INFORMATICA, 2025, Vol. 36, No. 3, 657–676

DOI: https://doi.org/10.15388/25-INFOR601

Quantifying Binary Classifier Algorithms Similarity with a Consensus Agreement Approach

Ana PERIŠIĆ[1,2,*], Sophie VANBELLE[3], Rafaela Brigita PETRIČEVIĆ[4]

[1] *Ruđera Boškovića 33 21000 Split, University of Split, Croatia*
[2] *Trg Andrije Hebranga 11, 22000, Šibenik, Šibenik University of Applied Sciences, Croatia*
[3] *Peter Debyeplein 1 6229 HA Maastricht, CAPHRI, Maastricht University, The Netherlands*
[4] *Split, Independent Researcher, Croatia*
e-mail: ana.perisic@pmfst.hr, sisak@vus.hr, sophie.vanbelle@maastrichtuniversity.nl, rafaelap98@gmail.com

Received: January 2025; accepted: September 2025

Abstract. Most classification algorithms involve subjective inputs or hyperparameters to be determined prior to performing the classification. When taking different input or hyperparameter values, each classification algorithm will comprise a collection of classifiers. In this work, we propose a data-driven methodology for assessing similarity in consensus agreement within such a collection of classifiers, and between two classification algorithms, conditional on the dataset of interest. The core of our approach lies in considering the variability introduced by different hyperparameter values for each algorithm when performing such comparisons. We address these problems by evaluating the similarity through consensus agreement and by proposing the application of asymmetric similarity indices based on the Jaccard coefficient. We present the proposed methodology on two publicly available datasets.

Key words: similarity, binary classification, consensus agreement, Jaccard coefficient, classifier sets.

1. Introduction

Classifying objects into classes is a common problem in many research fields. A plethora of supervised machine learning algorithms have been developed: Logistic Regression, Decision Trees, Random Forests (RF), Naive Bayes, Support Vector Machines, Neural Networks and many more. We typically evaluate these algorithms on their predictive performance by comparing predicted and true classes and evaluating measures such as accuracy, sensitivity, and specificity (Han *et al.*, 2012). This is because we are seeking to find the optimal predictive model, which means finding the model with the highest prediction performance.

Hyperparameter tuning is a crucial step in building an optimal predictive model, as most machine learning algorithms depend on carefully chosen hyperparameter values that

*Corresponding author.

significantly impact their performance (Bischl *et al.*, 2023). For example, when applying RF, the researcher needs to decide on the number of trees and the number of features to sample at each split point. Additionally, other subjective decisions or subjective inputs are required during the model-building process, such as determining the initial set of (relevant) predictors to include. Hyperparameter tuning is usually performed by selecting values that optimize a predetermined metric, such as accuracy, and this process can be automatized (Feurer and Hutter, 2019). Other decisions, such as selecting relevant predictors, can be theory-driven and do not necessarily involve hyperparameters, yet can still significantly impact the model and its classification results. The greater the impact of these decisions, the more sensitive the algorithm becomes to hyperparameter values, resulting in an increased diversity of the resulting classifications. This paper focuses on the study of this diversity when applying different hyperparameter values. We propose a data-driven methodology for assessing similarity between two classification algorithms, conditional on the dataset on which the algorithm is applied. This means that classifiers are evaluated through predicted classifications. The novelty of our approach stems from:

(a) evaluating each classification algorithm not as a single representative classifier, but as a collection of classifiers built under the same algorithm with varying hyperparameter values,
(b) accounting for the diversity among the corresponding classifiers associated with each binary classification algorithm.

The core of this research is therefore in evaluating similarity between two binary classification algorithms when taking into account different hyperparameter values and subjective inputs related to a specific algorithm. Accounting for the diversity of the classifiers associated with each binary classification algorithm imposes two main research questions:

(R1) How to evaluate the similarity of classifiers for a given binary classification algorithm when applying different hyperparameter values or different subjective inputs?
(R2) How to evaluate the classification similarity between two binary classification algorithms when applying different hyperparameter values or different subjective inputs for each algorithm?

In general, research question (R1) relates to the problem of evaluating similarity between k binary classifiers, while research question (R2) can be generalized to the problem of comparing two binary classifier sets. We focus on evaluating the similarity in consensus agreement where we quantify the extent to which all classifiers unanimously predict the same label for a given instance. The formal definition of the consensus agreement is presented in Section 3. We address these problems by evaluating the similarity through consensus agreement and by proposing the application of asymmetric similarity indices based on the Jaccard coefficient established in Perišić and Vanbelle (2024). Approaches for measuring similarity of binary classifiers are presented in Section 2, while Section 3 presents the basic theoretical framework related to binary classifiers and consensus agreement, and the methodological framework. In Section 4 we present the application of the proposed framework to publicly available datasets. Section 5 comprises the discussion, conclusion, and propositions for future work.

2. Comparing Binary Classification Algorithms

When measuring similarity, we can have different approaches to what similarity or dissimilarity means. These different approaches can be a result of different definitions of similarity, different fields of application, or different goals of the analysis. Moreover, across different fields, the nomenclature regarding the similarity assessment is not uniform. When evaluating the similarity of two binary classifiers, researchers have used terms (dis)similarity, resemblance, diversity, (dis)agreement. Thus, we are frequently working with the same concept, but using different names.

Evaluating similarity mostly implies quantification of similarity, which is often operationalized by applying a certain similarity measure. The broad application of measuring similarity resulted in a plethora of (dis)simmilarity/agreement/resemblance/diversity measures (see for instance, Choi *et al.*, 2010). Some of these measures have varying terminology across different fields or their application (for instance, the proportion of agreement and the simple matching coefficient).

When building classification models, especially on imbalanced datasets, we are often more interested in correctly identifying 1s (e.g. presence) than 0s. In such cases, we focus on one class rather than both classes equally. This can be achieved using asymmetric similarity coefficients, the most common being the Jaccard coefficient, which has been used for measuring the agreement between two classifiers. Extensions of this coefficient to multiple classifiers include the k-adic Jaccard coefficient (Warrens, 2009) and the 2-group k-adic Jaccard coefficient (Perišić and Vanbelle, 2024).

Many measures of the connection between two classifier outputs can be derived from the statistical literature, but there is less clarity on the subject when three or more classifiers are concerned (Kuncheva and Whitaker, 2003). Often, researchers simplify the problem of comparing $k > 2$ classifiers by evaluating pairwise similarities where they compare 2 classifiers and then take a weighted linear combination. Also, there have been extensions of some well-known similarity coefficients, like the simple matching and the Jaccard coefficient (Warrens, 2009). When it comes to measuring the similarity between two sets of classifiers, the situation worsens. This problem is often reduced to measuring the similarity of two classifiers by applying consensus-based methods. The consensus-based practice determines empirically a consensus category in each group of classifiers and reduces the problem to the case of two classifiers (Vanbelle and Albert, 2009). There have been some extensions of the well-known similarity measures, for instance the Cohen's kappa coefficient (Vanbelle and Albert, 2009), the Jaccard coefficient, and the Simple matching coefficient (Perišić and Vanbelle, 2024).

