dtComb: A Comprehensive R Library and Web Tool for Combining Diagnostic Tests

by S. Ilayda Yerlitaş Taştan, Serra Bersan Gengeç, Necla Koçhan, Gözde Ertürk Zararsız, Selcuk Korkmaz, and Gökmen Zararsız

Abstract The combination of diagnostic tests has become a crucial area of research, aiming to improve the accuracy and robustness of medical diagnostics. While existing tools focus primarily on linear combination methods, there is a lack of comprehensive tools that integrate diverse methodologies. In this study, we present dtComb, a comprehensive R package and web tool designed to address the limitations of existing diagnostic test combination platforms. One of the unique contributions of dtComb is offering a range of 142 methods to combine two diagnostic tests, including linear, non-linear, machine learning algorithms, and mathematical operators. Another significant contribution of dtComb is its inclusion of advanced tools for ROC analysis, diagnostic performance metrics, and visual outputs such as sensitivity-specificity curves. Furthermore, dtComb offers classification functions for new observations, making it an easy-to-use tool for clinicians and researchers. The web-based version is also available for non-R users, providing an intuitive interface for test combination and model training.

1 Introduction

A typical scenario often encountered in combining diagnostic tests is when the gold standard method combines two-category and two continuous diagnostic tests. In such cases, clinicians usually seek to compare these two diagnostic tests and improve the performance of these diagnostic test results by dividing the results into proportional results (Müller et al., 2019; Faria et al., 2016; Nyblom et al., 2006). However, this technique is straightforward and may not fully capture all potential interactions and relationships between the diagnostic tests. Linear combination methods have been developed to overcome such problems (Ertürk Zararsız, 2023).

Linear methods combine two diagnostic tests into a single score/index by assigning weights to each test, optimizing their performance in diagnosing the condition of interest (Neumann et al., 2023). Such methods improve accuracy by leveraging the strengths of both tests (Aznar-Gimeno et al., 2022; Bansal and Sullivan Pepe, 2013). For instance, Su and Liu (1993) found that Fisher's linear discriminant function generates a linear combination of markers with either proportional or disproportional covariance matrices, aiming to maximize sensitivity consistently across the entire selectivity spectrum under a multivariate normal distribution model. In contrast, another approach introduced by Pepe and Thomson (Pepe and Thompson, 2000) relies on ranking scores, eliminating the need for linear distributional assumptions when combining diagnostic tests. Despite the theoretical advances, when existing tools were examined, it was seen that they contained a limited number of methods. For instance, Kramar et al. developed a computer program called `mROC` that includes only the Su and Liu method (Kramar et al., 2001). Pérez-Fernández et al. presented a **movieROC** R package that includes methods such as Su and Liu, min-max, and logistic regression methods (Pérez-Fernández et al., 2021). An R package called **maxmzpAUC** that includes similar methods was developed by Yu and Park (Yu and Park, 2015).

On the other hand, non-linear approaches incorporating the non-linearity between the diagnostic tests have been developed and employed to integrate the diagnostic tests (Du et al., 2024; Ghosh and Chinnaiyan, 2005). These approaches incorporate the non-linear structure of tests into the model, which might improve the accuracy and reliability of the diagnosis. In contrast to some existing packages, which permit the use of non-linear approaches such as `splines`, `lasso`, and `ridge` regression, there is currently no package that employs these methods directly for combination and offers diagnostic performance. Machine-learning (ML) algorithms have recently been adopted to combine diagnostic tests

(Ahsan et al., 2024; Sewak et al., 2024; Agarwal et al., 2023; Prinzi et al., 2023). Many publications/studies focus on implementing ML algorithms in diagnostic tests (Salvetat et al., 2022, 2024; Ganapathy et al., 2023; Alzyoud et al., 2024; Zararsiz et al., 2016). For instance, DeGroat et al. performed four different classification algorithms (Random Forest, Support Vector Machine, Extreme Gradient Boosting Decision Trees, and k-Nearest Neighbors) to combine markers for the diagnosis of cardiovascular disease (DeGroat et al., 2024). The results showed that patients with cardiovascular disease can be diagnosed with up to 96% accuracy using these ML techniques. There are numerous applications where ML methods can be implemented `scikit-learn` (Pedregosa et al., 2011), `TensorFlow` (Abadi et al., 2015), **caret** (Kuhn, 2008)). The **caret** library is one of the most comprehensive tools developed in the R language (Kuhn, 2008). However, these are general tools developed only for ML algorithms and do not directly combine two diagnostic tests and provide diagnostic performance measures.

Apart from the aforementioned methods, several basic mathematical operations such as addition, multiplication, subtraction, and division can also be used to combine markers (Svart et al., 2024; Luo et al., 2024; Serban et al., 2024). For instance, addition can enhance diagnostic sensitivity by combining the effects of markers, whereas subtraction can more distinctly differentiate disease states by illustrating the variance across markers. On the other hand, there are several commercial (e.g. IBM SPSS, MedCalc, Stata, etc.) and open source R software packages (**ROCR** (Sing et al., 2005), **pROC** (Robin et al., 2011), **PRROC** (Grau et al., 2015), **plotROC** (Sachs, 2017)) that researchers can use for Receiver operating characteristic (ROC) curve analysis. However, these tools are designed to perform a single marker ROC analysis. As a result, there is currently no software tool that covers almost all combination methods.

