisfying the above condition can be regarded as the attention weights. One of the sets of the functions is the kernel set because a kernel K can be regarded as a scoring function estimating how vector $\mathbf{x}_i$ is close to vector $\mathbf{x}$. Hence, the attention weights can be represented as:

$$\alpha(\mathbf{x}, \mathbf{x}_i) = \frac{K(\mathbf{x}, \mathbf{x}_i)}{\sum_{j=1}^{n} K(\mathbf{x}, \mathbf{x}_j)}. \tag{8}$$

In terms of the attention mechanism [46], vector $\mathbf{x}$, vectors $\mathbf{x}_i$ and labels y_i are called *query*, *keys* and *values*, respectively. Weights $\alpha(\mathbf{x}, \mathbf{x}_i)$ can be extended by incorporating trainable parameters. For example, if we take the Gaussian kernel with a trainable vector of parameters $\mathbf{w} = (w_1, ..., w_n)$, then the attention weight can be represented as:

$$\begin{gathered}\alpha(\mathbf{x}, \mathbf{x}_i, \mathbf{w}) = \\ = \text{softmax}\left(-\left\|\mathbf{x} - \mathbf{x}_i\right\|^2 |\mathbf{w}\right) = \frac{\exp\left(w_i\|\mathbf{x}-\mathbf{x}_i\|^2\right)}{\sum_{j=1}^{n}\exp\left(w_i\|\mathbf{x}-\mathbf{x}_i\|^2\right)}.\end{gathered} \tag{9}$$

There exist several definitions of attention weights and the corresponding attention mechanisms, for example, the additive attention [46], multiplicative or dot-product attention [47,48]. We use a new attention mechanism which is based on the weighted RSFs training and the Huber's ϵ-contamination model.

4. Attention-based RSF.

4.1. Queries, keys and values in RSFs. The main idea behind Att-RSF is to adapt the Nadaraya-Watson regression to the original RSF. It can be done if we consider a prediction of each tree as a *value* in the terminology of the attention mechanism, define the parametric attention weight for each tree in a specific way, and find a simple way to compute the trainable parameters of the attention weight in accordance with some objective function which is responsible for the survival model accuracy.

First, predictions of trees as values in the Nadaraya-Watson regression are SFs $S_q(t|\mathbf{x}_i)$ or CHFs $H_q(t|\mathbf{x}_i)$, $q = 1, ..., Q$. Parameters of the attention weights are proposed to define through the Huber's ϵ-contamination model. The objective function depending on the trainable parameters of the Huber's ϵ-contamination model is proposed to define by using an approximation of the RSF C-index which is maximized to get the optimal attention to trainable parameters. Moreover, the approximation of the C-index is carried out in a way which leads to the standard quadratic optimization problem.

Denote a set of leaf nodes belonging to the k-th tree as $\mathcal{Q}^{(k)} = \{q_1^{(k)}, ..., q_{s_k}^{(k)}\}$, where $q_i^{(k)}$ is the i-th leaf in the k-th tree, $k = 1, ..., Q$; s_k is the number of leaves in the k-th tree. Suppose that an example $\mathbf{x}$ falls into the i-th leaf, i.e., into leaf $q_i^{(k)}$. Let us also introduce the mean vector $\mathbf{A}_k(\mathbf{x})$ as the mean of training example vectors, which fall into the i-th leaf of the k-th tree, i.e., there holds:

$$\mathbf{A}_k(\mathbf{x}) = \frac{1}{\#\mathcal{J}_i^{(k)}} \sum_{j \in \mathcal{J}_i^{(k)}} \mathbf{x}_j, \tag{10}$$

where $\mathcal{J}_i^{(k)}$ is the index set of examples which fall into leaf $q_i^{(k)}$, and there holds $\mathcal{J}_i^{(k)} \cap \mathcal{J}_l^{(k)} = \varnothing$ for arbitrary two leaves with indices i and l in the k-th tree such that $i \neq l$; $\#\mathcal{J}_i^{(k)}$ is the number of elements in $\mathcal{J}_i^{(k)}$.

It should be noted that a single example can fall only into one leaf from $\mathcal{Q}^{(k)}$. Therefore, there is no need to use the index of the leaf in the notation for $\mathbf{A}_k(\mathbf{x})$. Mean values $\mathbf{A}_k(\mathbf{x})$ play the role of *keys* in the terminology of the attention mechanism. Indeed, every leaf node localizes a set of examples from the training set, which are close to each other. Since the tree prediction is the average of SFs or CHFs associated with examples $\mathbf{x}_j$, $j \in \mathcal{J}_i^{(k)}$, from the local set, then it makes sense to average the corresponding feature vectors. In fact, $\mathbf{A}_k(\mathbf{x})$ can be regarded as a prototype of examples localized by the leaf node. This implies that the proposed Att-RSF deals with local subsets of examples as *keys* and *values*, but not with separate examples. It is important to point out that the attention-based model can be detailed to deal with every example. However, the number of trainable parameters rapidly increases in this case and may lead to overfitting and worse results.

We denote the CHF and the SF of example $\mathbf{x}$, which falls into leaf $q_i^{(k)}$, as $H_k(t|\mathbf{x})$ and $S_k(t|\mathbf{x})$, respectively. If an example $\mathbf{x}$ (training or testing) falls into leaf $q_i^{(k)} \in \mathcal{Q}^{(k)}$ of the k-th tree, then distance $d(\mathbf{x}, \mathbf{A}_k(\mathbf{x}))$ shows how far the feature vector $\mathbf{x}$ is from the mean feature vector of all examples which fall into leaf $q_i^{(k)}$. We use the L_2-norm for the distance definition, i.e., $d(\mathbf{x}, \mathbf{A}_k(\mathbf{x})) = \|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2$. Note that each tree has only a single leaf which an example falls into.

The final RSF prediction $H(t|\mathbf{x})$ for a testing example $\mathbf{x}$ is defined as:

$$H(t|\mathbf{x}) = \frac{1}{Q} \sum_{k=1}^{Q} H_k(t|\mathbf{x}). \tag{11}$$

Let us return to the definition of the Nadaraya-Watson regression model and rewrite it in terms of RSF as follows: $H_k(t|\mathbf{x})$,

$$H(t|\mathbf{x}) = \sum_{k=1}^{Q} \alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right) \cdot H_k(t|\mathbf{x}). \tag{12}$$

Here $\alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right)$ is the attention weight which does not depend on time t and conforms with the relevance of “mean example” $\mathbf{A}_k(\mathbf{x})$ to vector $\mathbf{x}$ and satisfies condition:

$$\sum_{k=1}^{Q} \alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right) = 1, \tag{13}$$

$\mathbf{w}$ is a vector of the trainable attention parameters which will be defined below in accordance with the model modification.

The same can be written for SFs as:

$$S(t|\mathbf{x}) = \sum_{k=1}^{Q} \alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right) \cdot S_k(t|\mathbf{x}). \tag{14}$$

In terms of the attention mechanism, $H_k(t|\mathbf{x})$, $k = 1, ..., Q$, are *values*, $\mathbf{A}_k(\mathbf{x})$, $k = 1, ..., Q$, are *keys*, and $\mathbf{x}$ is the *query*. A scheme of the introduced terms is depicted in Figure 1. It illustrates a survival tree with the leaf $q_i^{(k)}$ where the vector $\mathbf{x}$ falls into.

4.2. Attention weights and the ϵ-contamination model. The next question is how to define the attention weights $\alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right)$ depending on the trainable parameters $\mathbf{w}$ to compute the parameters. Incorporating weights into the softmax function as it is shown in (9) leads to a computationally hard optimization problem. Moreover, it will be seen below that the optimization problem for computing the attention weights is constrained, and it is difficult to solve it by using the gradient-based algorithms.

Taking into account the above, we use a simple representation of attention weights proposed in [21], which leads to the linear or quadratic optimization problem whose solution is the optimal vector $\mathbf{w}$ of trainable parameters. The representation is based on applying the Huber’s ϵ-contamination model [22] which is represented as $F = (1 - \epsilon) \cdot P + \epsilon \cdot R$.