Evaluating similarity between binary classifiers can serve different purposes. In addition to comparing technical aspects such as the complexity and execution time of the underlying algorithm, one can focus on comparing their predictive performances or examining their agreement when classifying objects. Researchers have adopted different approaches, focusing on only one or more features for comparison. For instance, Makhtar *et al.* (2011) argue that in order to assess the similarity of predictive models as a whole, it is necessary to assess the similarity of their individual elements (Input, Function and Output) independently. They proposed a methodology to measure classifier similarity based

on datasets, model functions, and confusion matrices. This includes the Dataset Similarity Coefficient, Similarity of Predictive Model Function, and Similarity of Output, which are combined into a single overall similarity coefficient.

The agreement between two classifiers can generally be evaluated by comparing the predicted classes for each object, as well as their agreement with reference values. When these reference values are the true classes, the agreement essentially reflects the predictive performance of an algorithm. Rather than comparing the agreement between two classifiers in general, researchers often focus on comparing the agreement between predicted and true classes, which is the basis of the prediction performance assessment, using performance measures like accuracy, sensitivity or AUC (see for instance, Labatut and Cherifi, 2012), the Matthews correlation coefficient, the Brier score, Cohen's Kappa (Chicco *et al.*, 2021). This is due to the fact that when building a prediction model, we aim to find a classifier that has the highest prediction performance. Given a dataset, the classifier with the highest prediction metrics is selected. As already noted, this procedure can be viewed as assessing the agreement between predicted classes and true classes.

Different procedures have been proposed for comparing multiple classifiers. For instance, the Friedman test (Demšar, 2006) followed by tests for multiple comparisons such as the mean-ranks test, sign-test or the Wilcoxon signed-rank test (Benavoli *et al.*, 2016). A broad range of studies analysed classification metrics in terms of specific characteristics and properties, such as coherency of metrics used for evaluating classifiers, consistency in binary classification, robustness of binary classification performance metrics, etc. Some examples can be found in Shirdel *et al.* (2024).

When assessing the agreement between two classifiers in terms of agreement with true classes, we can further distinguish between correct and incorrect predicted classifications. For instance, Petrakos *et al.* (2001) have separately assessed agreement on correct classification and agreement on incorrect classification by matching the results from individual classifiers. They organize the information as a cross-classification table and then apply the proportion of agreement (the simple matching coefficient), the proportion of specific agreement, and the kappa statistic to assess classifier agreement. Wang *et al.* (2023) established a framework that allows for visual comparison of two classification models and enables the interpretation of the two models' behaviour discrepancy on different feature conditions. This framework identifies data instances with disagreed predictions from the two compared classifiers and trains a discriminator to learn from the disagreed instances.

Comparing two classifiers by assessing the similarity in their predicted classifications has gained attention in the optimization of ensemble algorithms, often referred to as measuring ensemble diversity. Measuring ensemble diversity, as measuring classifier agreement or similarity, is not straightforward, and as a result, there is no universally accepted formal definition of diversity, nor a universally accepted measure. A natural way to measure the diversity of classifiers is to measure how much their classification decisions differ. Again, this difference can be compared by comparing the predicted classifications, i.e. comparing the predicted class of each object, or by examining the diversity in correct and incorrect classifications. Researchers have proposed a number of ways to measure diversity in ensembles of classifiers (e.g. Kuncheva and Whitaker, 2003; Wood *et al.*, 2023;

Tsymbal *et al.*, 2005) and to generalize a diversity measure between two classifiers to an ensemble of more than two classifiers. For instance, when considering more than two classifiers, we can differentiate between measures averaged over pairs of classifiers and measures defined on the ensemble of classifiers as a whole (Narasimhamurthy, 2005). Tsymbal *et al.* (2005) considered different measures of ensemble diversity, where they differentiated between the pairwise (that measure diversity in predictions of a pair of classifiers, or the total ensemble diversity, which is the average of all the classifier pairs in the ensemble), and non-pairwise (which measure diversity in predictions of the whole ensemble only). In the binary classification case, they propose the simple matching coefficient (referred to as the plain disagreement and the fail/non-fail disagreement), the Yule's Q statistics, the correlation coefficient and the kappa statistic. The non-pairwise measures rely on entropy and ambiguity. Kuncheva and Whitaker (2003) studied ten statistics which can measure diversity among binary classifier outputs (in terms of correct or incorrect vote for the class label): four averaged pairwise measures (the Q statistic, the correlation, the disagreement measure and the double fault measure) and six non-pairwise measures (the entropy of the votes, the difficulty index, the Kohavi-Wolpert variance, the interrater agreement, the generalized diversity, and the coincident failure diversity). All these pairwise measures have been proposed as measures of (dis)similarity in the numerical taxonomy literature (Sokal and Sneath, 1963). A theoretical analysis on six existing diversity measures (namely disagreement measure, double fault measure, KW variance, inter-rater agreement, generalized diversity and measure of difficulty) and some underlying relationships between them can be found in Tang *et al.* (2006). Also, one of the most widely used measures of classifier agreement is the Cohen's kappa coefficient which was also used as a measure of classifier diversity (Margineantu and Dietterich, 1997; Zouari *et al.*, 2005).

3. Methodology

This section outlines the proposed methodology for quantifying the similarity between binary classifier algorithms. First, we present the methodological framework for evaluating the similarity between two sets of binary classifiers, as established in Perišić and Vanbelle (2024). Then, we introduce a framework that integrates this approach into a broader methodology for assessing the similarity of binary classifier algorithms.

3.1. *Evaluating Similarity in Consensus Agreement for Binary Classification*

We start this section by introducing formal definitions of a binary classifier and a binary classifier set, followed by some fundamental definitions related to classifier agreement. Note that these definitions are made conditionally to a given dataset, i.e. the set of objects to be classified.

Definition 1 (*Binary classifier*). Let $O = \{o_1, \ldots, o_n\}$, $n \in \mathbb{N}$, be a finite set of n objects. A binary classifier on O is a function $c : O \to \{0, 1\}$.

Definition 2 (*Binary classifier set*). Let $O = \{o_1, \ldots, o_n\}$, $n \in \mathbb{N}$, be a finite set of n objects and $c_1, \ldots, c_k$, $k \in \mathbb{N}$, binary classifiers on O. Binary classifier set on O is a collection of classifiers $c_1, \ldots, c_k$. We write $C = \{c_1, c_2, \ldots c_k\}$.

We are examining the agreement of two binary classifiers. Although it is intuitively clear when two classifiers agree, we present a formal definition.

Definition 3. Let c_1 and c_2 be two binary classifiers. Binary classifiers agree on the classification of an object o if $c_1(o) = c_2(o)$.

This corresponds to, depending on the research area, the De Moivre definition of agreement (Hubert, 1977), consensus agreement, or the k-adic definition of similarity (Warrens, 2009). We evaluate the strength of the agreement on a set of objects $O = \{o_1, \ldots, o_n\}$. We are interested in the consensus agreement within a binary classifier set and distinguish between two types of consensus agreement: the consensus agreement on the presence ($c_1(o) = c_2(o) = 1$) and the consensus agreement on the absence ($c_1(o) = c_2(o) = 0$).