In this study, we developed **dtComb**, an R package encompassing nearly all existing combination approaches in the literature. **dtComb** has two key advantages, making it easy to apply and superior to the other packages: (1) it provides users with a comprehensive 142 methods, including linear and non-linear approaches, ML approaches, and mathematical operators; (2) it produces turnkey solutions to users from the stage of uploading data to the stage of performing analyses, performance evaluation, and reporting. Furthermore, it is the only package that illustrates linear approaches such as Minimax and Todor & Saplacan (Sameera et al., 2016; Todor et al., 2014). In addition, it allows for the classification of new, previously unseen observations using trained models. To our knowledge, no other tools were designed and developed to combine two diagnostic tests on a single platform with 142 different methods. In other words, **dtComb** has made more effective and robust combination methods ready for application instead of traditional approaches such as simple ratio-based methods. First, we review the theoretical basis of the related combination methods; then, we present an example implementation to demonstrate the applicability of the package. Finally, we present a user-friendly, up-to-date, and comprehensive web tool developed to facilitate **dtComb** for physicians and healthcare professionals who do not use the R programming language. The **dtComb** package is freely available on the CRAN network, the web application is freely available at `https://biotools.erciyes.edu.tr/dtComb/`, and all source code is available on GitHub (`https://github.com/gokmenzararsiz/dtComb`, `https://github.com/gokmenzararsiz/dtComb_Shiny`).

2 Material and methods

This section will provide an overview of the combination methods implemented in the literature. Before applying these methods, we will also discuss the standardization techniques available for the markers, the resampling methods during model training, and, ultimately, the metrics used to evaluate the model's performance.

2.1 Combination approaches

Linear combination methods

The **dtComb** package comprises eight distinct linear combination methods, which will be elaborated in this section. Before investigating these methods, we briefly introduce some notations which will be used throughout this section.

Notations:

Let D_i, $i = 1, 2, \ldots, n_1$ be the marker values of the ith individual in the diseased group, where $D_i = (D_{i1}, D_{i2})$ and H_j, $j = 1, 2, \ldots, n_2$ be the marker values of the jth individual in the healthy group, where $H_j = (H_{j1}, H_{j2})$. Let $x_{i1} = c(D_{i1}, H_{j1})$ be the values of the first marker, and $x_{i2} = c(D_{i2}, H_{j2})$ be values of the second marker for the ith individual ($i = 1, 2, \ldots, n$). Let $D_{i,\min} = \min(D_{i1}, D_{i2})$, $D_{i,\max} = \max(D_{i1}, D_{i2})$ $H_{j,\min} = \min(H_{j1}, H_{j2})$, $H_{j,\max} = \max(H_{j1}, H_{j2})$ and c_i be the resulting combination score of the ith individual.

- *Logistic regression:*

  Logistic regression is a statistical method used for binary classification. The logistic regression model estimates the probability of the binary outcome occurring based on the values of the independent variables. It is one of the most commonly applied methods in diagnostic tests, and it generates a linear combination of markers that can distinguish between control and diseased individuals. Logistic regression is generally less effective than normal-based discriminant analysis, like Su and Liu's multivariate normality-based method, when the normal assumption is met (Ruiz-Velasco, 1991; Efron, 1975). On the other hand, others have argued that logistic regression is more robust because it does not require any assumptions about the joint distribution of multiple markers (Cox and Snell, 1989). Therefore, it is essential to investigate the performance of linear combination methods derived from the logistic regression approach with non-normally distributed data.

  The objective of the logistic regression model is to maximize the logistic likelihood function. In other words, the logistic likelihood function is maximized to estimate the logistic regression model coefficients.

$$c_i = \frac{\exp(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2})}{1 + \exp(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2})}. \tag{1}$$

The logistic regression coefficients can provide the maximum likelihood estimation of the model, producing an easily interpretable value for distinguishing between the two groups.

- *Scoring based on logistic regression:*

  The method primarily uses a binary logistic regression model, with slight modifications to enhance the combination score. The regression coefficients, as predicted in Eq. (1), are rounded to a user-specified number of decimal places and subsequently used to calculate the combination score (León et al., 2006).

$$c = \beta_1 x_{i1} + \beta_2 x_{i2}. \tag{2}$$

- *Pepe & Thompson's method:* Pepe & Thompson have aimed to maximize the AUC or partial AUC to combine diagnostic tests, regardless of the distribution of markers (Pepe and Thompson, 2000). They developed an empirical solution of optimal linear combinations that maximize the Mann-Whitney U statistic, an empirical estimate of the ROC curve. Notably, this approach is distribution-free. Mathematically, they maximized the following objective function:

$$\text{maximize } U(\alpha) = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} I(D_{i1} + \alpha D_{i2} \geq H_{j1} + \alpha H_{j2}) \tag{3}$$

$$c = x_{i1} + \alpha x_{i2} \tag{4}$$

where $\alpha \in [-1, 1]$ is interpreted as the relative weight of x_{i2} to x_{i1} in the combination, the weight of the second marker. This formula aims to find α to maximize $U(a)$. Readers are referred to see (Pepe and Thomson) (Pepe and Thompson, 2000).

- *Pepe, Cai & Langton's method:* Pepe et al. observed that when the disease status and the levels of markers conform to a generalized linear model, the regression coefficients represent the optimal linear combinations that maximize the area under the ROC curves (Pepe et al., 2006). The following objective function is maximized to achieve a higher AUC value:

$$\text{maximize } U(\alpha) = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \Big(I[D_{i1} + \alpha D_{i2} > H_{j1} + \alpha H_{j2}] + \tfrac{1}{2} I[D_{i1} + \alpha D_{i2} = H_{j1} + \alpha H_{j2}] \Big). \tag{5}$$

Before calculating the combination score using Eq. (4), the marker values are normalized or scaled to be constrained within the scale of 0 to 1. In addition, it is noted that the estimate obtained by maximizing the empirical AUC can be considered as a particular case of the maximum rank correlation estimator from which the general asymptotic distribution theory was developed. Readers are referred to Pepe (2003, Chapters 4–6) for a review of the ROC curve approach and more details (Pepe, 2003).