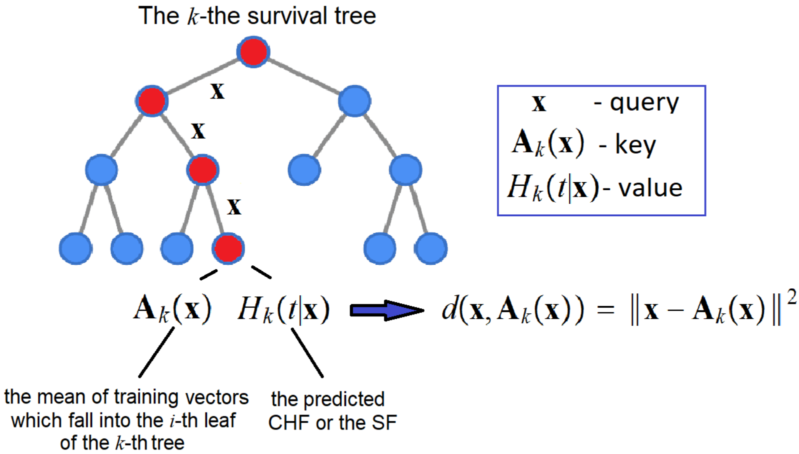
The k-the survival tree
$\mathbf{x}$
$\mathbf{x}$
$\mathbf{x}$
$\mathbf{x}$ - query
$\mathbf{A}_k(\mathbf{x})$ - key
$H_k(t|\mathbf{x})$- value
$\mathbf{A}_k(\mathbf{x})$ $H_k(t|\mathbf{x})$
$d(\mathbf{x},\mathbf{A}_k(\mathbf{x})) = \|\mathbf{x}-\mathbf{A}_k(\mathbf{x})\|^2$
the mean of training vectors which fall into the i-th leaf of the k-th tree
the predicted CHF or the SF

Fig. 1. A scheme of the introduced terms related to a survival tree and the attention mechanism, which include the query, keys and values

Here $P = (p_1, ..., p_Q)$ is a discrete probability distribution contaminated by another probability distribution denoted $R = (r_1, ..., r_Q)$, i.e., $p_1 + ... + p_Q = 1$ and $r_1 + ... + r_Q = 1$; the contamination parameter $\epsilon \in [0, 1]$ controls the degree of the contamination. It follows from the definition of the contamination model that R is a point in the unit simplex denoted as $U(1, Q)$ and having dimensionality Q. Hence, the subset of points F produced by the ϵ-contamination model is a subset of the unit simplex such that its center is the distribution P, its size is defined by hyperparameter ϵ. In particular, if $\epsilon = 1$, then the subset of points F coincides with the unit simplex, and if $\epsilon = 0$, then the subset of points F is reduced to point P.

If we assume that p_k is a result of the softmax operation, i.e., $p_k =$softmax$\left(-\|\mathbf{x}-\mathbf{A}_k(\mathbf{x})\|^2\right)$, and the probability r_k is nothing else but the trainable parameter w_k, i.e., $r_k = w_k$ for all $k = 1, ..., Q$, then the attention weight can be regarded as a result of contamination of the softmax operation, i.e., the attention weights $\alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right)$ can be represented as follows:

$$\alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right) =$$
$$= (1-\epsilon)\cdot \text{softmax}\left(-\|\mathbf{x}-\mathbf{A}_k(\mathbf{x})\|^2\right) + \epsilon \cdot w_k,\ k = 1, ..., Q. \quad (15)$$

It can be seen from the above that the softmax function depends only on the distance between $\mathbf{x}$ and $\mathbf{A}_k(\mathbf{x})$ and does not depend on trainable parameters. This implies that it can be regarded as a constant for every $\mathbf{x}$ which falls into the k-th tree. Moreover, the attention weight is linearly depends on trainable parameters $\mathbf{w}$ $=(w_1, ..., w_Q)$. The contamination parameter ϵ can be regarded as a tuning parameter and its optimal value can be selected by using the standard validation procedures. All vectors $\mathbf{w}$ satisfy condition $\mathbf{w} \cdot \mathbf{1}^{\mathrm{T}} = 1$, where $\mathbf{1}$ is the unit vector, and they form the unit simplex $U(1, Q)$. It is important to note that condition (13) is satisfied when we use the ϵ-contamination model because, according to this model, F is a probability distribution.

The property of the attention weights that the softmax operation does not depend on the trainable parameters is very important because it allows us to avoid complex computations for optimizing these parameters. These approaches are united by one idea of the linear approximation of softmax operation [34, 49, 50]. In contrast to the approximation approaches, we propose a quite different model where the softmax operation does not have trainable parameters, and the attention weights inherently depend on these parameters.

4.3. Optimization problem for computing trainable parameters. The next question is how to train the attention parameters $\mathbf{w}$ in order to compute the attention weights. In order to answer this question, we return to the C-index defined in (2) as an important measure for evaluation of the model accuracy and for comparison of different survival models. If to assume that the predicted SF of RSF depends on the trainable attention parameters $\mathbf{w}$, then the C-index should be expressed through these parameters. Then it can be maximized with respect to $\mathbf{w}$. This implies that our first aim is to write C-index as a function of $\mathbf{w}$. Let us rewrite (2) taking into account that the SF of the whole RSF is determined by the attention weights $\alpha\left(\mathbf{x}_i, \mathbf{A}_k(\mathbf{x}_i), \mathbf{w}\right)$ through the Nadaraya-Watson regression (see (14)):

$$C(\mathbf{w}) = \frac{1}{M} \sum_{i:\delta_i=1} \sum_{j:t_i<t_j} \mathbf{1}\left[S(t_i|\mathbf{x}_i, \alpha(\mathbf{x}_i)) - S(t_j|\mathbf{x}_j, \alpha(\mathbf{x}_j)) > 0\right]. \quad (16)$$

Here $\alpha(\mathbf{x}_i)$ is the short notation of the vector of the attention weights $\alpha\left(\mathbf{x}_i, \mathbf{A}_k(\mathbf{x}_i), \mathbf{w}\right)$, $k = 1, ..., Q$; $S(t_i|\mathbf{x}_i, \alpha(\mathbf{x}_i))$ is the ensemble predicted SF depending on the vector $\alpha(\mathbf{x}_i)$ of the attention weights of trees. The C-index depends on $\mathbf{w}$ through the attention weights. We use the short notation $C(\mathbf{w})$ in order to avoid the long expression for C as a function of the SFs and the attention weights.

The survival attention-based model learning means to compute optimal values of the trainable parameters $\mathbf{w}$ of the attention, which maximize the C-index $C(\mathbf{w})$ over values of non-negative weights w_q, $q = 1, ..., Q$, under constraint $\mathbf{w} \cdot \mathbf{1}^{\mathrm{T}} = 1$. In sum, we can write the following optimization problem:

$$\mathbf{w}_{opt} = \max_{\mathbf{w}} C(\mathbf{w}), \tag{17}$$

subject to $\mathbf{w} \cdot \mathbf{1}^{\mathrm{T}} = 1$ or $\mathbf{w} \in U(1, Q)$.

Figure 2 shows a scheme of the training process for computing the attention weights assigned to every tree in RSF. After training the original RSF, vectors $\mathbf{A}_k(\mathbf{x}_s)$ and functions $H_k(t|\mathbf{x}_s)$ are taken for all $k = 1, ..., Q$ and $s = 1, ..., n$. Pairs $(\mathbf{A}_k(\mathbf{x}_s), H_k(t|\mathbf{x}_s))$ allow us to write the optimization problem for maximizing the C-index $C(\mathbf{w})$ as a function of the trainable weights $\mathbf{w}$. Having optimal trainable parameters $\mathbf{w}_{opt}$, we can compute the attention weights $\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w})$ and then to find $H(t|\mathbf{x})$ by using (12).

