A Random Forest Model Building Using A priori Information for Diagnosis

Sergey Subbotin[1][0000-0001-5814-8268]

[1]Zaporizhzhia National Technical University, Zhukovsky str., 64,Zaporizhzhia, 69063, Ukraine
subbotin@zntu.edu.ua

Abstract. The problem of inductive model building on precedents for biomedical applications is considered. The model paradigm is a random forest as a set of decision tree classifiers working as ensemble. The apriori information taken from training data set is used in proposed method of random forest model building provide more accurate model saving general random character of a method. The resulting random forest provide more accurate model in comparison with a single decision tree, but its comparison with known methods of random forest model building proposed method is more accurate.

Keywords: medical diagnosis, decision tree, random forest, training, inductive learning

1 Introduction

The decision trees [1-2], which are hierarchical tree-like models for obtaining a decision on assigning a recognized instance to one of the possible classes, are useful for problems of biomedical diagnosis by precedents. However, the known methods of constructing models based on decision trees [1-24] are not always able to provide the required level of classification accuracy. The paradigm of a random forest [25-31] is used to improve the accuracy of models based on decision trees.

Random forest paradigm [25-31] is committee (ensemble) of decision trees and combines two main ideas: the Bagging method of L. Breiman [32, 33] and the method of random subspaces of T. Ho [26]. Generally, the idea of the method is to use a large ensemble of decision trees, each of which in itself gives not a lot of classification quality, but together joined they provide a good result due to the large number of predictors [25]. Decision trees are a good family of basic classifiers for bagging [32], since they are quite complex and can reach zero error on any sample. The method of random subspaces [26] allows to reduce the correlation between trees and avoid retraining. Basic methods are trained on different subsets of the features, which are also randomly distinguished.

The random forest paradigm has such advantages [25-31] as: the ability to efficiently process data with a large number of features and classes, insensitivity to scaling (and in general to any monotonic transformations) of feature values, resistance to the inclusion of irrelevant (useless) features, the ability to process both continuous and discrete features, the ability to work with missing data, possibility to be used for feature significance estimation, insensitivity to emissions in data due to random sampling, possibility of evaluating the resulting model ability to generalize (test for out-of-bag - unselected instances), high parallelizability and scalability, high prediction accuracy, the ability to balance the weight of each class on the entire sample, or on a subsample of each tree, low tendency to retraining (in practice trees almost always only improves composition, but upon validation, after reaching a certain number of trees, the learning curve goes asymptote), the random forests by modifying their definitions can be represented as nuclear methods, which are more convenient and interpretable for analysis, random forest can be converted into the *k*-nearest neighbors model.

However, a random forest has such disadvantages as [25-31]: not very high accuracy of models, lower interpretability of the model compared to a single tree, lack of formal conclusions (*p*-values) available for assessing the feature importance, poor quality of work for samples containing a large number of sparse features, a tendency to retraining on noisy data, inability to extrapolate, model bias for data including categorical variables with different amount levels, in favour of features with a large number of levels (when a feature has many levels, the tree will be more adaptable to these features, since they can get a higher value of the optimized functional such as information growth), tendency to give preference to small groups of correlated features that have similar significance for tags to large groups, a large size of the resulting models - the spatial complexity is estimated as $O(ST)$ for storing the model, where S is the volume of the initial sample, T is the number of trees, the large time spent on building the model compared to single decision tree.

The aim of this paper is to improve random forest building method preserving its random character, but concentrating the method on such solutions, which seems to be more convenient to increase model accuracy using the a priori information extracted from the training sample.

2 Formal problem statement

Let we have a training set of S precedents (observations, instances, cases) $<x, y>$, where $x = \{x^s\}$, $y=\{y^s\}$, $s = 1, 2, ..., S$, x^s is an s-th instance of the sample, y^s is an output feature value associated with the s-th instance, $x^s = \{x^s{}_j\}$, $j = 1, 2, ..., N$, $x^s{}_j$ is a value of the j-th input feature of the s-th instance, N is a number of input features.

Then the general model constructing task for the dependence $y=f(w, x)$ is to find such a model structure f and such values of model parameters w for which the model quality criterion F is satisfied. As a rule, for the problems of approximation the model quality criterion is determined as a function of the model error (1):

$$E = \sum_{s=1}^{S} \left(y^s - f(w, x^s)\right)^2 \to 0 . \tag{1}$$

The decision tree model structure consists of nodes and links (connections between nodes). The decision tree model parameters are the numbers of features used in tree nodes, as well as their boundary values for splitting the ranges of feature values.