- *Min-Max method:* The Pepe & Thomson method is straightforward if there are two markers. It is computationally challenging if we have more than two markers to be combined. To overcome the computational complexity issue of this method, Liu et al. (Liu et al., 2011) proposed a non-parametric approach that linearly combines the minimum and maximum values of the observed markers of each subject. This approach, which does not rely on the normality assumption of data distributions (i.e., distribution-free), is known as the Min-Max method and may provide higher sensitivity than any single marker. The objective function of the Min-Max method is as follows:

$$\text{maximize } U(\alpha) = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} I[D_{i,\max} + \alpha D_{i,\min} > H_{j,\max} + \alpha H_{j,\min}] \tag{6}$$

$$c = x_{i,\max} + \alpha x_{i,\min} \tag{7}$$

where $x_{i,\max} = \max(x_{i1}, x_{i2})$ and $x_{i,\min} = \min(x_{i1}, x_{i2})$.

The Min-Max method aims to combine repeated measurements of a single marker over time or multiple markers that are measured with the same unit. While the Min-Max method is relatively simple to implement, it has some limitations. For example, markers may have different units of measurement, so standardization can be needed to ensure uniformity during the combination process. Furthermore, it is unclear whether all available information is fully utilized when combining markers, as this method incorporates only the markers' minimum and maximum values into the model (Kang et al., 2016).

- *Su & Liu's method:* Su and Liu examined the combination score separately under the assumption of two multivariate normal distributions when the covariance matrices were proportional or disproportionate (Su and Liu, 1993). Multivariate normal distributions with different covariances were first utilized in classification problems (Anderson and Bahadur, 1962). Then, Su and Liu also developed a linear combination method by extending the idea of using multivariate distributions to the AUC, showing that the best

coefficients that maximize AUC are Fisher's discriminant coefficients. Assuming that $D \sim N(\mu_D, \Sigma_D)$ and $H \sim N(\mu_H, \Sigma_H)$ represent the multivariate normal distributions for the diseased and non-diseased groups, respectively. The Fisher's coefficients are as follows:

$$(\alpha, \beta) = (\Sigma_D + \Sigma_H)^{-1}\mu \tag{8}$$

where $\mu = \mu_D - \mu_H$. The combination score in this case is:

$$c = \alpha x_{i1} + \beta x_{i2}. \tag{9}$$

- *The Minimax method:* The Minimax method is an extension of Su & Liu's method (Sameera et al., 2016). Suppose that D follows a multivariate normal distribution $D \sim N(\mu_D, \Sigma_D)$, representing the diseased group, and H follows a multivariate normal distribution $H \sim N(\mu_H, \Sigma_H)$, representing the non-diseased group. Then Fisher's coefficients are as follows:

$$(\alpha, \beta) = [t\Sigma_D + (1-t)\Sigma_H]^{-1}(\mu_D - \mu_H). \tag{10}$$

  Given these coefficients, the combination score is calculated using Eq. (9). In this formula, t is a constant with values ranging from 0 to 1. This value can be hyper-tuned by maximizing the AUC.

- *Todor & Saplacan's method:* Todor and Saplacan's method uses the sine and cosine trigonometric functions to calculate the combination score (Todor et al., 2014). The combination score is calculated using $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, which maximizes the AUC within this interval. The formula for the combination score is given as follows:

$$c = \sin(\theta)x_{i1} + \cos(\theta)x_{i2}. \tag{11}$$

Non-linear combination methods

In addition to linear combination methods, the **dtComb** package includes seven non-linear approaches, which will be discussed in this subsection. In this subsection, we will use the following notations: x_{ij}: the value of the jth marker for the ith individual, $i = 1, 2, \ldots, n$, $j = 1, 2$, d: degree of polynomial regressions and splines, $d = 1, 2, \ldots, p$.

- *Logistic Regression with Polynomial Feature Space:* This approach extends the logistic regression model by adding extra predictors created by raising the original predictor variables to a certain power. This transformation enables the model to capture and model non-linear relationships in the data by including polynomial terms in the feature space (James et al., 2013). The combination score is calculated as follows:

$$c = \frac{\exp\left(\beta_0 + \beta_1 x_{ij} + \beta_2 x_{ij}^2 + \cdots + \beta_p x_{ij}^p\right)}{1 + \exp\left(\beta_0 + \beta_1 x_{ij} + \beta_2 x_{ij}^2 + \cdots + \beta_p x_{ij}^p\right)} \tag{12}$$

where c_i is the combination score for the ith individual and represents the posterior probabilities.

- *Ridge Regression with Polynomial Feature Space:* This method combines Ridge regression with expanded feature space created by adding polynomial terms to the original predictor variables. It is a widely used shrinkage method when we have multicollinearity between the variables, which may be an issue for least squares regression. This method aims to estimate the coefficients of these correlated variables by minimizing the residual sum of squares (RSS) while adding a term (referred to as a regularization term) to

prevent overfitting. The objective function is based on the L2 norm of the coefficient vector, which prevents overfitting in the model (Eq. (13)). The Ridge estimate is defined as follows:

$$\hat{\beta}^R = \underset{\beta}{\operatorname{argmin}} \; RSS + \lambda \sum_{j=1}^{2} \sum_{d=1}^{p} \beta_j^{d^2} \tag{13}$$

where

$$RSS = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{2} \sum_{d=1}^{p} \beta_j^d x_{ij}^d \right)^2 \tag{14}$$

and $\hat{\beta}^R$ denotes the estimates of the coefficients of the Ridge regression, and the second term is called a penalty term where $\lambda \geq 0$ is a shrinkage parameter. The shrinkage parameter, λ, controls the amount of shrinkage applied to regression coefficients. A cross-validation is implemented to find the shrinkage parameter. We used the **glmnet** package (Friedman et al., 2010) to implement the Ridge regression in combining the diagnostic tests.