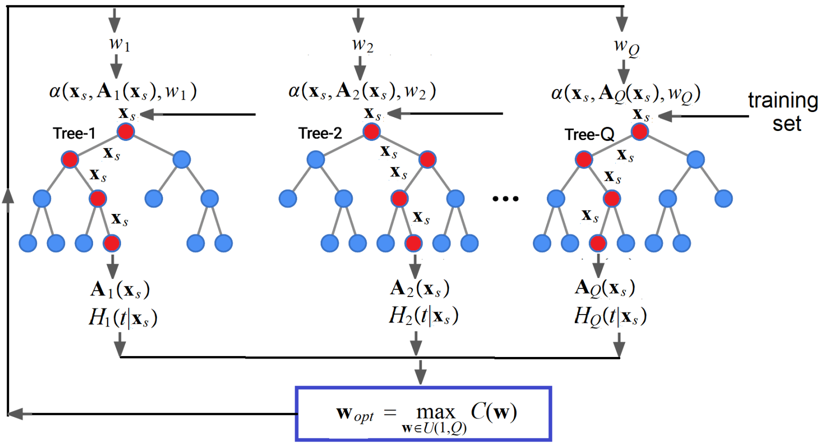
w_1
w_2
w_Q
$\alpha(\mathbf{x}_s, \mathbf{A}_1(\mathbf{x}_s), w_1)$
$\alpha(\mathbf{x}_s, \mathbf{A}_2(\mathbf{x}_s), w_2)$
$\alpha(\mathbf{x}_s, \mathbf{A}_Q(\mathbf{x}_s), w_Q)$
training set
Tree-1
Tree-2
Tree-Q
$\mathbf{x}_s$
$\mathbf{A}_1(\mathbf{x}_s)$
$H_1(t|\mathbf{x}_s)$
$\mathbf{A}_2(\mathbf{x}_s)$
$H_2(t|\mathbf{x}_s)$
$\mathbf{A}_Q(\mathbf{x}_s)$
$H_Q(t|\mathbf{x}_s)$
$\mathbf{w}_{opt} = \max_{\mathbf{w} \in U(1,Q)} C(\mathbf{w})$

Fig. 2. A scheme of training the attention weights assigned to every tree in the RSF taking an example $\mathbf{x}_s$ from the training set under the condition that the output of the k-th tree is the pair $(\mathbf{A}_k(\mathbf{x}_s), H_k(t|\mathbf{x}_s))$

It should be noted that it is difficult to solve the optimization problem (17) with the indicator functions in the objective function because it is a hard combinatorial problem. Moreover, the ensemble predictive measure is the CHF because it is the weighted sum of the tree CHFs in contrast to the SF which cannot be linearly expressed through the tree weights. However, it has been shown in [19] that a similar optimization problem can be solved by replacing SFs with CHFs and the indicator functions with hinge loss functions. Therefore, we first show that SFs can be replaced with the CHFs in the objective function. Indeed, it follows from (1) and from the monotonicity of SFs and CHFs that there holds:

$$\begin{aligned}&\mathbf{1}\left[S(t|\mathbf{x}_i) - S(t|\mathbf{x}_j) > 0\right] = \mathbf{1}\left[\ln S(t|\mathbf{x}_i) - \ln S(t|\mathbf{x}_j) > 0\right] \\ &= \mathbf{1}\left[H(t|\mathbf{x}_j) - H(t|\mathbf{x}_i) > 0\right]. \end{aligned} \tag{18}$$

Hence, the objective function (17) can be rewritten as follows:

$$C(\mathbf{w}) = \frac{1}{M}\sum_{i:\delta_i=1}\sum_{j:t_i<t_j}\mathbf{1}\left[H(t_j|\mathbf{x}_j,\alpha(\mathbf{x}_j)) - H(t_i|\mathbf{x}_i,\alpha(\mathbf{x}_i)) > 0\right]. \tag{19}$$

Now we can use (12) to express $H(t_i|\mathbf{x}_i,\alpha(\mathbf{x}_i))$ in the objective function (19) through the CHFs $H_q(t|\mathbf{x})$, $q = 1,...,Q$, obtained by every survival tree. Let us denote the set of all possible pairs (i,j) in (19), satisfying condition $\delta_i = 1$ for i and condition $t_i < t_j$ for j, as J. The objective function becomes:

$$\begin{gathered} C(\mathbf{w}) = \tfrac{1}{M}\textstyle\sum_{(i,j)\in J} \\ 1\left[\textstyle\sum_{q=1}^{Q}\left(\alpha_j^{(q)}(\mathbf{w})H_q(t|\mathbf{x}_j) - \alpha_i^{(q)}(\mathbf{w})H_q(t|\mathbf{x}_i)\right) > 0\right], \end{gathered} \tag{20}$$

where $\alpha_i^{(q)}(\mathbf{w}) = \alpha\left(\mathbf{x}_i, \mathbf{A}_q(\mathbf{x}_i), \mathbf{w}\right)$.

Let us return to the definition of $\alpha_i^{(q)}(\mathbf{w})$ by using the Huber's ϵ-contamination model as it is shown in (15). Then the inequality in (20) can be rewritten as:

$$\begin{aligned}&\left((1-\epsilon)\text{softmax}\left(-\|\mathbf{x}_j - \mathbf{A}_q(\mathbf{x}_j)\|^2\right) + \epsilon w_q\right)H_q(t|\mathbf{x}_j) \\ &- \left((1-\epsilon)\text{softmax}\left(-\|\mathbf{x}_i - \mathbf{A}_q(\mathbf{x}_i)\|^2\right) + \epsilon w_q\right)H_q(t|\mathbf{x}_i) \\ &> 0.\end{aligned}$$

Let us introduce the following notations:

$$\begin{aligned} D_q\left(t,\mathbf{x}_i,\mathbf{x}_j,\mathbf{w}\right) &= \alpha_j^{(q)}(\mathbf{w})\cdot H_q(t|\mathbf{x}_j) - \alpha_i^{(q)}(\mathbf{w})\cdot H_q(t|\mathbf{x}_i) \\ &= (1-\epsilon)\cdot F_q(t,\mathbf{x}_i,\mathbf{x}_j) + \epsilon\cdot w_q\cdot G_q(t,\mathbf{x}_i,\mathbf{x}_j), \end{aligned} \tag{21}$$

where:

$$F_q(t, \mathbf{x}_i, \mathbf{x}_j) = \text{softmax}\left(-\left\|\mathbf{x}_j - \mathbf{A}_q(\mathbf{x}_j)\right\|^2\right) \cdot H_q(t|\mathbf{x}_j) \\ - \text{softmax}\left(-\left\|\mathbf{x}_i - \mathbf{A}_q(\mathbf{x}_i)\right\|^2\right) \cdot H_q(t|\mathbf{x}_i), \qquad (22)$$

$$G_q(t, \mathbf{x}_i, \mathbf{x}_j) = H_q(t|\mathbf{x}_j) - H_q(t|\mathbf{x}_i). \qquad (23)$$

The above notations are introduced to simplify the complex expressions and to show how $D_q\left(t, \mathbf{x}_i, \mathbf{x}_j, \mathbf{w}\right)$ depends on the trainable parameters $\mathbf{w}$. Note that $F_q(\mathbf{x}_i, \mathbf{x}_j)$ and $G_q(\mathbf{x}_i, \mathbf{x}_j)$ do not depend on the trainable parameters $\mathbf{w}$ and are defined only by predictions of trees in the form of CHFs $H_q(t|\mathbf{x})$ and by examples which fall into the corresponding leaf nodes. We also do not include ϵ into $F_q(t, \mathbf{x}_i, \mathbf{x}_j)$ and $G_q(t, \mathbf{x}_i, \mathbf{x}_j)$ in order to highlight the hyperparameter in the optimization problem.

Hence, the following optimization problem can be written:

$$C(\mathbf{w}) = \max_{\mathbf{w} \in U(1,Q)} \frac{1}{M} \sum_{(i,j) \in J} \mathbf{1}\left[\sum_{q=1}^{Q} D_q\left(t, \mathbf{x}_i, \mathbf{x}_j, \mathbf{w}\right) > 0\right], \qquad (24)$$

subject to $\mathbf{w} \cdot \mathbf{1}^{\mathrm{T}} = 1$ and: $w_q \geqslant 0$, $q = 1, ..., Q$.

Problem (24) is hard to be solved. Therefore, we propose to replace the indicator function with the hinge loss function $l(x) = \max\left(0, x\right)$ similarly to the replacement proposed by Van Belle et al. [51]. This replacement is also used in the support vector machine where the hinge loss function is regarded as a desirable approximation of the indicator function.