A random forest is a collection of trees as structures with parameters, as well as a transformation that combines the results of trees wok and the weight of trees in making the final decision.

3 Literature review

The methods of decision tree constructing [1-24] hierarchically divide the initial space into regions, in which they evaluate the output average value for the instances hit in the region (cluster). This value is assigned to the output feature of all instances hitting in this cluster. The advantage of this group of methods is the simplicity and interpretability of the resulting models, as well as the possibility of passing the cluster analysis tasks and selecting informative features. The disadvantages of the methods of this group are the low accuracy of the obtained models.

The random forest method aims to improve classification accuracy joining set of separately generated random models. This makes possible to approximate the interclass boundary more accurately in comparison with single decision tree.

Let the training sample consist of S instances, the dimension of the feature space is N, and the parameter m (the number of selected features) is specified. Set the number of individual models (trees) in the ensemble (forest) T as well as the maximum acceptable number of instances in the tree node or the maximum allowable tree height as a stop criterion Z.

The most common way to build a committee's tree is called bagging (short for bootstrap aggregation):

To build the next t-th model in a form of decision tree ($t = 1, 2, ..., T$):

– set the number of features for the t-th model $m_t < N$ (usually, only one value is used for all models $m_t = m$);

– generate a random subsample with S repetitions from the training set (thus, some instances will hit into it two or more times – on average $S(1-S^{-1})^S$, and about S/e instances (here e is an Euler's number) will not be included into it at all). Those instances that are not hit in the sample are called out-of-bag (unselected).

– build a decision tree based on the generated subsample, and during the creation of the next tree node we will randomly select m_t features from the N initial features, then from the m_t randomly selected features we will choose the best feature, based on which we will split instances. The choice of the best of these features can be done in various ways, for example, on the basis of the Gini criterion or the criterion of information gain. The tree is built until the subsample is completely exhausted, or until the tree reaches the maximum given height Z, or until no more than Z instances are found in each leaf of the tree, and it is not subjected to the pruning procedure.

The optimal number of trees is chosen in such a way as to minimize the classifier error on the test set. In case of its absence, the out-of-bag error estimate is minimized, defined as a recognition error for those instances that did not fall into the training subsample due to repetitions (their number is approximately *S*/*e*).

It is recommended to set $m = \sqrt{N}$ in classification problems, and $m=N/3$ in regression problems. It is also recommended in the classification problems to build each tree until there is one instance in each leaf, and in regression tasks to build each tree until there are five objects in each leaf.

Recognition of sample instances by a trained model in the form of a forest of decision trees is performed by voting: each forest's tree assigns a recognized instance to one of the classes, and the class for which the most of trees voted wins. More formally, the total forest model for the recognized instance *x*', submitted to its inputs, will determine the output value using the formula:

– in classification problems (2):

$$y(x') = \arg \max_{k=1,2,\ldots,K} \left\{ \frac{1}{T} \sum_{t=1}^{T} \{1 \mid y_t(x') = k\} \right\}; \tag{2}$$

– in the evaluation problems (3):

$$y(x') = \frac{1}{T} \sum_{t=1}^{T} y_t(x') \, . \tag{3}$$

The disadvantage of the known methods for constructing a random forest of decision trees is that the resulting forest turns out to be more precise than a single tree, but at times or even more complicated because of the large number of constructed models in the form of trees of the corresponding forest. This essentially not only increases the time expenditures for the decision making and the requirements for memory resources of the computer, but also leads to a significant reduction in the generalizing properties and interpretablility of the forest model in comparison with the model of the single tree.

Therefore, an urgent task is to develop methods that allow to synthesize models in the form of a random forest, free from the disadvantages noted above, or characterized by them to a lesser extent.

4 The method of forming a random forest of decision trees based on a priori information

To solve the problem of the development of a method free from disadvantages mentioned above, it is proposed in the process of building a forest of decision trees, along with a casual approach to their construction to use deterministic component - take into account the a priori information, which will allow on-directs the formation of models so as to concentrate on the most promising directions, preserving the overall stochastic nature of the model building process.