- *Lasso Regression with Polynomial Feature Space:* Similar to Ridge regression, Lasso regression is also a shrinkage method that adds a penalty term to the objective function of the least square regression. The objective function, in this case, is based on the L1 norm of the coefficient vector, which leads to the sparsity in the model. Some of the regression coefficients are precisely zero when the tuning parameter λ is sufficiently large. This property of the Lasso method allows the model to automatically identify and remove less relevant variables and reduce the algorithm's complexity. The Lasso estimates are defined as follows:

$$\hat{\beta}^L = \underset{\beta}{\operatorname{argmin}} \; RSS + \lambda \sum_{j=1}^{2} \sum_{d=1}^{p} |\beta_j^d|. \tag{15}$$

  To implement the Lasso regression in combining the diagnostic tests, we used the **glmnet** package (Friedman et al., 2010).

- *Elastic-Net Regression with Polynomial Feature Space:* Elastic-Net Regression is a method that combines Lasso (L1 regularization) and Ridge (L2 regularization) penalties to address some of the limitations of each technique. The combination of the two penalties is controlled by two hyperparameters, $\alpha \in [0, 1]$ and λ, which enable you to adjust the trade-off between the L1 and L2 regularization terms (James et al., 2013). For the implementation of the method, the **glmnet** package is used (Friedman et al., 2010).

- *Splines:* Another non-linear combination technique frequently applied in diagnostic tests is the splines. Splines are a versatile mathematical and computational technique that has a wide range of applications. These splines are piecewise functions that make interpolating or approximating data points possible. There are several types of splines, such as cubic splines. Smooth curves are created by approximating a set of control points using cubic polynomial functions. When implementing splines, two critical parameters come into play: degrees of freedom and the choice of polynomial degrees (i.e., degrees of the fitted polynomials). These user-adjustable parameters, which influence the flexibility and smoothness of the resulting curve, are critical for controlling the behavior of splines. We used the splines package in the R programming language to implement splines.

- *Generalized Additive Models with Smoothing Splines and Generalized Additive Models with Natural Cubic Splines:* Regression models are of great interest in many fields to understand the importance of different inputs. Even though regression is widely used, the traditional linear models often fail in real life as effects may not be linear.

Another method called generalized additive models was introduced to identify and characterize non-linear regression (James et al., 2013). Smoothing Splines and Natural Cubic Splines are two standard methods used within GAMs to model non-linear relationships. To implement these two methods, we used the **gam** package in R (Hastie, 2025). The method of GAMs with Smoothing Splines is a more data-driven and adaptive approach where smoothing splines can automatically capture non-linear relationships without specifying the number of knots (specific points where two or more polynomial segments are joined together to create a piecewise-defined curve or surface) or the shape of the spline in advance. On the other hand, natural cubic splines are preferred when we have prior knowledge or assumptions about the shape of the non-linear relationship. Natural cubic splines are more interpretable and can be controlled by the number of knots (Elhakeem et al., 2022).

Mathematical operators

This section will mention four arithmetic operators, eight distance measurements, and the exponential approach. Also, unlike other approaches, in this section, users can apply logarithmic, exponential, and trigonometric (sinus and cosine) transformations on the markers. Let x_{ij} represent the value of the jth variable for the ith observation, with $i = 1, 2, \ldots, n$ and $j = 1, 2$. Let the resulting combination score for the ith individual be c_i.

- *Arithmetic Operators:* Arithmetic operators such as addition, multiplication, division, and subtraction can also be used in diagnostic tests to optimize the AUC, a measure of diagnostic test performance. These mathematical operations can potentially increase the AUC and improve the efficacy of diagnostic tests by combining markers in specific ways. For example, if high values in one test indicate risk, while low values in the other indicate risk, subtraction or division can effectively combine these markers.
- *Distance Measurements:* While combining markers with mathematical operators, a distance measure is used to evaluate the relationships or similarities between marker values. It's worth noting that, as far as we know, no studies have been integrating various distinct distance measures with arithmetic operators in this context. Euclidean distance is the most commonly used distance measure, which may not accurately reflect the relationship between markers. Therefore, we incorporated a variety of distances into the package we developed. These distances are given as follows (Minaev et al., 2018; Pandit et al., 2011; Cha, 2007).

Euclidean:

$$c = \sqrt{(x_{i1} - 0)^2 + (x_{i2} - 0)^2}. \tag{16}$$

Manhattan:

$$c = |x_{i1} - 0| + |x_{i2} - 0|. \tag{17}$$

Chebyshev:

$$c = \max\{|x_{i1} - 0|, |x_{i2} - 0|\}. \tag{18}$$

Kulczynskid:

$$c = \frac{|x_{i1} - 0| + |x_{i2} - 0|}{\min\{x_{i1}, x_{i2}\}}. \tag{19}$$

Lorentzian:

$$c = \ln(1 + |x_{i1} - 0|) + \ln(1 + |x_{i2} - 0|). \tag{20}$$

Taneja:

$$c = z_1 \left(\log \left(\frac{z_1}{\sqrt{x_{i1}\epsilon}} \right) \right) + z_2 \left(\log \left(\frac{z_2}{\sqrt{x_{i2}\epsilon}} \right) \right) \tag{21}$$

where $z_1 = \frac{x_{i1}-0}{2}$, $z_2 = \frac{x_{i2}-0}{2}$.

Kumar-Johnson:

$$c = \frac{(x_{i1}^2 - 0)^2}{2(x_{i1}\epsilon)^{3/2}} + \frac{(x_{i2}^2 - 0)^2}{2(x_{i2}\epsilon)^{3/2}}, \quad \epsilon = 0.00001. \tag{22}$$

Avg:

$$c = \frac{|x_{i1} - 0| + |x_{i2} - 0| + \max\{(x_{i1} - 0), (x_{i2} - 0)\}}{2}. \tag{23}$$

- *Exponential approach:* The exponential approach is another technique to explore different relationships between the diagnostic measurements. The methods in which one of the two diagnostic tests is considered as the base and the other as an exponent can be represented as $x_{i1}^{(x_{i2})}$ and $x_{i2}^{(x_{i1})}$. The specific goals or hypothesis of the analysis, as well as the characteristics of the diagnostic tests, will determine which method to use.