Definition 4 (*Consensus agreement*). Let $C = \{c_1, \ldots, c_k\}$ be a finite set of classifiers and $O = \{o_1, \ldots, o_n\}$ a finite set of objects.

(1) A set of classifiers C has a consensus agreement on the presence for an object $o \in O$ if $c_j(o) = 1$, $j = 1, 2, \ldots, k$.
(2) A set of classifiers C has a consensus agreement on the absence for an object $o \in O$ if $c_j(o) = 0$, $j = 1, 2, \ldots, k$.

When evaluating the similarity of classifiers $c_1, c_2, \ldots, c_k$, $k \geqslant 2$, we propose the application of the $k-$adic Jaccard coefficient, defined as

$$J = \frac{a^{(k)}}{n - d^{(k)}}, \tag{1}$$

where $a^{(k)}$ is the number of objects classified as 1 by all classifiers $c_1, c_2, \ldots, c_k$, and $d^{(k)}$ is the number of objects classified as 0 by all classifiers $c_1, c_2, \ldots, c_k$. The value of this coefficient can be interpreted as the percentage of objects classified as 1 by all classifiers out of objects that were classified as 1 by at least one classifier.

When evaluating the similarity of 2 binary classifier sets, we propose the application of the 2-group k-adic Jaccard coefficient established in Perišić and Vanbelle (2024). The 2-group k-adic Jaccard coefficient is defined as follows. Let C_1 and C_2 be two sets of binary classifiers on the set of objects O. The 2-group k-adic Jaccard similarity coefficient is obtained by applying steps (1) to (3) as follows:

(1) Determine the k-adic Jaccard coefficient separately for each classifier set by calculating $J_{C_l} = \frac{a_{C_l}}{n - d_{C_l}}$, $l = 1, 2$, where a_{C_l} is the number of objects classified as 1 by all classifiers from C_l, and d_{C_l} is the number of objects classified as 0 by all classifiers from C_l, $l = 1, 2$.

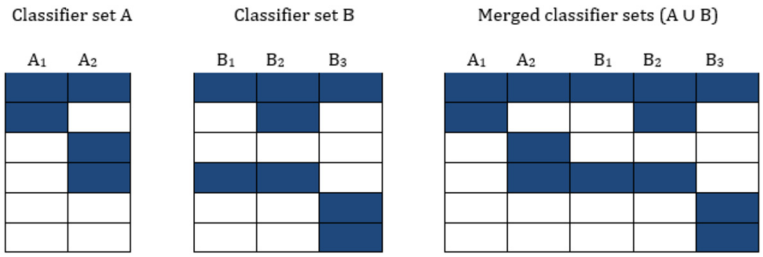

Fig. 1. Example 1A: Binary classifier sets.

(2) Merge the classifiers from sets C_1 and C_2 into one set $C = C_1 \cup C_2 = \{c_{11}, \ldots, c_{1k_1}, c_{21}, \ldots, c_{2k_2}\}$. Determine the k-adic Jaccard coefficient J_C for the classifier set C, $J_C = \frac{a_C}{n-d_C}$, where a_C is the number of objects classified as 1 by all classifiers from C, and d_C is the number of objects classified as 0 by all classifiers from C.

(3) Determine the 2-group k-adic Jaccard similarity coefficient as, depending whether one set of classifiers can be considered as the reference,

$$J_{group}(C_1, C_2) = \frac{J_C}{\frac{1}{2}(J_{C_1} + J_{C_2})}, \tag{2}$$

or

$$J^{ref}_{group}(C_1, C_2) = \frac{J_C}{J_{C_{ref}}}, \quad \text{where } C_{ref} \text{ is a reference set, } ref \in \{1, 2\}. \tag{3}$$

It can be shown that $0 \leqslant J^{ref}_{group}(C_1, C_2), J_{group}(C_1, C_2) \leqslant 1$. The interpretation of the 2-group k-adic Jaccard coefficient depends on the selected approach. In both cases, the coefficient gives the percentage of a within-set consensus agreement on presence that two classifier sets share. When applying Eq. (2), it represents the percentage of the mean within-set similarity preserved when merging the two sets. In the case of Eq. (3), it is the percentage of the reference within-set similarity preserved when merging the two sets. We present the calculation and interpretation of the $J_{group}(C_1, C_2)$ coefficient on two simple examples.

Example 1. Assume that we have two sets of binary classifiers: classifier set A comprising classifiers A_1 and A_2, and a classifier set B comprising classifiers B_1, B_2, and B_3. Further assume that a set of 6 objects is classified on a binary scale as shown in Fig. 1. In the figure, objects are represented as rows, classifiers as columns, blue cells indicate a classification of 1 (presence), and white cells indicate a classification of 0 (absence).

We quantify within-set similarity in consensus agreement on presence by calculating the k-adic Jaccard coefficient (see equation (1)). We obtain $J_A = \frac{1}{4} = 25\%$ for classifier set A, and $J_B = \frac{1}{5} = 20\%$ for classifier set B. When merging classifier sets A and B, the within-set similarity in consensus agreement drops to $J_{A \cup B} = \frac{1}{6} \approx 16.67\%$. To

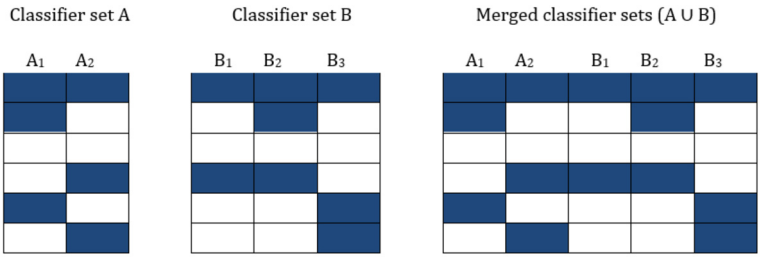

Fig. 2. Example 1B: Binary classifier sets.

determine the similarity in consensus agreement on presence between classifier sets A and B, let's first consider that there is no reference classifier (see Eq. (2)). We obtain $J_{group}(A, B) = \frac{J_{A\cup B}}{\frac{1}{2}(J_A+J_B)} = \frac{\frac{1}{6}}{\frac{1}{2}(\frac{1}{4}+\frac{1}{5})} \approx 74.1\%$. This means that when merging classifier sets A and B, 74.1% of the mean within-set similarity in consensus agreement of A and B would be preserved (i.e. 25.9% would be lost). Now assume that classifier A is a reference. We have $J_{group}(A, B) = \frac{J_{A\cup B}}{J_A} = \frac{2}{3} \approx 66.7\%$ (see Eq. (3)). It means that 66.7% of the within-set similarity in consensus agreement of classifier set A would be preserved when merging classifier sets A and B. Similarly, when B is the reference group, we have $J_{group}(A, B) = \frac{J_{A\cup B}}{J_A+J_B} = \frac{5}{6} \approx 83.3\%$.