By adding the regularization term $R(\mathbf{w})$, the optimization problem can be written as:

$$\min_{\mathbf{w} \in U(1,Q)} \left\{ \sum_{(i,j) \in J} \max\left(0, \sum_{q=1}^{Q} D_q\left(\mathbf{x}_i, \mathbf{x}_j, \mathbf{w}\right)\right) + \lambda R(\mathbf{w}) \right\}. \qquad (25)$$

Here λ is a hyper-parameter which controls the strength of the regularization. Let us introduce the variables:

$$\xi_{ij} = \max\left(0, \sum_{q=1}^{Q} D_q\left(\mathbf{x}_i, \mathbf{x}_j, \mathbf{w}\right)\right). \qquad (26)$$

If we take the regularization term in the form $R(\mathbf{w}) = \|\mathbf{w}\|^2$, then the optimization problem can be written in the following form:

$$\min_{\mathbf{w}} \left\{ \sum_{(i,j)\in J} \xi_{ij} + \lambda \|\mathbf{w}\|^2 \right\}, \tag{27}$$

subject to $\mathbf{w} \in U(1, Q)$ and:

$$\xi_{ij} \geqslant \sum_{q=1}^{Q} D_q \left(\mathbf{x}_i, \mathbf{x}_j, \mathbf{w}\right), \ \ \xi_{ij} \geqslant 0, \ \ \{i, j\} \in J. \tag{28}$$

After substituting (21) into constraints (28), we get:

$$\begin{aligned} &\xi_{ij} \geqslant \sum_{q=1}^{Q} \left((1-\epsilon)\cdot F_q(\mathbf{x}_i, \mathbf{x}_j) + \epsilon \cdot w_q \cdot G_q(\mathbf{x}_i, \mathbf{x}_j)\right), \\ &\xi_{ij} \geqslant 0, \ \ \{i, j\} \in J. \end{aligned} \tag{29}$$

We get the quadratic optimization problem with linear constraints (29) and $\mathbf{w} \in U(1, Q)$. The problem has $\#J + Q$ variables.

In spite of the superficial simplicity of the problem (27) and (29), it has a huge amount of constraints. Therefore, to simplify it, we propose its relaxation in the following way. K constraints are randomly selected from all constraints and are used in the optimization problem. Repeating random selections several times and solving the obtained optimization problems, the obtained trainable parameters $\mathbf{w}$ are averaged and the results are used to compute the attention weights.

Let us consider two important special cases when $\epsilon = 0$ and $\epsilon = 1$. In the first case ($\epsilon = 0$), the subset of training parameters is reduced to the point $F_q(t, \mathbf{x}_i, \mathbf{x}_j)$. This implies that the optimization problem should not be solved. The attention weights are not trainable and have the simple form:

$$\alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x})\right) = \text{softmax}\left(-\left\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\right\|^2\right), \ k = 1, ..., Q. \tag{30}$$

This implies that case $\epsilon = 0$ can be regarded as a special case of non-parametric attention mechanism. In the second case ($\epsilon = 1$), the subset of training parameters coincides with the unit simplex $U(1, Q)$ such that the constraints are reduced to:

$$\xi_{ij} \geqslant \sum_{q=1}^{Q} w_q \left(H_q(t|\mathbf{x}_j) - H_q(t|\mathbf{x}_i) \right), \; \xi_{ij} \geqslant 0. \tag{31}$$

The above coincides with the weighted RSF proposed in [19]. This case does not take into account the distance $\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2$, i.e., the attention weight does not depend on the feature vector $\mathbf{x}$ and is defined only by some ability of every tree averaged over the training set.

The application of the Huber's ϵ-contamination model significantly simplifies the training and testing phases of Att-RSF because we solve the standard quadratic optimization problem with the convex objective function and linear constraints instead of the complex gradient-based optimization algorithms. At the same time, the complexity of the training phase for computing the optimal trainable parameters of the attention is defined by the complexity of solving the quadratic optimization problem. It depends on the number of selected linear constraints K and on the number of repetitions of the optimization problem solving with restricted numbers of constraints. At the same time, the testing phase is very simple, and it is defined by computing the attention weights $\alpha\left(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}\right)$ in (15) under the condition that the attention parameters $\mathbf{w}$ are known. The simplicity of the testing phase allows us to get predictions in case of intensive online data traffic whereas the same cannot be realized in the training phase.

5. Numerical experiments. In order to study how Att-RSF outperforms the original RSF [10] and the weighted RSF [19], we compare Att-RSF with these models. The proposed Att-RSF as well as the original RSF and the weighted RSF are tested on the following real benchmark datasets.

The **Primary Biliary Cirrhosis (PBC) Dataset** consists of information about 418 patients with primary biliary cirrhosis of the liver from the Mayo Clinic trial [52], 257 of whom have censored data. Each patient is described by 17 features such as age, sex, ascites, hepatom, spiders, edema, bili and chol, etc. The dataset can be downloaded via the "randomForestSRC" R package.

The **German Breast Cancer Study Group 2 (GBSG2) Dataset** contains observations of 686 women [53]. Each woman is described by 10 features: age of the patients in years, menopausal status, tumor size, tumor grade, number of positive nodes, hormonal therapy, progesterone receptor, estrogen receptor, recurrence-free survival time, censoring indicator (0 - censored, 1 - event). The dataset can be obtained via the "TH.data" R package.

The **Chronic Myelogenous Leukemia Survival (CML) Dataset** is simulated according to the structure of the data by the German CML Study

Group used in [54]. The dataset consists of 507 observations with 7 features: a factor with 54 levels indicating the study center; a factor with levels trt1, trt2, trt3 indicating the treatment group; sex (0 = female, 1 = male); age in years; risk group (0 = low, 1 = medium, 2 = high); censoring status (FALSE = censored, TRUE = dead); time survival or censoring time in days. The dataset can be obtained via the “multcomp” R package (cml).

The **Bladder Cancer Dataset (BLCD)** [55] (Chapter 21) consists of observations of 86 patients after surgery assigned to placebo or chemotherapy (thiopeta). The endpoint is time to recurrence in months. Data on the number of tumors removed at surgery was also collected. The dataset is available at http://www.stat.rice.edu/~sneeley/STAT553/Datasets/survivaldata.txt.

The **Lupus Nephritis Dataset (LND)** [56] consists of observations of 87 persons with lupus nephritis. followed for 15+ years after an initial renal biopsy (the starting point of follow-up). This data set only contains time to death/censoring, indicator, duration and log(1+duration), where duration is the duration of untreated disease prior to biopsy. The dataset is available at http://www.stat.rice.edu/~sneeley/STAT553/Datasets/survivaldata.txt.

The **Heart Transplant Dataset (HTD)** consists of observations of 69 patients receiving heart transplants [57]. This dataset is available at http://lib.stat.cmu.edu/datasets/stanford.

The **Veterans’ Administration Lung Cancer Study (Veteran) Dataset** [57] consists of observations of 137 males with advanced inoperable lung cancer. The patients were randomly assigned to either a standard chemotherapy treatment or a test chemotherapy treatment. Several additional variables were also measured on the patients. The dataset can be obtained via the “survival” R package.

The **Wisconsin Prognostic Breast Cancer (WPBC) dataset** [58] contains records representing follow-up data for one breast cancer case observed by Dr. Wolberg in 1984. WPBC consists of 198 instances having 34 features. The dataset can be obtained via the “TH.data” R package.

The **Gastric Cancer Dataset (GCD)** [59] contains data on the survival of 90 patients (4 features) with locally advanced, non-resectable gastric carcinoma. The dataset can be obtained via the “coxphw” R package.

The **Microarray Breast Cancer Gene expression profiling dataset (MBC)** [60] is for predicting the clinical outcome of breast cancer. It contains 4707 expression values on 78 patients with survival information. The dataset can be obtained via the “randomForestSRC” R package.

Att-RSF is implemented by means of a software in Python. The software implementing the weighted RSF is available at https://github.com/andruekonst/weighted-random-survival-forest.