Machine-learning algorithms Machine-learning algorithms have been increasingly implemented in various fields, including the medical field, to combine diagnostic tests. Integrating diagnostic tests through ML can lead to more accurate, timely, and personalized diagnoses, which are particularly valuable in complex medical cases where multiple factors must be considered. In this study, we aimed to incorporate almost all ML algorithms in the package we developed. We took advantage of the **caret** package in R (Kuhn, 2008) to achieve this goal. This package includes 190 classification algorithms that could be used to train models and make predictions. Our study focused on models that use numerical inputs and produce binary responses depending on the variables/features and the desired outcome. This selection process resulted in 113 models we further implemented in our study. We then classified these 113 models into five classes using the same idea given in (Zararsiz et al., 2016): (i) discriminant classifiers, (ii) decision tree models, (iii) kernel-based classifiers, (iv) ensemble classifiers, and (v) others. Like in the **caret** package, `mlComb()` sets up a grid of tuning parameters for a number of classification routines, fits each model, and calculates a performance measure based on resampling. After the model fitting, it uses the `predict()` function to calculate the probability of the "event" occurring for each observation. Finally, it performs ROC analysis based on the probabilities obtained from the prediction step.

2.2 Standardization

Standardization is converting/transforming data into a standard scale to facilitate meaningful comparisons and statistical inference. Many statistical techniques frequently employ standardization to improve the interpretability and comparability of data. We implemented five different standardization methods that can be applied for each marker, the formulas of which are listed below:

- Z-score: $\frac{x - \text{mean}(x)}{\text{sd}(x)}$
- T-score: $\left(\frac{x - \text{mean}(x)}{\text{sd}(x)} \times 10\right) + 50$
- min_max_scale: $\frac{x - \min(x)}{\max(x) - \min(x)}$
- scale_mean_to_one: $\frac{x}{\text{mean}(x)}$
- scale_sd_to_one: $\frac{x}{\text{sd}(x)}$

2.3 Model building

After specifying a combination method from the **dtComb** package, users can build and optimize model parameters using functions like `mlComb()`, `linComb()`, `nonlinComb()`, and `mathComb()`, depending on the specific model selected. Parameter optimization is done using n-fold cross-validation, repeated n-fold cross-validation, and bootstrapping methods for linear and non-linear approaches (i.e., `linComb()`, `nonlinComb()`). Additionally, for machine-learning approaches (i.e., `mlComb()`), all of the resampling methods from the **caret** package are used to optimize the model parameters. The total number of parameters being

optimized varies across models, and these parameters are fine-tuned to maximize the AUC. The returned object stores input data, preprocessed and transformed data, trained model, and resampling results.

2.4 Evaluation of model performances

A confusion matrix, as shown in Table 1, is a table used to evaluate the performance of a classification model and shows the number of correct and incorrect predictions. It compares predicted and actual class labels, with diagonal elements representing the correct predictions and off-diagonal elements representing the number of incorrect predictions.

Table 1: Confusion Matrix for Model Evaluation

	Actual class labels		
Predicted labels	Positive	Negative	Total
Positive	TP	FP	TP+FP
Negative	FN	TN	FN+TN
Total	TP+FN	FP+TN	n

TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative, n: Sample size

The **dtComb** package uses the **OptimalCutpoints** (López-Ratón et al., 2014) package to generate the confusion matrix and then **epiR** (Stevenson and Sergeant, 2025), including different performance metrics, to evaluate the performances. Various performance metrics accuracy rate (ACC), Kappa statistic (κ), sensitivity (SE), specificity (SP), apparent and true prevalence (AP, TP), positive and negative predictive values (PPV, NPV), positive and negative likelihood ratio (PLR, NLR), the proportion of true outcome negative subjects that test positive (False T+ proportion for true D-), the proportion of true outcome positive subjects that test negative (False T- proportion for true D+), the proportion of test-positive subjects that are outcome negative (False T+ proportion for T+), the proportion of test negative subjects (False T- proportion for T-) that are outcome positive measures are available in the **dtComb** package. These metrics are summarized in Table 2 .

Table 2: Performance metrics and formulas

Performance Metric	**Formula**
Accuracy	$\text{ACC} = \frac{\text{TP}+\text{TN}}{2}$
Kappa	$\kappa = \frac{\text{ACC}-P_e}{1-P_e}$
	$P_e = \frac{(\text{TN}+\text{FN})(\text{TP}+\text{FP})+(\text{FP}+\text{TN})(\text{FN}+\text{TN})}{n^2}$
Sensitivity (Recall)	$\text{SE} = \frac{\text{TP}}{\text{TP}+\text{FN}}$
Specificity	$\text{SP} = \frac{\text{TN}}{\text{TN}+\text{FP}}$
Apparent Prevalence	$\text{AP} = \frac{\text{TP}}{n} + \frac{\text{FP}}{n}$
True Prevalence	$\text{TP} = \frac{\text{AP}+\text{SP}-1}{\text{SE}+\text{SP}-1}$
Positive Predictive Value (Precision)	$\text{PPV} = \frac{\text{TP}}{\text{TP}+\text{FP}}$
Negative Predictive Value	$\text{NPV} = \frac{\text{TN}}{\text{TN}+\text{FN}}$
Positive Likelihood Ratio	$\text{PLR} = \frac{\text{SE}}{1-\text{SP}}$
Negative Likelihood Ratio	$\text{NLR} = \frac{1-\text{SE}}{\text{SP}}$
The Proportion of True Outcome Negative Subjects That Test Positive	$\frac{\text{FP}}{\text{FP}+\text{TN}}$
The Proportion of True Outcome Positive Subjects That Test Negative	$\frac{\text{FN}}{\text{TP}+\text{FN}}$