At the selecting a subsample to build a partial tree model, it is proposed to select samples for inclusion in the model randomly, but taking into account their individual informativeness, minimizing their similarity (increasing diversity) – maximizing the distance to the nearest instance.

At the current tree constructing, the choice of the root node is proposed to make randomly, but taking into account individual estimates of the informativeness of the features, increasing the chances of those features to be used in the root node, which are individually the most significant, and have not yet used in the roots of previously formed trees.

At the current non-root node forming, it is proposed to take into account the individual informativeness of the features their relationships with each other with respect to the output parameter.

1. Initialization: set the training sample <*x*, *y*>.

2. Estimate the a priori information:

– determine individual estimates of the informativeness of each *j*-th input feature [34-46] respectively to the output feature I_j, $j = 1, 2, ..., N$;

– determine individual estimates of the informativeness of each *i*-th input feature [34-46] respectively to the *j*-th input feature $I_{i,j}$, $j = 1, 2, ..., N$;

– estimate the individual informativeness of each sample instance I^s [35, 44], $s = 1, 2, ..., S$;

– determine the measure of the distance between the *s*-th and *p*-th instances of the sample in the feature space $d^{s,p}$. For small size samples evaluate by (4):

$$d^{s,p} = d^{p,s} = \sum_{j=1}^{N}(x_j^s - x_j^p)^2, \ s = 1, 2, ..., S, \ p = 1, 2, ..., s. \quad (4)$$

For large samples, instead of the distance between instances, it is possible to use the distances between their locally sensitive hashes [47] ealuated as (5):

$$d^{s,p} = H^s - H^p, \ s = 1, 2, ..., S, \ p = 1, 2, ..., s, \quad (5)$$

where H^s is a locally sensitive hash for *s*-th instance.

Calculation of hashes, unlike calculation of distances, will not require loading all instances into computer memory: hashes can be calculated in just one sample pass, operating with fragments in memory.

3. Building a forest of *T* decision trees. To build the *t*-th model as a decision tree ($t = 1, 2, ..., T$):

– set the number of attributes for the *t*-th model m_t <*N* (usually, only one value is used for all models m_t =*m*);

– generate a random subsample of size *S* with repetitions from the training sample. When selecting a subsample to build a partial model, it is proposed to select the samples for inclusion in it randomly, but taking into account their individual informativeness, minimizing their similarity (improving diversity) – maximizing the distance to the nearest instance using the modified rule of roulette [48] determined by (6):

$$V^s = \begin{cases} 1, r^s \le \dfrac{I^s}{\sum\limits_{p=1}^{S} I^p} \left(\dfrac{\max\limits_{p=1,2,\ldots,S} \{d^{s,p} \mid x^p \in \Theta\}}{\dfrac{1}{S - |\Theta|} \sum\limits_{g=1}^{S} \max\limits_{p=1,2,\ldots,S} \{d^{g,p} \mid x^g \notin \Theta, x^p \in \Theta\}} \right); \\ 0, \text{otherwise}, \end{cases} \quad (6)$$

where V^s is the binary decision to use the *s*-th instance at subsample forming (1 is use, 0 is not to use), Θ is a set of already selected instances for inclusion in the subsample, r^s is- a random number in the range [0, 1], generated to select *s*- th instance.

This rule should be applied consistently to the original data sample, as long as $|\Theta| \le S'$, where *S'* is a desired subsample volume.

The proposed rule will preserve the random nature of the selection, providing each instance with a chance to be selected for inclusion in the subsample, but will increase the chances of being selected for the individually most informative instances, as well as for those instances that are most different in the feature space from the already selected instances;

– for a given basic method of decision tree building, determine for each feature the value of the partitioning criterion I_{jM} used by the method (it is necessary to provide $I_{jM} \in [0,1]$), on the basis of which to determine the randomized selection criterion of the feature for splitting the tree in the root node by (7)-(9):

$$I_{jRand} = I_{jM} I_{jt} \hat{I}_j r, \quad (7)$$

$$I_{jt} = \frac{1}{1 + \sum\limits_{p=1}^{t-1} \{1 \mid root(p) = j\}}, \quad (8)$$

$$\hat{I}_j = \frac{I_j}{\max\limits_{i=1,2,\ldots,N} \{I_i\}}, \quad (9)$$

where *root*(*t*) is a number of the feature used in the root node of the *t*-th tree from already constructed forest of trees, *r* is a random number in the range [0, 1] generated for the new tree.