In order to evaluate the C-index for each dataset, we perform a cross-validation with 100 repetitions by taking 75% of examples from each dataset for training and 25% of examples for testing. Examples for training and testing are randomly selected in each run. Different values for hyperparameters λ and ϵ have been tested, choosing those leading to the best results. The number of survival trees in RSF is 200. The number K of constraints randomly selected from all constraints in the optimization problem for computing optimal trainable parameters $\mathbf{w}$ is 3000, and the number of solutions to the optimization problem with different subsets of constraints is 10.

Table 1 illustrates the C-indices of the original RSF, the weighted RSF (WRSF), and the Att-RSF model obtained for the above datasets under the condition the depth of survival trees in the models is equal to 2. It can be seen from Table 1 that Att-RSF outperforms RSF as well as WRSF for all considered datasets.

Table 1. Comparison of the C-index obtained for RSF, WRSF and Att-RSF by using different datasets

Dataset	RSF	WRSF	Att-RSF
PBC	0.878	0.931	0.943
GBSG2	0.879	0.928	0.959
BLCD	0.876	0.926	0.987
CML	0.869	0.923	0.979
LND	0.875	0.924	0.940
HTD	0.859	0.931	0.931
Veteran	0.870	0.929	0.970
WPBC	0.914	0.943	0.969
GCD	0.518	0.524	0.692
MBC	0.802	0.868	0.914

To formally show the outperformance of the proposed Att-RSF model, we apply the t-test which has been proposed and described by Demsar [61] for testing whether the average difference in the performance of two models, Att-RSF and WRSF, is significantly different from zero. Since we use differences between accuracy measures of Att-RSF and WRSF, then they are compared with 0. The t-statistics is distributed in accordance with the Student distribution with $10 - 1$ degrees of freedom. The obtained p-value and the 95% confidence interval for the mean 0.046 are $p = 0.0135$ and $[0.012, 0.079]$, respectively. One can see from the t-test that Att-RSF clearly outperforms WRSF due to condition $p < 0.05$. Better results can be carried out for models Att-RSF and RSF. We get the p-value and the 95% confidence interval for the mean 0.094, which are $p = 1.4 \cdot 10^{-5}$ and $[0.069, 0.120]$, respectively. The second test also demonstrates the outperformance of Att-RSF in comparison with RSF.

The next interesting question is how the C-index depends on the contamination hyperparameter ϵ for different datasets. The C-index as a function of the contamination hyperparameter for the CML dataset is depicted in Figure 3 by the solid line with the circle markers. For comparison purposes, lines with the triangle and square markers correspond to RSF and WRSF, respectively. It can be seen from Figure 3 that Att-RSF provides the best results (the largest C-index) when $\epsilon = 0.75$. However, the optimal choice of the contamination hyperparameter depends on a considered dataset. For example, Figure 4 illustrates the case when the optimal contamination hyperparameter is equal to 1 for the HTD dataset. This implies that WRSF outperforms Att-RSF for all ϵ and has the same C-index for $\epsilon = 1$.

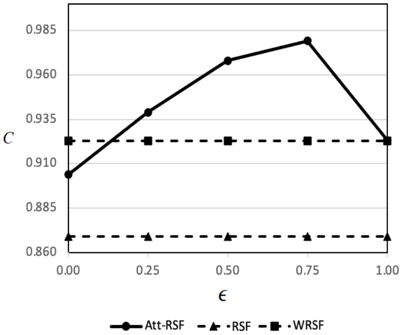

Fig. 3. The dependence of the C-index on the contamination hyperparameter for the CML dataset

6. Conclusion. A new RSF model based on using the attention mechanism has been presented in the paper. The first main idea behind this model is to adapt the Nadaraya-Watson kernel regression to RSF. The second main idea is to apply the Huber's ϵ-contamination model in order to represent the attention weights as the linear function of the trainable attention parameters. Att-RSF has demonstrated outperforming results in comparison with RSF and WRSF for the most considered datasets. This implies that Att-RSF can be an accurate tool for survival analysis especially when tabular data are used for training.

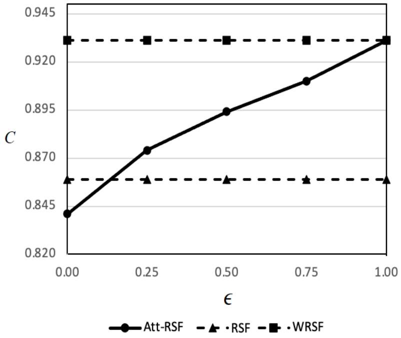

Fig. 4. The dependence of the C-index on the contamination hyperparameter for the HTD dataset

We have considered only one type of the trainable parameters, namely, parameters produced by the Huber's ϵ-contamination model. However, many other types of parameters can be proposed, for example, parameters of the softmax operation. It should be noted that additional parameters lead to more complex optimization problems for computing the attention weights which cannot be solved in a simple way. However, they can be solved by using the gradient-based algorithms and can provide more accurate predictions. The study of other attention mechanisms is an interesting direction for further research.

Another interesting direction for further research is to consider measures of the model accuracy different from the used C-index, for example, the Brier score. The use of other measures may lead to better results.

It should be pointed out that the softmax operation in the attention mechanism defined by the Gaussian kernel in the Nadaraya-Watson kernel regression can be also replaced with other operations if considering different kernels. This consideration and the analysis of the kernel types can be also regarded as an interesting direction for further research.

References

1. Hosmer D., Lemeshow S., May S. Applied Survival Analysis: Regression Modeling of Time to Event Data. — New Jersey : John Wiley & Sons, 2008.
2. DeepSurv: Personalized treatment recommender system using a Cox proportional hazards deep neural network / Katzman J., Shaham U., Cloninger A., Bates J., Jiang T., and Kluger Y. // BMC medical research methodology. — 2018. — Vol. 18, no. 24. — P. 1–12.
3. A Deep Active Survival Analysis Approach for Precision Treatment Recommendations: Application of Prostate Cancer / Nezhad M., Sadati N., Yang K., and Zhu D. — 2018. —

Apr. — arXiv:1804.03280v1.

4. Wang P., Li Y., Reddy C. Machine Learning for Survival Analysis: A Survey // ACM Computing Surveys (CSUR). — 2019. — Vol. 51, no. 6. — P. 1–36.
5. Zhao L., Feng D. DNNSurv: Deep Neural Networks for Survival Analysis Using Pseudo Values. — 2020. — Mar. — arXiv:1908.02337v2.
6. Cox D. Regression models and life-tables // Journal of the Royal Statistical Society, Series B (Methodological). — 1972. — Vol. 34, no. 2. — P. 187–220.
7. Tibshirani R. The lasso method for variable selection in the Cox model // Statistics in medicine. — 1997. — Vol. 16, no. 4. — P. 385–395.
8. Survival SVM: a practical scalable algorithm. / Belle V. V., Pelckmans K., Suykens J., and Huffel S. V. // ESANN. — 2008. — P. 89–94.
9. Bou-Hamad I., Larocque D., Ben-Ameur H. A review of survival trees // Statistics Surveys. — 2011. — Vol. 5. — P. 44–71.
10. Ishwaran H., Kogalur U. Random Survival Forests for R // R News. — 2007. — Vol. 7, no. 2. — P. 25–31.
11. Breiman L. Random forests // Machine learning. — 2001. — Vol. 45, no. 1. — P. 5–32.
12. Hu C., Steingrimsson J. Personalized Risk Prediction in Clinical Oncology Research: Applications and Practical Issues Using Survival Trees and Random Forests // Journal of Biopharmaceutical Statistics. — 2018. — Vol. 28, no. 2. — P. 333–349.
13. Relative Risk Forests for Exercise Heart Rate Recovery as a Predictor of Mortality / Ishwaran H., Blackstone E., Pothier C., and Lauer M. // Journal of the American Statistical Association. — 2004. — Vol. 99. — P. 591–600.
14. Mogensen U., Ishwaran H., Gerds T. Evaluating Random Forests for Survival Analysis using Prediction Error Curves // Journal of Statistical Software. — 2012. — Vol. 50, no. 11. — P. 1–23.
15. Random survival forests for dynamic predictions of a time-to-event outcome using a longitudinal biomarker / Pickett K., Suresh K., Campbell K., Davis S., and Juarez-Colunga E. // BMC Medical Research Methodology. — 2021. — Vol. 21, no. 1. — P. 1–14.
16. Schmid M., Wright M., Ziegler A. On the use of Harrell’s C for clinical risk prediction via random survival forests // Expert Systems with Applications. — 2016. — Vol. 63. — P. 450–459.
17. Wright M., Dankowski T., Ziegler A. Unbiased split variable selection for random survival forests using maximally selected rank statistics // Statistics in Medicine. — 2017. — Vol. 36, no. 8. — P. 1272–1284.
18. Zhou L., Wang H., Xu Q. Survival forest with partial least squares for high dimensional censored data // Chemometrics and Intelligent Laboratory Systems. — 2018. — Vol. 179. — P. 12–21.
19. A weighted random survival forest / Utkin L., Konstantinov A., Chukanov V., Kots M., Ryabinin M., and Meldo A. // Knowledge-Based Systems. — 2019. — Vol. 177. — P. 136–144.
20. Evaluating the yield of medical tests / Harrell F., Califf R., Pryor D., Lee K., and Rosati R. // Journal of the American Medical Association. — 1982. — Vol. 247. — P. 2543–2546.
21. Utkin L., Konstantinov A. Attention-based Random Forest and Contamination Model // Neural Networks. — 2022. — Vol. 154. — P. 346–359.
22. Huber P. Robust Statistics. — New York : Wiley, 1981.
23. Witten D., Tibshirani R. Survival analysis with high-dimensional covariates // Statistical Methods in Medical Research. — 2010. — Vol. 19, no. 1. — P. 29–51.
24. Zhang H., Lu W. Adaptive Lasso for Cox’s proportional hazards model // Biometrika. — 2007. — Vol. 94, no. 3. — P. 691–703.