Performance Metric	Formula
The Proportion of Test Positive Subjects That Are Outcome Negative	$\frac{FP}{TP+FN}$
The Proportion of Test Negative Subjects That Are Outcome Positive	$\frac{FN}{FN+TN}$

2.5 Prediction of the test cases

The class labels of the observations in the test set are predicted with the model parameters derived from the training phase. It is critical to emphasize that the same analytical procedures employed during the training phase have also been applied to the test set, such as normalization, transformation, or standardization. More specifically, if the training set underwent Z-standardization, the test set would similarly be standardized using the mean and standard deviation derived from the training set. The class labels of the test set are then estimated based on the cut-off value established during the training phase and using the model's parameters that are trained using the training set.

2.6 Technical details and the structure of dtComb

The **dtComb** package is implemented using the R programming language (`https://www.r-project.org/`) version 4.2.0. Package development was facilitated with **devtools** (Wickham et al., 2022) and documented with **roxygen2** (Wickham et al., 2025). Package testing was performed using 271 unit tests (Wickham, 2011). Double programming was performed using Python (`https://www.python.org/`) to validate the implemented functions (Shiralkar, 2010).

To combine diagnostic tests, the **dtComb** package allows the integration of eight linear combination methods, seven non-linear combination methods, arithmetic operators, and, in addition to these, eight distance metrics within the scope of mathematical operators and a total of 113 machine-learning algorithms from the **caret** package (Kuhn, 2008). These are summarized in Table 3.

Table 3: Features of dtComb

Modules (Tab Panels)	Features
Combination Methods	• Linear Combination Approach (8 Different methods) • Non-linear Combination Approach (7 Different Methods) • Mathematical Operators (14 Different methods) • Machine-Learning Algorithms (113 Different Methods) (Kuhn, 2008)
Preprocessing	• Five standardization methods applicable to linear, non-linear, mathematical methods • 16 preprocessing methods applicable to ML (Kuhn, 2008)
Resampling	• Three different methods for linear and non-linear combination methods: - Bootstrapping - Cross-validation - Repeated cross-validation • 12 different resampling methods for ML (Kuhn, 2008)
Cutpoints	• 34 different methods for optimum cutpoints (López-Ratón et al., 2014)

3 Results

Table 4 summarizes the existing packages and programs, including **dtComb**, along with the number of combination methods included in each package. While `mROC` offers only one linear combination method, **maxmzpAUC** and **movieROC** provide five linear combination techniques each, and **SLModels** includes four. However, these existing packages primarily focus on linear combination approaches. In contrast, **dtComb** goes beyond these limitations by integrating not only linear methods but also non-linear approaches, machine learning algorithms, and mathematical operators.

Table 4: Comparison of dtComb vs. existing packages and programs

Packages & Programs	Linear Comb.	Non-linear Comb.	Math. Operators	ML algorithms
mROC (Kramar et al., 2001)	1	-	-	-
maxmzpAUC (Yu and Park, 2015)	5	-	-	-
movieROC (Pérez-Fernández et al., 2021)	5	-	-	-
SLModels (Aznar-Gimeno et al., 2023)	4	-	-	-
dtComb	8	7	14	113

3.1 Dataset

To demonstrate the functionality of the **dtComb** package, we conduct a case study using four different combination methods. The data used in this study were obtained from patients who presented at Erciyes University Faculty of Medicine, Department of General Surgery, with complaints of abdominal pain (Zararsiz et al., 2016; Akyildiz et al., 2010). The dataset comprised D-dimer levels (*D_dimer*) and leukocyte counts (*log_leukocyte*) of 225 patients, divided into two groups (*Group*): the first group consisted of 110 patients who required an immediate laparotomy (*nedeed*). In comparison, the second group comprised 115 patients who did not (*not_nedeed*). After the evaluation of conventional treatment, the patients who underwent surgery due to their postoperative pathologies are placed in the first group. In contrast, those with a negative result from their laparotomy were assigned to the second group. All the analyses were performed by following a workflow given in Figure 1. First of all, the **dtComb** package should be loaded in order to use related functions.