Now, consider a slightly different case where two classifiers have both consensus agreement on presence (row 1) and consensus agreement on absence (row 3) for the same sets of objects, as presented in Fig. 2.

Here we have $J_A = J_B = J_{A\cup B} = \frac{1}{5} = 20\%$. Thus, $J_{group}(A, B) = J^{ref}_{group}(A, B) = 1$, regardless of the group taken as a reference. Since $J_{group} \leqslant 1$, this is the highest possible level of similarity in consensus agreement on presence, and thus 100% of the similarity of the consensus agreement on presence would be preserved (0% would be lost) when merging classifier sets A and B.

In the application part, none of the classifier set is considered as a reference (Eq. (2)). The properties of that 2-group k-adic Jaccard coefficient and the derivation of the associated confidence interval can be found in Perišić and Vanbelle (2024).

3.2. *Quantifying Binary Classifier Algorithms Similarity*

In this Section we present the framework for measuring similarity in consensus agreement between two binary classification algorithms. We will consider each binary classification algorithm to be a set of $k \geqslant 2$ binary classifiers, obtained by changing hyperparameter values/subjective inputs. Let $\mathcal{H}$ be a set of hyperparameter values selected for a certain binary classification algorithm. In general, we are evaluating a set of classifiers $C = \{c_h, h \in \mathcal{H}\}$, where $\mathcal{H}$ is an index set. In this paper, we are evaluating algorithms through a collection of classifiers obtained by applying different hyperparameter values or subjective decisions when adjusting the algorithm. So $\mathcal{H}$ will be related to the set of hyperparameter values of interest. Although a binary classifier set can be created in an arbitrary way, we consider

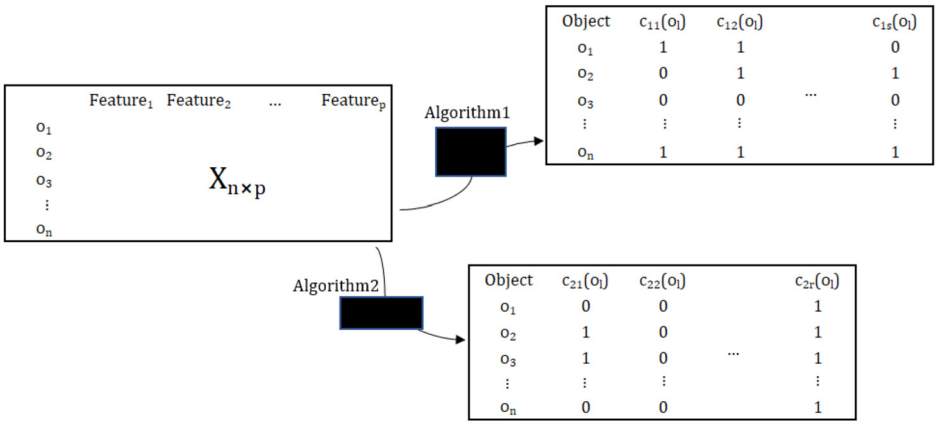

Fig. 3. Two classifier sets example. $X_{n\times p}$ presents the data matrix comprising values of p features for n objects that are classified by applying two algorithms; each algorithm takes into account different hyperparameters, thus comprises a set of classifiers; classification by each algorithm is presented as a matrix where $c_{ij}(o_l)$ presents the predicted classification of an object o_l by the classifier c_{ij}.

that there is some reasonable logic behind the creation of such a collection. For instance, a classifier set can be created by evaluating classification rules set by experts from a certain field or institution. We introduce some examples of such collections. Note that we use the term feature to represent traits and variables.

Example 2. Let $O = \{o_1, \ldots, o_n\}$ be a set of objects/items/subjects and $X_1, \ldots, X_p$ a set of features. The objects from O will be classified on a binary scale by applying a certain algorithm on the basis of the feature values. In other words, we are working with a sample of n tuples $(x_1(i), x_2(i), \ldots, x_p(i), c(i))$, $i = 1, 2, \ldots, n$, where $x_1(i), x_2(i), \ldots, x_p(i)$ represents the observed feature values and $c(i) \in \{0, 1\}$ is the predicted class for the object o_i.

(E1) Consider a classifier set $C = \{c_1, c_2, c_3\}$ of three logistic regression classifiers where c_1 represents a model with all features included, c_2 represents a model with a $1 < k < p$ features included (for instance, a model built in a stepwise fashion), and c_3 is a univariate model that includes only one feature with the highest (univariate) prediction performance.

(E2) Consider a classifier set $C = \{c_{ij},\ i \in I, j \in J\}$, where c_{ij} is a random forest classifier with hyperparameter values i (number of trees) and j (number of features) that are chosen from a predefined sets of possible values or the values of interest, I and J respectively.

When applied to a dataset of interest, each binary classifier produces a binary output vector, while each algorithm, taking into account different hyperparameter values, produces a set of binary vectors. Figure 3 presents a visualization of some possible collections of binary classifiers related to two algorithms, applied on a set of objects $\{o_1, o_2, \ldots, o_n\}$, where each algorithm is represented by a set of classifiers.

Research question (R1) requires assessing the similarity between k, $k \geqslant 2$ binary vectors. We propose that for a given dataset of interest and a selected binary classification algorithm, the evaluation of the similarity of classifiers is performed as follows.

($Step_1$) Select the set of hyperparemeter values that defines a set of classifiers.
($Step_2$) For each classifier, perform binary classification.
($Step_3$) Apply the k-adic Jaccard coefficient to the set of observed classifications.

Research question (R2) requires assessing the similarity between 2 sets or collections of binary vectors. Here, we assume that each binary classification set comprises at least two binary classifiers and propose that for a given dataset and two binary classification algorithms, the similarity between two binary classification algorithms is evaluated as follows.

($Step_1$) For each binary classification algorithm, select the set of hyperparameter values that defines a set of classifiers.
($Step_2$) For each classifier, perform binary classification. This results with two sets of binary classifications.
($Step_3$) Apply the 2-group k-adic Jaccard coefficient to the two sets of observed classifications.

4. Application

This section presents the application of the k-adic Jaccard similarity coefficient and the 2-group k-adic Jaccard similarity coefficient for measuring similarity in consensus agreement for a binary classification algorithm, and between two binary classification algorithms respectively. A part of the calculations was conducted in Petričević (2023). We consider datasets with a binary response variable and different features used as predictors to build a predictive model for classification purposes with the application of different classification algorithms. We involve different hyperparameter values/inputs for each classification algorithm and then evaluate the similarity in consensus agreement. We employ three classification algorithms: logistic regression (LR), random forests (RF) and conditional random forests (CF). The logistic regression models were built with four different sets of predictors. Thus, the first set of binary classifiers comprises four logit-based classifiers. The second set of binary classifiers comprises random forest classifiers built under different hyperparameter values: the number of decision trees in the forest and the number of features considered by each tree when splitting a node. For building prediction models, we used the R package *randomForest* (Liaw and Wiener, 2002). The third set of binary classifiers again comprises random forest classifiers, but built under the conditional random forest framework (Strobl *et al.*, 2007, 2008). The resulting forests are unbiased and overcome variable-selection bias which is the major weak spot of the classical approaches. The variable selection bias refers to biasedness in preferring continuous variables and variables with many categories. Again, the models were built under different hyperparameter values: number of decision trees in the forest and the number of features considered by

each tree when splitting a node. The models were built using the R package *party* (Strobl *et al.*, 2009).