25. Support vector methods for survival analysis: a comparison between ranking and regression approaches / Belle V. V., Pelckmans K., Huffel S. V., and Suykens J. // Artificial intelligence in medicine. — 2011. — Vol. 53, no. 2. — P. 107–118.

26. Zhu X., Yao J., Huang J. Deep convolutional neural network for survival analysis with pathological images // 2016 IEEE International Conference on Bioinformatics and Biomedicine. — IEEE. — 2016. — P. 544–547.

27. Image-based Survival Analysis for Lung Cancer Patients using CNNs / Haarburger C., Weitz P., Rippel O., and Merhof D. — 2018. — Aug. — arXiv:1808.09679v1.

28. Decision tree for competing risks survival probability in breast cancer study / Ibrahim N., Kudus A., Daud I., and Bakar M. A. // International Journal Of Biological and Medical Research. — 2008. — Vol. 3, no. 1. — P. 25–29.

29. Wang H., Zhou L. Random survival forest with space extensions for censored data // Artificial intelligence in medicine. — 2017. — Vol. 79. — P. 52–61.

30. An attentive survey of attention models / Chaudhari S., Mithal V., Polatkan G., and Ramanath R. — 2019. — Apr. — arXiv:1904.02874.

31. Correia A., Colombini E. Attention, please! A survey of neural attention models in deep learning. — 2021. — Mar. — arXiv:2103.16775.

32. Correia A., Colombini E. Neural Attention Models in Deep Learning: Survey and Taxonomy. — 2021. — Dec. — arXiv:2112.05909.

33. A Survey of Transformers / Lin T., Wang Y., Liu X., and Qiu X. — 2021. — Jul. — arXiv:2106.04554.

34. Random Features for Kernel Approximation: A Survey on Algorithms, Theory, and Beyond / Liu F., Huang X., Chen Y., and Suykens J. — 2021. — Jul. — arXiv:2004.11154v5.

35. Niu Z., Zhong G., Yu H. A review on the attention mechanism of deep learning // Neurocomputing. — 2021. — Vol. 452. — P. 48–62.

36. Ronao C., Cho S.-B. Random Forests with Weighted Voting for Anomalous Query Access Detection in Relational Databases // Artificial Intelligence and Soft Computing. ICAISC 2015. — Cham : Springer. — 2015. — Vol. 9120 of Lecture Notes in Computer Science. — P. 36–48.

37. Xuan S., Liu G., Li Z. Refined Weighted Random Forest and Its Application to Credit Card Fraud Detection // Computational Data and Social Networks. — Cham : Springer International Publishing. — 2018. — P. 343–355.

38. Zhang X., Wang M. Weighted Random Forest Algorithm Based on Bayesian Algorithm // Journal of Physics: Conference Series. — IOP Publishing. — 2021. — Vol. 1924. — P. 1–6.

39. Weighted vote for trees aggregation in Random Forest / Daho M., Settouti N., Lazouni M., and Chikh M. // 2014 International Conference on Multimedia Computing and Systems (ICMCS). — IEEE. — 2014. — April. — P. 438–443.

40. Utkin L., Kovalev M., Meldo A. A deep forest classifier with weights of class probability distribution subsets // Knowledge-Based Systems. — 2019. — Vol. 173. — P. 15–27.

41. Utkin L., Kovalev M., Coolen F. Imprecise weighted extensions of random forests for classification and regression // Applied Soft Computing. — 2020. — Vol. 92, no. Article 106324. — P. 1–14.

42. Development and validation of a prognostic model for survival time data: application to prognosis of HIV positive patients treated with antiretroviral therapy / May M., Royston P., Egger M., Justice A., and Sterne J. // Statistics in Medicine. — 2004. — Vol. 23. — P. 2375–2398.

43. Random Survival Forests / Ishwaran H., Kogalur U., Blackstone E., and Lauer M. // Annals of Applied Statistics. — 2008. — Vol. 2. — P. 841–860.

44. Nadaraya E. On estimating regression // Theory of Probability & Its Applications. — 1964. — Vol. 9, no. 1. — P. 141–142.

45. Watson G. Smooth regression analysis // Sankhya: The Indian Journal of Statistics, Series A. — 1964. — P. 359–372.

46. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate. — 2014. — Sep. — arXiv:1409.0473.

47. Luong T., Pham H., Manning C. Effective approaches to attention-based neural machine translation // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. — The Association for Computational Linguistics. — 2015. — P. 1412–1421.

48. Attention is all you need / Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., and Polosukhin I. // Advances in Neural Information Processing Systems. — 2017. — P. 5998–6008.

49. Rethinking Attention with Performers / Choromanski K., Likhosherstov V., Dohan D., Song X., Gane A., Sarlos T., Hawkins P., Davis J., Mohiuddin A., Kaiser L., Belanger D., Colwell L., and Weller A. // 2021 International Conference on Learning Representations. — 2021.

50. Schlag I., Irie K., Schmidhuber J. Linear transformers are secretly fast weight programmers // International Conference on Machine Learning 2021. — PMLR. — 2021. — P. 9355–9366.

51. Support vector machines for survival analysis / Belle V. V., Pelckmans K., Suykens J., and Huffel S. V. // Proceedings of the Third International Conference on Computational Intelligence in Medicine and Healthcare (CIMED2007). — 2007. — P. 1–8.

52. Fleming T., Harrington D. Counting processes and survival aalysis. — Hoboken, NJ, USA : John Wiley & Sons, 1991.

53. Sauerbrei W., Royston P. Building multivariable prognostic and diagnostic models: transformation of the predictors by using fractional polynomials // Journal of the Royal Statistics Society Series A. — 1999. — Vol. 162, no. 1. — P. 71–94.

54. Randomized comparison of interferon-alpha with busulfan and hydroxyurea in chronic myelogenous leukemia. The German CML study group / Hehlmann R., Heimpel H., Hasford J., Kolb H., Pralle H., Hossfeld D., Queisser W., Loeffler H., Hochhaus A., and Heinze B. // Blood. — 1994. — Vol. 84, no. 12. — P. 4064–4077.

55. Pagano M., Gauvreau K. Principles of biostatistics. — Pacific Grove, CA : Duxbury, 2000.

56. Abrahamowicz M., MacKenzie T., Esdaile J. Time-dependent hazard ratio: modelling and hypothesis testing with application in lupus nephritis // JASA. — 1996. — Vol. 91. — P. 1432–1439.

57. Kalbfleisch J., Prentice R. The Statistical Analysis of Failure Time Data. — New York : John Wiley and Sons, 1980.

58. Street W., Mangasarian O., Wolberg W. An inductive learning approach to prognostic prediction // Proceedings of the Twelfth International Conference on Machine Learning. — San Francisco : Morgan Kaufmann. — 1995. — P. 522–530.