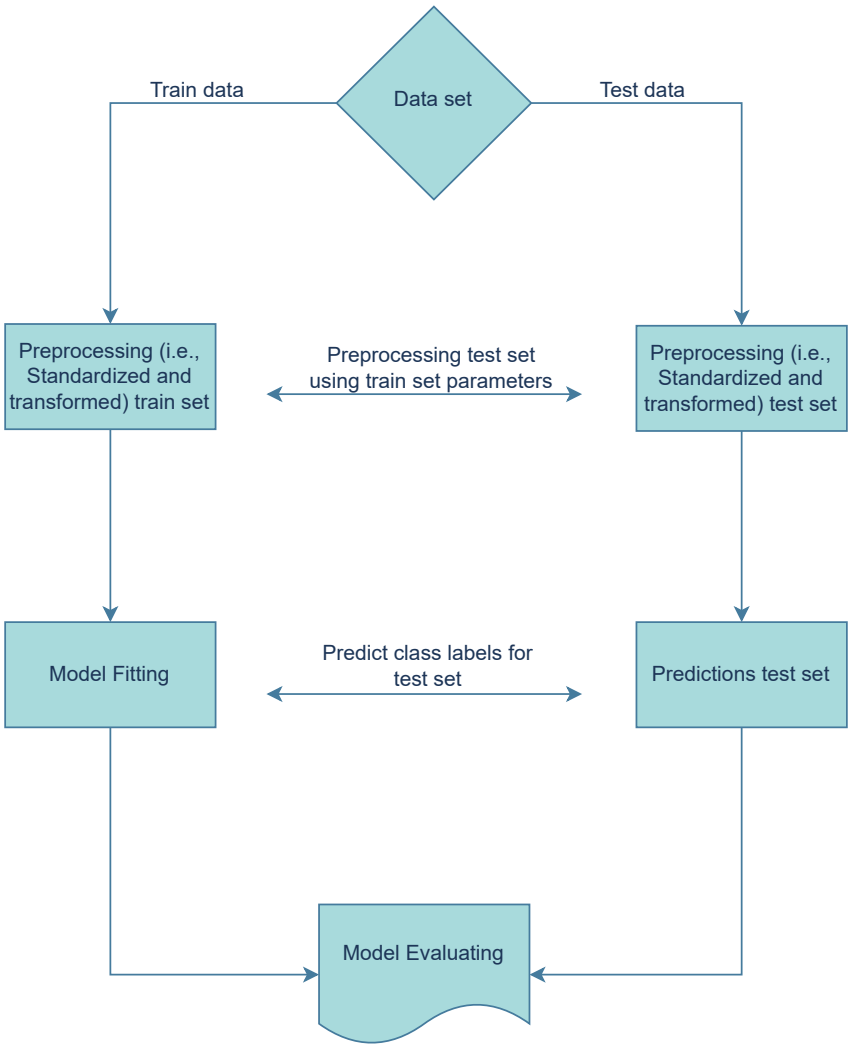

Figure 1: Combination steps of two diagnostic tests. The figure presents a schematic representation of the sequential steps involved in combining two diagnostic tests using a combination method.

```
# load dtComb package
library(dtComb)
```

Similarly, the laparotomy data can be loaded from the R database by using the following R code:

```
# load laparotomy data
data(laparotomy)
```

3.2 Implementation of the dtComb package

In order to demonstrate the applicability of the **dtComb** package, the implementation of an arbitrarily chosen method from each of the linear, non-linear, mathematical operator and machine learning approaches is demonstrated and their performance is compared. These methods are Pepe, Cai & Langton for linear combination, Splines for non-linear, Addition for mathematical operator and Support Vector Machine (SVM) for machine-learning. Before applying the methods, we split the data into two parts: a training set comprising 70% of the data and a test set comprising the remaining 30%.

```
# Splitting the data set into train and test (70%-30%)
set.seed(2128)
```

```
inTrain <- caret::createDataPartition(laparotomy$group, p = 0.7,
                                      list = FALSE)
trainData <- laparotomy[inTrain, ]
colnames(trainData) <- c("Group", "D_dimer", "log_leukocyte")
testData <- laparotomy[-inTrain, -1]

# define marker and status for combination function
markers <- trainData[, -1]
status <- factor(trainData$Group, levels = c("not_needed", "needed"))
```

The model is trained on `trainData` and the resampling parameters used in the training phase are chosen as ten repeat five fold repeated cross-validation. Direction = '<' is chosen, as higher values indicate higher risks. The Youden index was chosen among the cut-off methods. We note that markers are not standardised and results are presented at the confidence level (CI 95%). Four main combination functions are run with the selected methods as follows.

```
# PCL method
fit.lin.PCL <- linComb(markers = markers,  status = status, event = "needed",
                       method = "PCL", resample = "repeatedcv", nfolds = 5,
                       nrepeats = 10, direction = "<",
                       cutoff.method = "Youden")

# splines method (degree = 3 and degrees of freedom = 3)
fit.nonlin.splines <- nonlinComb(markers = markers, status = status,
                                 event = "needed", method = "splines",
                                 resample = "repeatedcv", nfolds = 5,
                                 nrepeats = 10, cutoff.method = "Youden",
                                 direction = "<", df1 = 3, df2 = 3)

# add operator
fit.add <- mathComb(markers = markers, status = status, event = "needed",
                    method = "add", direction = "<",
                    cutoff.method = "Youden")

# SVM
fit.svm <- mlComb(markers = markers, status = status, event = "needed",
                  method = "svmLinear", resample = "repeatedcv",
                  nfolds  = 5, nrepeats = 10, direction = "<",
                  cutoff.method = "Youden")
```

Various measures were considered to compare model performances, including AUC, ACC, SEN, SPE, PPV, and NPV. AUC statistics, with 95% CI, have been calculated for each marker and method. The resulting statistics are as follows: 0.816 (0.751–0.880), 0.802 (0.728–0.877), 0.879 (0.825–0.932), 0.911 (0.868–0.954), 0.877 (0.824-0.929), and 0.875 (0.821-0.930) for D-dimer, Log(leukocyte), Pepe, Cai & Langton, Splines, Addition, and SVM. The results revealed that the predictive performances of markers and the combination of markers are significantly higher than random chance in determining the use of laparotomy ($p < 0.05$). The highest sensitivity and NPV were observed with the Addition method, while the highest specificity and PPV were observed with the Splines method. According to the overall AUC and accuracies, the combined approach fitted with the Splines method performed better than the other methods (Figure 2). Therefore, the Splines method will be used in the subsequent analysis of the findings.