We perform the calculations on a balanced dataset and a highly imbalanced dataset by employing two publicly available datasets: *heart disease* (HD) dataset (Janosi *et al.*, 1988) and *stroke prediction* (SP) dataset (Kaggle, 2023). We randomly split both datasets into a train and test dataset where 70% of data are used for training the models. We assess the prediction performance of each classifier on the test dataset by evaluating AUC, accuracy, sensitivity, specificity, precision, and the F1 measure. We need to emphasize that our main goal is not to find the best prediction model, but to construct three collections of predictive models. Thus, some models presented may have lower performance than expected, i.e. by fine-tuning we could find better models. We next describe the results separately for each application.

4.1. *Application I: The HD Dataset*

The HD dataset comprises 13 features and 297 instances (after data cleaning). The goal is to build a predictive model, i.e. a binary classifier, for distinguishing presence (classified as 1) and absence (classified as 0) of heart disease. The binary response variable is balanced with 46% of instances being classified as 1 and 54% being classified as 0.

The first set of binary classifiers comprises four logistic regression models (logit based classifiers). The multivariable logit models built for the heart disease dataset were built in a stepwise fashion with a combination of forward and backward selection. The criterion for feature selection was the AIC. We evaluated 4 models: the full model (*lr_heart1*), and three stepwise models with the three lowest AIC values (*lr_heart2*, *lr_heart3*, *lr_heart3*). These models differ in the predictors included as presented in Table 1. The disease is classified as present if the estimated probability exceeds the threshold of $p = 0.5$.

Table 1
Predictors in logit models (+ = included, − = excluded) for the HD data.

	Logit models			
Predictor	*lr_heart1*	*lr_heart2*	*lr_heart3*	*lr_heart4*
age	+	−	−	−
sex	+	+	+	+
cp	+	+	+	+
trestbps	+	+	+	+
chol	+	+	+	+
fbs	+	−	−	−
restecg	+	−	−	−
thalach	+	−	+	−
exang	+	+	+	−
oldpeak	+	−	−	−
slope	+	+	+	+
ca	+	+	+	+
thal	+	+	+	+

Table 2
Model valuation for the HD data; cv presents the coefficient of variation for a prediction metric over a set of classifiers.

Model			Accuracy	Sensitivity	Specificity	Precision	F1	AUC
LR	*lr_heart1*		0.82	0.85	0.79	0.79	0.82	0.82
	lr_heart2		0.80	0.85	0.74	0.76	0.80	0.80
	lr_heart3		0.81	0.85	0.77	0.77	0.81	0.81
	lr_heart4		0.80	0.83	0.77	0.77	0.80	0.80
	cv (%)		1.24	1.28	2.14	1.68	1.15	1.22
CF	$m = 2$	$n = 50$	0.82	0.88	0.77	0.78	0.82	0.82
		$n = 200$	0.80	0.85	0.74	0.76	0.80	0.80
		$n = 500$	0.80	0.85	0.74	0.76	0.80	0.80
	$m = 4$	$n = 50$	0.81	0.85	0.77	0.77	0.81	0.81
		$n = 200$	0.82	0.85	0.79	0.79	0.82	0.82
		$n = 500$	0.77	0.85	0.70	0.72	0.78	0.77
	$m = 10$	$n = 50$	0.78	0.85	0.72	0.74	0.79	0.79
		$n = 200$	0.78	0.85	0.72	0.74	0.79	0.79
		$n = 500$	0.78	0.85	0.72	0.74	0.79	0.79
	cv (%)		2.02	0.92	3.75	2.75	1.67	1.96
RF	$m = 2$	$n = 50$	0.81	0.90	0.72	0.75	0.82	0.81
		$n = 200$	0.84	0.90	0.79	0.80	0.85	0.85
		$n = 500$	0.83	0.88	0.79	0.80	0.8	0.83
	$m = 4$	$n = 50$	0.84	0.90	0.79	0.80	0.85	0.85
		$n = 200$	0.82	0.90	0.74	0.77	0.83	0.82
		$n = 500$	0.82	0.88	0.77	0.78	0.82	0.82
	$m = 10$	$n = 50$	0.76	0.80	0.72	0.73	0.76	0.76
		$n = 200$	0.77	0.83	0.72	0.73	0.78	0.77
		$n = 500$	0.77	0.83	0.72	0.73	0.78	0.77
	cv (%)		3.79	4.3	4.12	3.7	3.79	3.79

The second set of binary classifiers comprises nine random forest classifiers built under different hyperparameter values: number of decision trees in the forest $n \in \{50, 200, 500\}$, and the number of features considered by each tree when splitting a node $m \in \{2, 4, 10\}$.

The third set of binary classifiers comprises nine random forest classifiers built under the conditional random forest framework with different hyperparameter values: number of decision trees in the forest $n \in \{50, 200, 500\}$, and the number of features considered by each tree when splitting a node $m \in \{2, 4, 10\}$. The performance metrics for the three sets of classifiers are summarized in Table 2.

For each algorithm, we evaluate the within-set similarity in consensus agreement on presence by calculating the k-adic Jaccard coefficient. The results are presented in Table 3 with 95% confidence intervals presented in brackets. The LR based classifiers have the highest level of within-set similarity, while the RF based classifiers have the lowest level of within-set similarity. This means that classifications predicted by LR classifiers are less affected by subjective decisions on the number of predictors compared to the sensitivity of RF and CF classifications when changing the hyperparameter values, all with respect to consensus agreement on the presence. Also, we can conclude that the RF based classifiers are more sensitive to changing hyperparameter values than the CF based classifiers, since the within-set similarity in consensus agreement is lower for the RF based classifiers.

Table 3
Within-set similarity for the HD data, confidence intervals presented in brackets.

Classifier set	*k-adic Jaccard*
LR	0.90 (0.81, 0.99)
CF	0.83 (0.72, 0.95)
RF	0.77 (0.64, 0.89)

Table 4
The between-set similarity in consensus agreement for the HD data.

Classifier set	*CF*	*RF*
LR	0.87 (0.79, 0.95)	0.90 (0.83, 0.97)
CF	1	0.90 (0.82, 0.96)

These results are consistent with the dispersion of the performance measures presented in Table 2. The accuracy, specificity, precision, AUC and the F1 measure have the highest dispersion level for the RF algorithm, and the lowest for the LR algorithm.

The between-set similarity in consensus agreement on presence is calculated by applying the 2-group k-adic similarity coefficient. Results are presented in Table 4 with 95% confidence intervals presented in brackets. The similarity in consensus agreement is similar between pairs of algorithms, with slightly lower similarity between LR and CF. This means that merging classifiers LR and CF would result in less preserved mean within-set similarity in consensus agreement than merging classifiers LR and RF or RF and CF.