59. Stablein D., Carter J., Novak J. Analysis of Survival Data with Nonproportional Hazard Functions // Controlled Clinical Trials. — 1981. — Vol. 2. — P. 149–159.

60. Gene expression profiling predicts clinical outcome of breast cancer / Veer L. V., Dai H., Vijver M. V. D., He Y., Hart A., Mao M., Peterse H., Kooy K. V. D., Marton M., Witteveen A., and Schreiber G. // Nature. — 2002. — Vol. 12. — P. 530–536.

61. Demsar J. Statistical comparisons of classifiers over multiple data sets // Journal of Machine Learning Research. — 2006. — Vol. 7. — P. 1–30.

Utkin Lev — Ph.D., Dr.Sci., Professor, Head of the institute, Institute of computer science and technology, Peter the Great St.Petersburg Polytechnic University. Research interests: machine learning, imprecise probability theory, decision making. The number of publications — 300. lev.utkin@gmail.com; 29, Politekhnicheskaya St., 195251, St. Petersburg, Russia; office phone: +7(812)775-0530.

Konstantinov Andrei — Ph.D., Graduate student, Peter the Great St. Petersburg Polytechnic University. Research interests: machine learning, computer vision and image processing. The number of publications — 15. andrue.konst@gmail.com; 29, Politekhnicheskaya St., 195251, St. Petersburg, Russia; office phone: +7(911)954-5565.

Acknowledgements. This work is supported by the Russian Science Foundation under grant 21-11-00116.

УДК 004.8 **DOI** 10.15622/ia.21.5.1

Л.В. Уткин, А.В. Константинов,

СЛУЧАЙНЫЙ ЛЕС ВЫЖИВАЕМОСТИ И РЕГРЕССИЯ НАДАРАЯ-УОТСОНА

Уткин Л.В., Константинов А.В. **Случайный лес выживаемости и регрессия Надарая-Уотсона.**

Аннотация. В статье представлен случайный лес выживаемости на основе модели внимания (Att-RSF). Первая идея, лежащая в основе леса, состоит в том, чтобы адаптировать ядерную регрессию Надарая-Уотсона к случайному лесу выживаемости таким образом, чтобы веса регрессии или ядра можно было рассматривать как обучаемые веса внимания при важном условии, что предсказания случайного леса выживаемости представлены в виде функций времени, например, функции выживания или кумулятивной функции риска. Каждый обучаемый вес, присвоенный дереву и примеру из обучающей или тестовой выборки, определяется двумя факторами: способностью соответствующего дерева предсказывать и особенностью примера, попадающего в лист дерева. Вторая идея Att-RSF состоит в том, чтобы применить модель загрязнения Хьюбера для представления весов внимания как линейной функции обучаемых параметров внимания. C-индекс Харрелла (индекс конкордации) как показатель качества предсказания случайного леса выживаемости используется при формировании функции потерь для обучения весов внимания. Использование C-индекса вместе с моделью загрязнения приводит к стандартной задаче квадратичной оптимизации для вычисления весов, которая имеет целый ряд простых алгоритмов решения. Численные эксперименты с реальными наборами данных, содержащими данные о выживаемости, иллюстрируют предлагаемую модель Att-RSF.

Ключевые слова: машинное обучение, случайный лес выживаемости, функция выживаемости, C-индекс, кумулятивная функция риска, модель внимания, модель засорения Хьюбера.

Литература

1. Hosmer D., Lemeshow S., May S. Applied Survival Analysis: Regression Modeling of Time to Event Data. — New Jersey : John Wiley & Sons, 2008.
2. DeepSurv: Personalized treatment recommender system using a Cox proportional hazards deep neural network / Katzman J., Shaham U., Cloninger A., Bates J., Jiang T., and Kluger Y. // BMC medical research methodology. — 2018. — Vol. 18, no. 24. — P. 1–12.
3. A Deep Active Survival Analysis Approach for Precision Treatment Recommendations: Application of Prostate Cancer / Nezhad M., Sadati N., Yang K., and Zhu D. — 2018. — Apr. — arXiv:1804.03280v1.
4. Wang P., Li Y., Reddy C. Machine Learning for Survival Analysis: A Survey // ACM Computing Surveys (CSUR). — 2019. — Vol. 51, no. 6. — P. 1–36.
5. Zhao L., Feng D. DNNSurv: Deep Neural Networks for Survival Analysis Using Pseudo Values. — 2020. — Mar. — arXiv:1908.02337v2.
6. Cox D. Regression models and life-tables // Journal of the Royal Statistical Society, Series B (Methodological). — 1972. — Vol. 34, no. 2. — P. 187–220.
7. Tibshirani R. The lasso method for variable selection in the Cox model // Statistics in medicine. — 1997. — Vol. 16, no. 4. — P. 385–395.

8. Survival SVM: a practical scalable algorithm. / Belle V. V., Pelckmans K., Suykens J., and Huffel S. V. // ESANN. — 2008. — P. 89–94.

9. Bou-Hamad I., Larocque D., Ben-Ameur H. A review of survival trees // Statistics Surveys. — 2011. — Vol. 5. — P. 44–71.

10. Ishwaran H., Kogalur U. Random Survival Forests for R // R News. — 2007. — Vol. 7, no. 2. — P. 25–31.

11. Breiman L. Random forests // Machine learning. — 2001. — Vol. 45, no. 1. — P. 5–32.

12. Hu C., Steingrimsson J. Personalized Risk Prediction in Clinical Oncology Research: Applications and Practical Issues Using Survival Trees and Random Forests // Journal of Biopharmaceutical Statistics. — 2018. — Vol. 28, no. 2. — P. 333–349.

13. Relative Risk Forests for Exercise Heart Rate Recovery as a Predictor of Mortality / Ishwaran H., Blackstone E., Pothier C., and Lauer M. // Journal of the American Statistical Association. — 2004. — Vol. 99. — P. 591–600.

14. Mogensen U., Ishwaran H., Gerds T. Evaluating Random Forests for Survival Analysis using Prediction Error Curves // Journal of Statistical Software. — 2012. — Vol. 50, no. 11. — P. 1–23.

15. Random survival forests for dynamic predictions of a time-to-event outcome using a longitudinal biomarker / Pickett K., Suresh K., Campbell K., Davis S., and Juarez-Colunga E. // BMC Medical Research Methodology. — 2021. — Vol. 21, no. 1. — P. 1–14.

16. Schmid M., Wright M., Ziegler A. On the use of Harrell's C for clinical risk prediction via random survival forests // Expert Systems with Applications. — 2016. — Vol. 63. — P. 450–459.

17. Wright M., Dankowski T., Ziegler A. Unbiased split variable selection for random survival forests using maximally selected rank statistics // Statistics in Medicine. — 2017. — Vol. 36, no. 8. — P. 1272–1284.

18. Zhou L., Wang H., Xu Q. Survival forest with partial least squares for high dimensional censored data // Chemometrics and Intelligent Laboratory Systems. — 2018. — Vol. 179. — P. 12–21.

19. A weighted random survival forest / Utkin L., Konstantinov A., Chukanov V., Kots M., Ryabinin M., and Meldo A. // Knowledge-Based Systems. — 2019. — Vol. 177. — P. 136–144.

20. Evaluating the yield of medical tests / Harrell F., Califf R., Pryor D., Lee K., and Rosati R. // Journal of the American Medical Association. — 1982. — Vol. 247. — P. 2543–2546.

21. Utkin L., Konstantinov A. Attention-based Random Forest and Contamination Model // Neural Networks. — 2022. — Vol. 154. — P. 346–359.

22. Huber P. Robust Statistics. — New York : Wiley, 1981.

23. Witten D., Tibshirani R. Survival analysis with high-dimensional covariates // Statistical Methods in Medical Research. — 2010. — Vol. 19, no. 1. — P. 29–51.

24. Zhang H., Lu W. Adaptive Lasso for Cox's proportional hazards model // Biometrika. — 2007. — Vol. 94, no. 3. — P. 691–703.