The proposed criterion I_{jRand}, preserving the criterion I_{jM} of the chosen method as a whole, will give it randomized properties, as well as increase the chances of those features to be placed in the root node that have not yet placed in the roots of previously formed trees, as well as individually are the most significant.

– select the best feature as the root node of the created *t*-th tree using the criterion I_{jRand}, and then split the sample according to the selected feature;

– build a decision tree based on the generated subsample using the resulting root node, and during the creation of the next node of the tree from the *N* initial features, randomly select the m_t features, taking into account their individual informativeness and interrelation with each other using the criterion (10):

$$I_{j*Rand} = I_{jM} \max(\hat{I}_j, I_{j,temp})r, \tag{10}$$

where is $temp$ a number for the current node of the feature, located in the parent node.

After that, from the randomly selected m_t features, using the base method, select the best feature, on which basis the partition will be performed. The tree is built until the subsample is completely exhausted or until the tree reaches the maximum given height *Z*, or until no more than *Z* instances are found in each leaf of the tree, and it does not undergo the pruning of branches.

The proposed method preserving the generally random nature of the selection of subsample instances in the process of building decision trees forest will increase the chances to be selected for constructing of the particular models of those instances that are individually more informative, and will also strive to provide a variety of instances used to construct particular models, also preserving the random principle of selection of a subset of candidate features for the formation of the current node of the synthesized decision tree will increase the chances to becoming the root of the tree of the individually most informative features that not yet used as root, and at the next node forming it will increase the chances to being used of those features that are individually the most significant and also most closely associated with the feature already used in the parent node.

5 Experiments and results

The proposed method were implemented as software and experimentally investigated in solving the problem of diagnosing the development of recurrent respiratory infections in young children.

Diseases of the respiratory system occupy a leading place among the entire pathology of young children of early age. It is problematic to examine children from whom infectious episodes acquire a protracted, severe, recurrent course. To create rehabilitation and prevention programs, it is necessary to determine the most significant risk factors, as well as their combination, which was solved within the framework of the task of diagnosing the development of recurrent respiratory infections in young children [48].

The data sample for the experiments was obtained in [48] on the basis of a survey of 108 children. The observations were characterized by 42 features reflecting the presence or absence of chronic diseases in the parents and the child: x_1 is child gender,

x_2 is a father's illness – allergy, x_3 is a father's illness – other chronic diseases, x_4 is a father's illness – healthy, x_5 is a mother's illness – allergy, x_6 is a maternal diseases – other chronic diseases, x_7 is a maternal diseases – healthy, x_8 is a normal pregnancy, x_9 is a mother sick during pregnancy, x_{10} is a normal delivery, x_{11} is a pregnancy was complete, x_{12} – whether the child was in the anaesthesiology department, x_{13} – whether the child was in the neonatal pathology department, x_{14} is a breastfeeding, x_{15} is a breastfeeding for up to one month, x_{16} is a breastfeeding for up to three months, x_{17} is a breastfeeding for up to one year, x_{18} is a breastfeeding feeding up to the present moment, x_{19} is a breastfeeding with a term of over a year, x_{20} is a diagnosis – cytomegalovirus infection, x_{21} id a diagnosis – anemia, x_{22} is a diagnosis – abnormality of the urinary system, x_{23} is a diagnosis – obstructive bronchitis, x_{24} is a diagnosis – recurrent obstructive bronchitis, x_{25} is a diagnosis – congenital heart disease, x_{26} is a diagnosis – urinary tract infection, x_{27} is a diagnosis – otitis, x_{28} is a diagnosis – acute respiratory viral infection, x_{29} is a diagnosis – acute stenotic laryngotracheobronchitis, x_{30} is a diagnosis – pneumonia, x_{31} is a diagnosis – rickets Ca-norm, x_{32} is a diagnosis – enterobiosis, x_{33} is a diagnosis – hypotrophy, x_{34} is a diagnosis –allergy, x_{35} is a diagnosis – central nervous system damage, x_{36} is a parents smoke, x_{37} is a older children in the family, x_{38} is a child attends care facility, $x_{\ 9}$ is a unfavourable living conditions, x_{40} is a child use antibiotics of 1-2 groups, x_{41} is a child use antibiotics of 3rd group, x_{42} is a child episodically ailing. The output feature y take a value "0" or "1" depending on the child's exposure to the development of recurrent respiratortion infections. The fragment of data sample is presented in the Table 1.