4.2. *Application II: The SP Dataset*

The SP dataset comprises 10 features and 4908 instances (after data cleaning). The predictive models are binary classifiers, classifying instances likely to have a stroke as 1 and those unlikely as 0. The dataset is highly imbalanced where 4% of instances are classified as 1 and 96% are classified as 0. We performed a combination of random undersampling and oversampling for the training data for the SP dataset, which resulted in a balanced training set.

We build the prediction models for the SP data following a similar approach to that used for the HD data (see 4.1). The set of the LR based classifiers comprises the full model (*lr_stroke1*), and three stepwise models: the best performing stepwise model constructed with the combination of forward and backward elimination (*lr_stroke2*), the second best model obtained by the stepwise method with backward elimination (*lr_stroke3*), and the second best model obtained by the stepwise method with forward elimination (*lr_stroke4*). These models differ in predictors (features) included, as presented in Table 5. The RF and CF classifiers are also built in the same fashion as for the HD dataset (see Appendix 4.1). The only difference is in the hyperparameter values related to the number of features considered by each tree when splitting a node $m \in \{2, 4, 8\}$. The performance metrics for evaluated classifiers are summarized in Table 6.

Table 5
Predictors in logit models (+ = included, − = excluded) for the SP dataset.

	Model			
Predictor	*lr_stroke1*	*lr_stroke2*	*lr_stroke3*	*lr_stroke4*
gender	+	−	+	−
age	+	+	+	+
hypertension	+	+	+	+
heart_disease	+	−	−	−
ever_married	+	−	−	−
work_type	+	+	+	+
Residence_type	+	−	−	−
avg_glucose_level	+	+	+	+
bmi	+	+	+	−
smoking_status	+	+	+	+

Table 6
Model valuation for the SP data.

Model			Accuracy	Sensitivity	Specificity	Precision	F1	AUC
LR	*lr_stroke1*		0.74	0.74	0.75	0.99	0.85	0.75
	lr_stroke2		0.75	0.75	0.77	0.99	0.85	0.76
	lr_stroke3		0.74	0.74	0.74	0.99	0.85	0.74
	lr_stroke4		0.75	0.75	0.75	0.99	0.85	0.75
	cv (%)		0.37	0.34	1.65	0.07	0.21	0.96
CF	$90m = 2$	$n = 50$	0.79	0.80	0.65	0.98	0.88	0.72
		$n = 200$	0.79	0.80	0.68	0.98	0.88	0.74
		$n = 500$	0.79	0.80	0.68	0.98	0.88	0.74
	$m = 4$	$n = 50$	0.82	0.83	0.63	0.98	0.90	0.73
		$n = 200$	0.83	0.83	0.67	0.98	0.90	0.75
		$n = 500$	0.82	0.83	0.67	0.98	0.90	0.75
	$m = 8$	$n = 50$	0.83	0.84	0.58	0.98	0.91	0.71
		$n = 200$	0.83	0.84	0.60	0.98	0.90	0.72
		$n = 500$	0.83	0.84	0.58	0.98	0.90	0.71
	cv (%)		1.93	2.11	6.36	0.17	1.11	1.91
RF	$m = 2$	$n = 50$	0.86	0.88	0.40	0.97	0.92	0.64
		$n = 200$	0.86	0.88	0.44	0.98	0.92	0.66
		$n = 500$	0.86	0.88	0.46	0.98	0.92	0.67
	$m = 4$	$n = 50$	0.92	0.95	0.23	0.97	0.96	0.59
		$n = 200$	0.92	0.950	0.26	0.97	0.96	0.61
		$n = 500$	0.92	0.95	0.25	0.97	0.96	0.60
	$m = 8$	$n = 50$	0.91	0.94	0.19	0.97	0.95	0.57
		$n = 200$	0.91	0.94	0.23	0.97	0.96	0.58
		$n = 500$	0.91	0.94	0.23	0.97	0.95	0.58
	cv (%)		3.06	3.52	30.63	0.33	1.52	5.43

We evaluate the within-set similarity in consensus agreement on presence by calculating the k-adic Jaccard coefficient. The results are presented in Table 7. As in the case of the HD data, the LR based classifiers have the highest level of within-set similarity, while the RF based classifiers have the lowest level of within-set similarity. Again, this means that classifications predicted by LR based classifiers are less affected by subjective

Table 7
Within-set similarity for the SP data.

Classifier set	*k-adic Jaccard*
LR	0.92 (0.90, 0.95)
CF	0.63 (0.58, 0.68)
RF	0.24 (0.18, 0.29)

Table 8
The between-set similarity in consensus agreement for the SP data.

Classifier set	*CF*	*RF*
LR	0.61 (0.57, 0.65)	0.22 (0.18, 0.27)
CF	1	0.36 (0.29, 0.41)

decisions on the number of predictors compared to the sensitivity of RF and CF classifications when changing the hyperparameter values, all with respect to consensus agreement on the presence. Also, as in the case of the HD dataset, we can conclude that classifications predicted by RF based classifiers are more sensitive to changing hyperparameter values than classifications predicted by the CF based classifiers, since the within-set similarity in consensus agreement is lower for the RF based classifiers. Compared to the HD dataset case, in the case of the SP data, the RF and CF algorithms exhibit notably higher sensitivity to hyperparameter values (due to the lower values of the similarity coefficient), all in terms of the within-set similarity in consensus agreement. These results are consistent with the dispersion of the performance measures presented in Table 6. All performance measures have the highest dispersion level for the RF algorithm, and the lowest for the LR algorithm.

The between-set similarity in consensus agreement on presence is calculated by applying the 2-group k-adic similarity coefficient, and the results are presented in Table 8 with 95% confidence intervals in brackets. LR-based and CF-based algorithms have the highest level of similarity in consensus agreement on presence, while the LR-based algorithm and the RF-based algorithm have the lowest level of similarity in consensus agreement on presence.

5. Conclusion and Discussion

Binary classification is a fundamental task in machine learning, with broad applications across various scientific and practical fields. A vast and growing number of developed binary classification algorithms supports the importance and encourages the application of binary classification methods. The extensive collection of developed algorithms challenges researchers to develop methods to find the best-suiting algorithm for a particular problem and the dataset of interest. This typically involves comparing the predictive performance of binary classification models.

Building an effective prediction model typically requires hyperparameter tuning, meaning that when evaluating a single binary classification algorithm with different hyperparameter values, we are effectively assessing a collection of classifiers.

This research makes a significant contribution to the comparison of binary classification algorithms. Instead of focusing on prediction performances, it compares the resulting classifications. The core of our approach lies in considering the variability introduced by different hyperparameter values for each algorithm when performing such comparisons. This means that, rather than selecting a single representative classifier for each binary classification algorithm (typically the one with the best prediction performance), we consider a set of classifiers as being a representative set of a binary classification algorithm. This approach accounts for the variability of that set.