25. Support vector methods for survival analysis: a comparison between ranking and regression approaches / Belle V. V., Pelckmans K., Huffel S. V., and Suykens J. // Artificial intelligence in medicine. — 2011. — Vol. 53, no. 2. — P. 107–118.

26. Zhu X., Yao J., Huang J. Deep convolutional neural network for survival analysis with pathological images // 2016 IEEE International Conference on Bioinformatics and Biomedicine. — IEEE. — 2016. — P. 544–547.

27. Image-based Survival Analysis for Lung Cancer Patients using CNNs / Haarburger C., Weitz P., Rippel O., and Merhof D. — 2018. — Aug. — arXiv:1808.09679v1.

28. Decision tree for competing risks survival probability in breast cancer study / Ibrahim N., Kudus A., Daud I., and Bakar M. A. // International Journal Of Biological and Medical Research. — 2008. — Vol. 3, no. 1. — P. 25–29.

29. Wang H., Zhou L. Random survival forest with space extensions for censored data // Artificial intelligence in medicine. — 2017. — Vol. 79. — P. 52–61.

30. An attentive survey of attention models / Chaudhari S., Mithal V., Polatkan G., and Ramanath R. — 2019. — Apr. — arXiv:1904.02874.

31. Correia A., Colombini E. Attention, please! A survey of neural attention models in deep learning. — 2021. — Mar. — arXiv:2103.16775.

32. Correia A., Colombini E. Neural Attention Models in Deep Learning: Survey and Taxonomy. — 2021. — Dec. — arXiv:2112.05909.

33. A Survey of Transformers / Lin T., Wang Y., Liu X., and Qiu X. — 2021. — Jul. — arXiv:2106.04554.

34. Random Features for Kernel Approximation: A Survey on Algorithms, Theory, and Beyond / Liu F., Huang X., Chen Y., and Suykens J. — 2021. — Jul. — arXiv:2004.11154v5.

35. Niu Z., Zhong G., Yu H. A review on the attention mechanism of deep learning // Neurocomputing. — 2021. — Vol. 452. — P. 48–62.

36. Ronao C., Cho S.-B. Random Forests with Weighted Voting for Anomalous Query Access Detection in Relational Databases // Artificial Intelligence and Soft Computing. ICAISC 2015. — Cham : Springer. — 2015. — Vol. 9120 of Lecture Notes in Computer Science. — P. 36–48.

37. Xuan S., Liu G., Li Z. Refined Weighted Random Forest and Its Application to Credit Card Fraud Detection // Computational Data and Social Networks. — Cham : Springer International Publishing. — 2018. — P. 343–355.

38. Zhang X., Wang M. Weighted Random Forest Algorithm Based on Bayesian Algorithm // Journal of Physics: Conference Series. — IOP Publishing. — 2021. — Vol. 1924. — P. 1–6.

39. Weighted vote for trees aggregation in Random Forest / Daho M., Settouti N., Lazouni M., and Chikh M. // 2014 International Conference on Multimedia Computing and Systems (ICMCS). — IEEE. — 2014. — April. — P. 438–443.

40. Utkin L., Kovalev M., Meldo A. A deep forest classifier with weights of class probability distribution subsets // Knowledge-Based Systems. — 2019. — Vol. 173. — P. 15–27.

41. Utkin L., Kovalev M., Coolen F. Imprecise weighted extensions of random forests for classification and regression // Applied Soft Computing. — 2020. — Vol. 92, no. Article 106324. — P. 1–14.

42. Development and validation of a prognostic model for survival time data: application to prognosis of HIV positive patients treated with antiretroviral therapy / May M., Royston P., Egger M., Justice A., and Sterne J. // Statistics in Medicine. — 2004. — Vol. 23. — P. 2375–2398.

43. Random Survival Forests / Ishwaran H., Kogalur U., Blackstone E., and Lauer M. // Annals of Applied Statistics. — 2008. — Vol. 2. — P. 841–860.

44. Nadaraya E. On estimating regression // Theory of Probability & Its Applications. — 1964. — Vol. 9, no. 1. — P. 141–142.

45. Watson G. Smooth regression analysis // Sankhya: The Indian Journal of Statistics, Series A. — 1964. — P. 359–372.

46. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate. — 2014. — Sep. — arXiv:1409.0473.

47. Luong T., Pham H., Manning C. Effective approaches to attention-based neural machine translation // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. — The Association for Computational Linguistics. — 2015. — P. 1412–1421.

48. Attention is all you need / Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., and Polosukhin I. // Advances in Neural Information Processing Systems. — 2017. — P. 5998–6008.

49. Rethinking Attention with Performers / Choromanski K., Likhosherstov V., Dohan D., Song X., Gane A., Sarlos T., Hawkins P., Davis J., Mohiuddin A., Kaiser L., Belanger D., Colwell L., and Weller A. // 2021 International Conference on Learning Representations. — 2021.

50. Schlag I., Irie K., Schmidhuber J. Linear transformers are secretly fast weight programmers // International Conference on Machine Learning 2021. — PMLR. — 2021. — P. 9355–9366.

51. Support vector machines for survival analysis / Belle V. V., Pelckmans K., Suykens J., and Huffel S. V. // Proceedings of the Third International Conference on Computational Intelligence in Medicine and Healthcare (CIMED2007). — 2007. — P. 1–8.

52. Fleming T., Harrington D. Counting processes and survival aalysis. — Hoboken, NJ, USA : John Wiley & Sons, 1991.

53. Sauerbrei W., Royston P. Building multivariable prognostic and diagnostic models: transformation of the predictors by using fractional polynomials // Journal of the Royal Statistics Society Series A. — 1999. — Vol. 162, no. 1. — P. 71–94.

54. Randomized comparison of interferon-alpha with busulfan and hydroxyurea in chronic myelogenous leukemia. The German CML study group / Hehlmann R., Heimpel H., Hasford J., Kolb H., Pralle H., Hossfeld D., Queisser W., Loeffler H., Hochhaus A., and Heinze B. // Blood. — 1994. — Vol. 84, no. 12. — P. 4064–4077.

55. Pagano M., Gauvreau K. Principles of biostatistics. — Pacific Grove, CA : Duxbury, 2000.

56. Abrahamowicz M., MacKenzie T., Esdaile J. Time-dependent hazard ratio: modelling and hypothesis testing with application in lupus nephritis // JASA. — 1996. — Vol. 91. — P. 1432–1439.

57. Kalbfleisch J., Prentice R. The Statistical Analysis of Failure Time Data. — New York : John Wiley and Sons, 1980.

58. Street W., Mangasarian O., Wolberg W. An inductive learning approach to prognostic prediction // Proceedings of the Twelfth International Conference on Machine Learning. — San Francisco : Morgan Kaufmann. — 1995. — P. 522–530.

59. Stablein D., Carter J., Novak J. Analysis of Survival Data with Nonproportional Hazard Functions // Controlled Clinical Trials. — 1981. — Vol. 2. — P. 149–159.

60. Gene expression profiling predicts clinical outcome of breast cancer / Veer L. V., Dai H., Vijver M. V. D., He Y., Hart A., Mao M., Peterse H., Kooy K. V. D., Marton M., Witteveen A., and Schreiber G. // Nature. — 2002. — Vol. 12. — P. 530–536.

61. Demsar J. Statistical comparisons of classifiers over multiple data sets // Journal of Machine Learning Research. — 2006. — Vol. 7. — P. 1–30.

Уткин Лев Владимирович — д-р техн. наук, профессор, директор, институт компьютерных наук и технологий, Санкт-Петербургский политехнический университет Петра Великого. Область научных интересов: машинное обучение, неточная теория вероятностей, принятие решений. Число научных публикаций — 300. lev.utkin@gmail.com; улица Политехническая, 29, 195251, Санкт-Петербург, Россия; р.т.: +7(812)775-0530.

Константинов Андрей Владимирович — канд. техн. наук, аспирант, Санкт-Петербургский политехнический университет Петра Великого. Область научных интересов: машинное обучение, компьютерное зрение и обработка изображений. Число научных публикаций — 15. andrue.konst@gmail.com; Политехническая улица, 29, 195251, Санкт-Петербург, Россия; р.т.: +7(911)954-5565.

Поддержка исследований. Работа выполнена при поддержке Российского научного фонда в рамках гранта 21-11-00116.