Table 1. Fragment of data sample

s	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	…	x_{42}	y
1	0	0	0	0	0	0	1	0	1	1	…	0	1
2	1	1	0	0	0	0	1	1	0	1	…	0	1
3	1	0	0	1	0	0	1	1	0	1	…	0	1
4	0	0	1	0	1	0	0	0	1	1	…	0	1
5	0	0	1	0	0	0	1	1	0	1	…	1	1
6	0	1	0	0	0	0	1	1	0	1	…	0	1
7	1	0	0	1	0	1	0	1	0	1	…	0	1
8	1	0	0	1	0	1	0	0	1	1	…	0	1
…	…	…	…	…	…	…	…	…	…	…	…	…	…
12	0	0	0	1	0	0	1	1	0	1	…	0	1
13	1	0	0	1	0	0	1	1	0	1	…	0	1
14	1	0	0	1	0	0	1	1	0	0	…	0	1
15	1	0	0	1	0	0	1	0	0	1	…	0	1
16	0	0	0	1	0	1	0	0	1	1	…	0	1
17	1	0	0	1	0	0	1	1	0	1	…	0	0
18	0	0	0	1	0	0	1	0	0	1	…	0	1
19	0	0	0	1	0	0	1	1	1	1	…	0	1
20	0	0	0	1	0	0	1	0	0	1	…	0	1
21	0	0	0	1	0	0	1	0	0	1	…	1	0
22	0	1	1	0	1	1	0	0	1	0	…	0	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…
108	0	0	0	1	0	0	1	1	1	0	…	0	0

In the experiments, the original sample was randomly divided into training and test samples of the same size. Wherein it was ensured the preservation of the frequencies of the classes in the training sample relative to the original sample. On the basis of the training sample, models were built in the form of decision trees. For each model, the errors were determined on the basis of training and test data, as well as the time of work creation.

The results of the experiments are presented in the Table 2. Here E_{tr} is a model error for training sample, t_{tr} is a time of building, t_{trr} is a time of recognition of training sample, E_{tst} is a model error for test sample, t_{tst} is a time of recognition of test sample.

Table 2. Results of experiments

Model architecture and training method	E_{tr}	t_{tr}, sec	t_{trr}, sec	E_{tst}	t_{tst}, sec
Single decision tree, Breiman's method [2]	0	0.73	0.09	3	0.09
Random forest, method [25]	0	5.78	0.65	1	0.65
Random forest, proposed method	0	5.92	0.65	0	0.65

The results of the experiments presented in the Table 1 show that the proposed method allows to obtain models that is better in accuracy (of lower error) in comparison with existent tree building methods [2, 25], but requires a little more time to build a model, the speed of the constructed model for proposed method does not essentially differ from the speed of a random forest model constructed using the method [25], but, as was to be expected, random forest-based models require significantly more time for calculations compared to the single-tree model [2].

Increasing the accuracy of problem solving by a model built on the basis of the proposed method is provided, on the one hand, by increasing the diversity of decisions for root feature selection, and, on the other hand, by taking into account the individual informativeness of features at forming the decision-making hierarchy. Whereas the proposed method preserves the generally random nature of the formation of the forest of the decisive trees.

6 Conclusion

The problem of inductive model building on precedents for biomedical applications is considered in the paper.

The model paradigm is a random forest as a set of decision tree classifiers working as ensemble.

A random forest method is proposed. It for a given training sample builds a set of trees for hierarchical clustering of instances. The a priori information taken from training data set is used in proposed method of random forest model building to provide more accurate model saving general random character of a method.

The software implementing the proposed methods has been developed and studied at the diagnosis problem solving. The conducted experiments have confirmed the performance of the developed software and allow recommending it for use in practice.

The resulting random forest provide more accurate model in comparison with a single decision tree, but it comparison with known methods of random forest model building proposed method is more accurate.

The prospects for further research are to test the proposed methods on a wider set of applied problems, to study the dependence of the speed and accuracy (error) of method's work on the sample volume and the feature number in the original sample.