The comparison of two binary classification algorithms is performed by applying the 2-group k-adic similarity coefficient. This approach is based on the k-adic similarity approach, or the De Moivre definition of agreement. The variability within each classifier set is quantified by calculating the k-adic similarity coefficients. Specifically, we apply the k-adic similarity coefficient to assess the within-set similarity, which quantifies the similarity of k classifiers related to a particular classification algorithm. Furthermore, when classifiers are built under a certain classification algorithm by varying hyperparameter values and subjective inputs, the k-adic similarity coefficient can be used to assess the sensitivity of an algorithm to changes in these values.

The 2-group k-adic Jaccard coefficient and the k-adic Jaccard coefficient have a broad potential of application which has yet to be explored. For instance, it could be of interest to evaluate the within-set similarity of a classifier set of a classification across different hyperparameter spaces. This can be performed by calculating the k-adic Jaccard coefficient for different $\mathcal{H}$, and could be used as a measure of robustness. In future research, we plan to relax the strict consensus-based approach of the k-adic similarity coefficient. While this method evaluates the agreement between classifiers, we plan to develop similarity coefficients that can measure partial agreement between two sets of classifiers.

The proposed methodology applies the asymmetric similarity coefficients. However, when the goal of the analysis is to take into account both agreement on presence and agreement on absence, we involve the symmetric coefficients, like the simple matching coefficient. A version of this coefficient that can be applied for quantifying binary classifier algorithms similarity can be found in Perišić and Vanbelle (2024).

The presented methodology has several limitations and raises important open questions. The first one relates to the selection of hyperparameter values. The hyperparameter space is multidimensional, with each dimension representing a specific hyperparameter. The same complexity applies for the space related to subjective input values. Key questions to address include:

- How many distinct hyperparameters should be chosen when forming a set of binary classifiers for some classification algorithm?
- What criteria should guide the selection of hyperparameter values?
- Is it justified to compare the sensitivity to hyperparameters when different hyperparameters or subjective inputs are considered for each algorithm?

For example, in the application described in Section 4, logistic regression was found to be less sensitive to variations in hyperparameters and inputs, while random forest was more sensitive. Logistic regression classifiers were created by selecting different predictors during model building, whereas random forests were formed by altering parameters such as the number of trees in the forest and the number of features considered at each split. This suggests that it may not always be appropriate to consider the same hyperparameter dimensions across algorithms. Thus, comparing algorithm sensitivity to changing hyperparameter values will sometimes imply evaluating different hyperparameter dimensions for each classification algorithm. However, we should be aware that results on algorithm sensitivity could be affected by selected hyperparameter dimensions.

Another limitation arises when comparing classifier sets of different sizes. If we assess the within-set similarity of a collection of classifiers derived from various hyperparameter settings for a given algorithm, increasing the number of classifiers in the set (i.e. evaluating a broader range of hyperparameter values) could increase the heterogeneity within the set. This is evident in the examples provided in Appendix A. Consequently, the selection of both the number of classifiers and the specific hyperparameter values should be approached with care, particularly if no predefined rule dictates which hyperparameters are of interest.

A. Appendix: On the Number of Evaluated Classifiers

In the application part presented in Section 4, the number of classifiers related to LR algorithm (i.e. 4) is much smaller than the number of classifiers related to RF and CF (i.e. 9). This can affect the value of the similarity coefficient. We analyse this impact by taking samples of 4 classifiers from the CF and RF based classifier sets separately, and evaluating the within- and the between-set similarity.

A.1. *HD Dataset*

The mean value of the 126 k-adic Jaccard coefficients obtained within sets of 4 classifiers is presented in Table 9. These means are slightly higher than when considering the set of 9 classifiers.

We further evaluate the between-set similarities when taking subsets of 4 classifiers from the RF and CF classifier sets separately (i.e. 126 sets for each algorithm) and calculate the mean of the 2-group k-adic Jaccard similarity coefficients (see Table 10). The mean

Table 9
Within-set similarity for the HD data, mean values of the 4-classifier subsets, the range $(x_{\min}, x_{\max})$ presented in brackets.

Classifier set	*k-adic Jaccard*
CF	0.85 (0.83, 0.93)
RF	0.81 (0.77, 0.90)

Table 10
The between-set similarity in consensus agreement for the HD data-mean values of the 4-classifier subsets, the range ($x_{\min}, x_{\max}$) presented in brackets.

Classifier set	*CF*	*RF*
LR	0.90 (0.86, 0.96)	0.90 (0.87, 0.93)
CF	1	0.89 (0.84, 0.95)

Table 11
Within-set similarity for the SP data, mean values of the 4-classifier subsets, the range ($x_{\min}, x_{\max}$) presented in brackets.

Classifier set	*k-adic Jaccard*
CF	0.71 (0.65, 0.84)
RF	0.34 (0.25, 0.61)

Table 12
The between-set similarity in consensus agreement for the SP data-mean values of the 4-classifier subsets, the range ($x_{\min}, x_{\max}$) presented in brackets.

Classifier set	*CF*	*RF*
LR	0.81 (0.69, 0.89)	0.24 (0.19, 0.30)
CF	1	0.37 (0.26, 0.46)

coefficient values slightly differ from the case when the 9 classifiers were considered at the same time.

A.2. *SP Dataset*

For the SP dataset, when considering sets of 4 classifiers, the mean value of the 126 k-adic Jaccard coefficients (see Table 11) is higher than when considering the set of 9 classifiers. Here also, we evaluate the between-set similarity when taking subsets of 4 classifiers from the RF and CF classifier sets (126 sets for each algorithm) and calculate the mean of the 2-group k-adic Jaccard similarity coefficients (see Table 12). All mean coefficient values are higher than in the case with 9 classifiers. As in the case of 9 classifiers, in the case of evaluating the mean values of the 4-classifier subsets, RF has the lowest within-set similarity. Also, the lowest between-set similarity is again related to LR and RF, while the highest between-set similarity is related to LR and CF.

References

Benavoli, A., Corani, G., Mangili, F. (2016). Should we really use post-hoc tests based on mean-ranks? *The Journal of Machine Learning Research*, 17(1), 152–161.

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A.-L., Deng, D., Lindauer, M. (2023). Hyperparameter optimization: foundations, algorithms, best practices, and open challenges. *WIREs Data Mining and Knowledge Discovery*, 13(2), e1484. https://doi.org/10.1002/widm.1484.

Chicco, D., Warrens, M.J., Jurman, G. (2021). The Matthews Correlation Coefficient (MCC) is more informative than Cohen's Kappa and brier score in binary classification assessment. *IEEE Access*, 9, 78368–78381. https://doi.org/10.1109/ACCESS.2021.3084050.

Choi, S., Cha, S., Tappert, C.C. (2010). A survey of binary similarity and distance measures. *Journal of Systemics, Cybernetics and Informatics*, 8(1), 43–48.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7, 1–30.

Feurer, M., Hutter, F. (2019). Hyperparameter optimization. In: Hutter, F., Kotthoff, L., Vanschoren, J. (Eds.), *Automated Machine Learning: Methods, Systems, Challenges*. Springer International Publishing, pp. 3–33. https://doi.org/10.1007/978-3-030-05318-5_1.

Han, J., Kamber, M., Pei, J. (2012). 8 – Classification: Basic concepts. In: Han, J., Kamber, M., Pei, J. (Eds.), *Data Mining* (third edition). Morgan Kaufmann, pp. 327–391. https://doi.org/10.1016/B978-0-12-381479-1.00008-3.

Hubert, L. (1977). Kappa revisited. *Psychological Bulletin*, 84(2), 289–297. https://doi.org/10.1037/0033-2909.84.2.289.

Janosi, A., Steinbrunn, W., Pfisterer, M., Detrano, R. (1988). Heart Disease [Dataset]. *UCI Machine Learning Repository*. https://doi.org/10.24432/C52P4X.

Kaggle (2023). Stroke prediction dataset [data retrieved from Kaggle]. https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset.

Kuncheva, L.I., Whitaker, C.J. (2003). Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2), 181–207. https://doi.org/10.1023/A:1022859003006.

Labatut, V., Cherifi, H. (2012). Accuracy measures for the comparison of classifiers. arXiv preprint. arXiv:1207.3790.

Liaw, A., Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3), 18–22. https://CRAN.R-project.org/doc/Rnews/.

Makhtar, M., Neagu, D.C., Ridley, M.J. (2011). Binary classification models comparison: on the similarity of datasets and confusion matrix for predictive toxicology applications. In: Böhm, C., Khuri, S., Lhotská, L., Pisanti, N. (Eds.), *Information Technology in Bio- and Medical Informatics*. Springer, Berlin Heidelberg, pp. 108–122.

Margineantu, D.D., Dietterich, T.G. (1997). Pruning adaptive boosting. *ICML*, 97, 211–218.

Narasimhamurthy, A. (2005). Evaluation of diversity measures for binary classifier ensembles. In: Oza, N.C., Polikar, R., Kittler, J., Roli, F. (Eds.), *Multiple Classifier Systems*. Springer, Berlin Heidelberg, pp. 267–277.

Perišić, A., Vanbelle, S. (2024). Two-group k-adic similarity coefficients for binary classifiers. *Journal of Classification*, 41(2), 325–345. https://doi.org/10.1007/s00357-024-09498-8.

Petrakos, M., Atli Benediktsson, J., Kanellopoulos, I. (2001). The effect of classifier agreement on the accuracy of the combined classifier in decision level fusion. *IEEE Transactions on Geoscience and Remote Sensing*, 39(11), 2539–2546. https://doi.org/10.1109/36.964992.

Petričević, R.B. (2023). *Primjena modificiranog k-adskog jaccardovog koeficijenta sličnosti za usporedbu dvaju skupova binarnih klasifikatora*. Master's thesis, University of Split, Faculty of Science in Split.

Shirdel, M., Di Mauro, M., Liotta, A. (2024). Worthiness benchmark: a novel concept for analyzing binary classification evaluation metrics. *Information Sciences*, 678, 120882. https://doi.org/10.1016/j.ins.2024.120882.

Sokal, R.R., Sneath, P.H.A. (1963). *Principles of Numerical Taxonomy*. W. H. Freeman; Company.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., Zeileis, A. (2008). Conditional variable importance for random forests. *BMC Bioinformatics*, 9 307. https://doi.org/10.1186/1471-2105-9-307.

Strobl, C., Boulesteix, A.-L., Zeileis, A., Hothorn, T. (2007). Bias in random forest variable importance measures: illustrations, sources and a solution. *BMC Bioinformatics*, 8, 25. https://doi.org/10.1186/1471-2105-8-25.

Strobl, C., Hothorn, T., Zeileis, A. (2009). Party on! *The R Journal*, 1(2), 14–17. https://doi.org/10.32614/RJ-2009-013.

Tang, E.K., Suganthan, P.N., Yao, X. (2006). An analysis of diversity measures. *Machine Learning*, 65(1), 247–271. https://doi.org/10.1007/s10994-006-9449-2.

Tsymbal, A., Pechenizkiy, M., Cunningham, P. (2005). Diversity in search strategies for ensemble feature selection. *Information Fusion*, 6(1), 83–98.

Vanbelle, S., Albert, A. (2009). Agreement between two independent groups of raters. *Psychometrika*, 74, 477–491. https://doi.org/10.1007/S11336-009-9116-1.

Wang, J., Wang, L., Zheng, Y., Yeh, C.-C.M., Jain, S., Zhang, W. (2023). Learning-from-disagreement: a model comparison and visual analytics framework. *IEEE Transactions on Visualization & Computer Graphics*, 29(09), 3809–3825. https://doi.org/10.1109/TVCG.2022.3172107.

Warrens, M.J. (2009). k-adic similarity coefficients for binary (presence/absence) data. *Journal of Classification*, 26, 227–245. https://doi.org/10.1007/s00357-009-9032-1.

Wood, D., Mu, T., Webb, A.M., Reeve, H.W., Lujan, M., Brown, G. (2023). A unified theory of diversity in ensemble learning. *Journal of Machine Learning Research*, 24(359), 1–49.

Zouari, H., Heutte, L., Lecourtier, Y. (2005). Using diversity measure in building classifier ensembles for combination method analysis. In: Kurzyński, M., Puchała, E., Woźniak, M., żołnierek, A. (Eds.), *Computer Recognition Systems*. Springer, Berlin Heidelberg, pp. 337–344.

A. Perišić received her PhD at the University of Ljubljana (Slovenia) in 2021, after completing her master's degree in mathematics and postgraduate studies in Economics at University of Zagreb (Croatia). Currently she is working at the Šibenik University of Applied Sciences as a college professor, and at the Department of Mathematics, University of Split as a postdoctoral researcher. Her research interests span a wide range of topics in the development and application of statistical methodologies, with significant contributions to modeling for customer churn prediction, clustering of mixed-type data, developing composite indicators and the development of similarity coefficients for binary data sets. She has contributed to a number of journal articles and conference papers, and participated in diverse research projects, including industry-academia collaborations.

S. Vanbelle completed a master's degree in mathematics (ULiègge, Belgium) and a master's degree in Biostatistics (UHasselt, Belgium) before obtaining her PhD at ULiège in 2009. She is currently associate professor in the department of Methodology & Statistics at the faculty of Health, Medicine and Life Sciences, Maastricht University, The Netherlands. Her research focuses on the development and application of statistical methodology for reliability and agreement studies, with particular interest in complex and multilevel designs and intensive longitudinal data. She has authored numerous peer-reviewed articles and contributed to tutorials and reviews. In addition, she is actively engaged in the statistical community, including service within the Belgian Region of the International Biometric Society.

R.B. Petričević graduated in mathematics from the Faculty of Science, University of Split, in 2023. As part of her master's thesis, she worked on quantifying binary classifier algorithms similarity with a consensus agreement approach. She is currently employed at OTP Bank as a Data Warehouse Specialist, where she works on the implementation of a new data warehouse. Her responsibilities also include developing regulatory reports for the Croatian National Bank and the European Central Bank, as well as supporting data-driven decision-making within the bank.