References

1. Amit, Y., Geman, D., Wilder, K.: Joint induction of shape features and tree classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(11), pp. 1300–1305 (1997)
2. Breiman, L., Friedman, J. H., Stone, C. J., Olshen, R. A.: Classification and regression trees. Chapman & Hall / CRC, Boca Raton (1984)
3. Rabcan, J., Rusnak, P., Subbotin, S.: Classification by fuzzy decision trees inducted based on Cumulative Mutual Information. In: 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2018 - Proceedings , Slavske, 20-24 February 2018, pp. 208-212 (2018)
4. Dietterich, T. G., Kong, E. B.: Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. Machine Learning. 255 (1995).
5. Friedman, J.H.: A recursive partitioning decision rule for nonparametric classification. IEEE Transactions on Computers. 100(4), pp. 404–408 (1977)
6. Geurts, P., Irrthum, A., Wehenkel, L.: Supervised learning with decision tree-based methods in computational and systems biology. Molecular Biosystems. 5(12), pp. 1593–1605 (2009).
7. Geurts, P.: Contributions to decision tree induction: bias/variance tradeoff and time series classification. PhD Thesis. University of Liège (2002)
8. Murthy, K. V. S., Kasif, S., Salzberg, S.: A system for induction of oblique decision trees. Journal of Artificial Intelligence Research archive. 2 (1) (1994)
9. Hothorn, T., Hornik, K., Zeileis, A.: Unbiased recursive partitioning: A conditional inference framework. Journal of Computational and Graphical Statistics. 15(3), pp. 651–674 (2006)
10. Kim, H., Loh, W.-Y.: Classification trees with unbiased multiway splits. Journal of the American Statistical Association. 96 (454) (2001)
11. Kufrin, R.: Decision trees on parallel processors. Machine Intelligence and Pattern Recognition. 20 (1997)
12. Kwok, S. W., Carter, C.: Multiple decision trees. In: Uncertainty in Artificial Intelligence. 9 (4), pp. 327–338 (1990)
13. Mingers, J.: An empirical comparison of pruning methods for decision tree induction. Machine learning. 4(2), pp. 227–243 (1989)
14. Murthy, K. V. S.: On growing better decision trees from data. PhD thesis. The Johns Hopkins University (1995)
15. Quinlan, J.R.: C4.5 Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann. (1992)

16. Oliinyk, A.A., Subbotin, S.A.: The decision tree construction based on a stochastic search for the neuro-fuzzy network synthesis. Optical Memory and Neural Networks (Information Optics), 24 (1), pp. 18-27 (2015). doi: 10.3103/S1060992X15010038
17. Quinlan, J. R.: Induction of decision trees. Machine learning. 1(1), pp. 81– 106 (1986)
18. Quinlan, J. R.: Simplifying decision trees. International Journal of Man-Machine Studies. 27 (1987). doi:10.1016/S0020-7373(87)80053-6
19. Quinlan, R. : Learning efficient classification procedures. In: Machine Learning: an artificial intelligence approach, Michalski, Carbonell & Mitchell (eds.), Morgan Kaufmann, pp. 463-482 (1983). doi:10.1007/978-3-662-12405-5_15
20. Subbotin, S., Kirsanova, E.: The regression tree model building based on a cluster-regression approximation for data-driven medicine. CEUR Workshop Proceedings. 2255, pp. 155-169 (2018)
21. Strobl, C., Boulesteix A.-L., Augustin T.: Unbiased split selection for classification trees based on the gini index. Computational Statistics & Data Analysis, 52(1), pp. 483–501 (2007)
22. Utgoff, P. E.: Incremental induction of decision trees. Machine learning. 4(2), pp. 161-186 (1989). doi:10.1023/A:1022699900025
23. Wehenkel, L.: On uncertainty measures used for decision tree induction. In: Information Processing and Management of Uncertainty in Knowledge-Based Systems. pp. 413-418 (1996)
24. White, A. P., Liu, W. Z. Technical note: Bias in information-based measures in decision tree induction. Machine Learning. 15(3), pp. 321–329 (1994)
25. Breiman, L.: Random Forests. Machine Learning. 45 (1), pp. 5–32 (2001). doi:10.1023/A:1010933404324.
26. Ho, T. K.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence. 20(8), pp. 832–844 (1998)
27. Botta, V.: A walk into random forests: adaptation and application to genome-wide association studies. PhD Thesis. Université de Lièg (2013)
28. Denisko, D., Hoffman, M.: Classification and interaction in random forests. Proceedings of the National Academy of Sciences of the United States of America. 115 (8), pp 1690–1692 (2018). doi:10.1073/pnas.1800256115. PMC 5828645. PMID 29440440.
29. Louppe, G.: Understanding Random Forests: From Theory to Practice. PhD Thesis, University of Liege (2014)
30. Boulesteix, A.-L., Janitza, S., Kruppa, J., König, I. R.: Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2(6), pp. 493–507 (2012)
31. Criminisi, A., Shotton, J.: Decision Forests for Computer Vision and Medical Image Analysis. Springer (2013)
32. Panovand, P., Džeroski, S.: Combining bagging and random subspaces to create better ensembles. Advances in intelligent data analysis. VII, pp. 118–129 (2007)
33. Wolpert, D. H., Macready, W. G.: An efficient method to estimate bagging's generalization error. Machine Learning. 35(1), pp. 41–55 (1999)
34. De Mántaras, R. L.: A distance-based attribute selection measure for decision tree induction. Machine learning. 6(1), pp. 81–92 (1991).
35. Subbotin, S.: The instance and feature selection for neural network based diagnosis of chronic obstructive bronchitis. Studies in Computational Intelligence, vol. 606, pp. 215-228 (2015)

36. Miyakawa, M.: Criteria for selecting a variable in the construction of efficient decision trees. IEEE Transactions on Computers. 38(1), pp. 130–141 (1989).
37. Subbotin, S.A.: Methods of sampling based on exhaustive and evolutionary search Automatic Control and Computer Sciences. 47(3), pp 113-121 (2013)
38. Altmann, A., Toloşi, L., Sander, O., Lengauer, T.: Permutation importance: a corrected feature importance measure. Bioinformatics. 26 (2010). doi:10.1093/bioinformatics/btq134. PMID 20385727.
39. Subbotin S.: Quasi-relief method of informative features selection for classification. In: 2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 - Proceedings, 11–14 September 2018, pp. 318-321 (2018). doi: 10.1109/STC-CSIT.2018.8526627
40. Oliinyk, A., Subbotin, S., Lovkin, V., Leoshchenko, S., Zaiko, T.: Feature selection based on parallel stochastic computing 2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 - Proceedings, 1, 11 September 2018 through 14 September 2018, pp. 347-351. (2018) DOI: 10.1109/STC-CSIT.2018.8526729
41. Painsky, A., Rosset, S. Cross-Validated Variable Selection in Tree-Based Methods Improves Predictive Performance. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39 (11), pp. 2142–2153. (2017). doi:10.1109/tpami.2016.2636831. PMID 28114007.
42. Oliinyk, A., Subbotin, S., Lovkin, V., Leoshchenko, S., Zaiko, T.: Development of the indicator set of the features informativeness estimation for recognition and diagnostic model synthesis. In: 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2018 - Proceedings, 20-24 February 2018, pp. 903-908 (2018)
43. Mingers, J.: An empirical comparison of selection measures for decision-tree induction. Machine learning. 3(4), pp. 319–342 (1989).
44. Subbotin, S.: The instance and feature selection for neural network based diagnosis of chronic obstructive bronchitis. In: Studies in Computational Intelligence. 606, pp. 215-228 (2015). doi: 10.1007/978-3-319-19147-8_13
45. Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., Zeileis, A.: Conditional variable mportance or random forests. BMCbioinformatics. 9 (2008)
46. Subbotin, S., Oleynik, A.: Entropy based evolutionary search for feature selection. In: The Experience of Designing and Application of CAD Systems in Microelectronics - Proceedings of the 9th International Conference, CADSM 2007, Lviv-Polyana, 20-24 February 2007, pp. 442-443 (2007). doi: 10.1109/CADSM.2007.4297612
47. Pauleve, L.; Jegou, H.; Amsaleg, L..: Locality sensitive hashing: A comparison of hash function types and querying mechanisms. Pattern Recognition Letters. 31 (11), pp. 1348–1358 (2010). doi:10.1016/j.patrec.2010.04.004.
48. Gerasimchuk, T., Zaitsev, S., Subbotin, S.: The use of artificial immune systems to predict the risk of recurrent respiratory infections in young children. In: Diahnostyka ta likuvannya infektsiyno oposeredkovanykh somatychnykh zakhvoryuvanʹ u ditey: mizhrehionalʹna naukovo-praktychna konferentsiya, Donetsk, 10–11 February 2011. pp, 27–29 (2011)