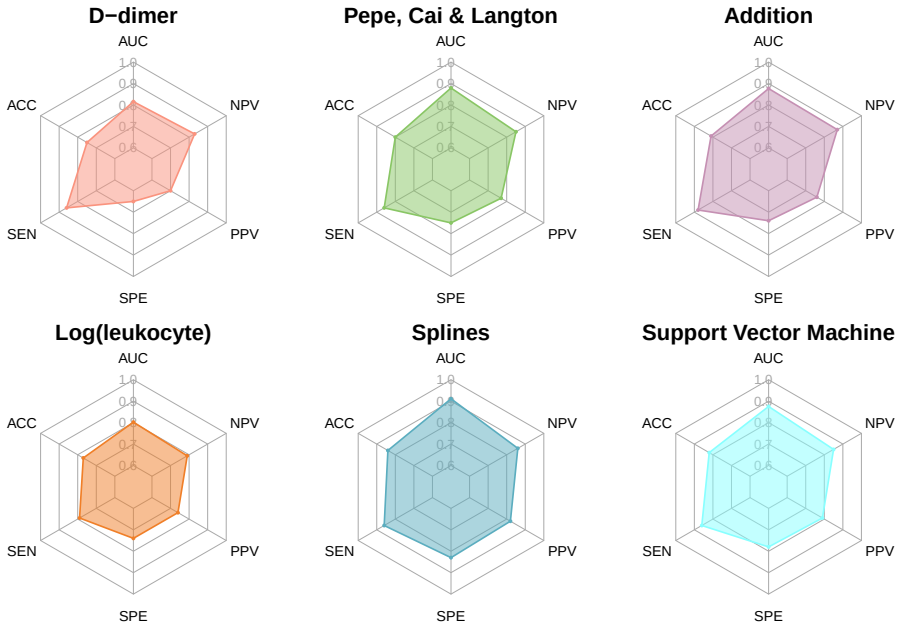

Figure 2: Radar plots of trained models and performance measures of two markers. Radar plots summarize the diagnostic performances of two markers and various combination methods in the training dataset. These plots illustrate the performance metrics such as AUC, ACC, SEN, SPE, PPV, and NPV measurements. In these plots, the width of the polygon formed by connecting each point indicates the model's performance in terms of AUC, ACC, SEN, SPE, PPV, and NPV metrics. It can be observed that the polygon associated with the Splines method occupies the most expensive area, which means that the Splines method performed better than the other methods.

The area under ROC curves for D-dimer levels and leukocyte counts on the logarithmic scale and combination score were 0.816, 0.802, and 0.911, respectively. The ROC curves generated with the combination score from the splines model, D-dimer levels, and leukocyte count markers are also given in Figure 3, showing that the combination score has the highest AUC. It is observed that the splines method significantly improved between 9.5% and 10.9% in AUC statistics compared to D-dimer level and leukocyte counts, respectively.

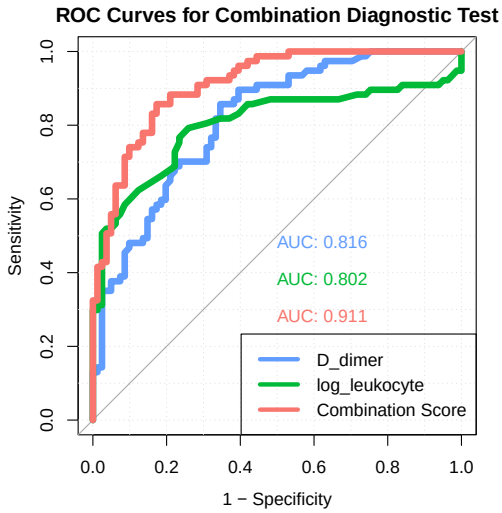

Figure 3: ROC curves. ROC curves for combined diagnostic tests, with sensitivity displayed on the y-axis and 1-specificity displayed on the x-axis. As can be observed, the combination score produced the highest AUC value, indicating that the combined strategy performs the best overall.

For the AUC of markers and the spline model:

```
fit.nonlin.splines$AUC_table
```

```
#>                    AUC     SE.AUC LowerLimit UpperLimit        z     p.value
#> D_dimer     0.8156966 0.03303310  0.7509530  0.8804403  9.556979 1.212446e-21
#> log_leukocyte 0.8022286 0.03791768  0.7279113  0.8765459  7.970652 1.578391e-15
#> Combination  0.9111752 0.02177326  0.8685004  0.9538500 18.884417 1.532212e-79
```

Here: SE: Standard Error.

To see the results of the binary comparison between the combination score and markers:

```
fit.nonlin.splines$MultComp_table
```

```
#>   Marker1 (A)   Marker2 (B)   AUC (A)   AUC (B)     |A-B|  SE(|A-B|)        z
#> 1 Combination     D_dimer 0.9111752 0.8156966 0.09547860 0.02390580 3.9939507
#> 2 Combination log_leukocyte 0.9111752 0.8022286 0.10894661 0.03456988 3.1514896
#> 3   D_dimer log_leukocyte 0.8156966 0.8022286 0.01346801 0.04847560 0.2778308
#>         p-value
#> 1 6.498138e-05
#> 2 1.624400e-03
#> 3 7.811423e-01
```

Controlling Type I error using Bonferroni correction, comparison of combination score with markers yielded significant results ($p < 0.05$).

To demonstrate the diagnostic test results and performance measures for the non-linear combination approach, the following code can be used:

```
fit.nonlin.splines$DiagStatCombined
```

```
#>           Outcome +    Outcome -      Total
#> Test +           66           14         80
#> Test -           11           67         78
#> Total            77           81        158
#>
#> Point estimates and 95% CIs:
#> -------------------------------------------------------------
#> Apparent prevalence *                 0.51 (0.43, 0.59)
#> True prevalence *                     0.49 (0.41, 0.57)
#> Sensitivity *                         0.86 (0.76, 0.93)
#> Specificity *                         0.83 (0.73, 0.90)
#> Positive predictive value *           0.82 (0.72, 0.90)
#> Negative predictive value *           0.86 (0.76, 0.93)
#> Positive likelihood ratio             4.96 (3.05, 8.06)
#> Negative likelihood ratio             0.17 (0.10, 0.30)
#> False T+ proportion for true D- *     0.17 (0.10, 0.27)
#> False T- proportion for true D+ *     0.14 (0.07, 0.24)
#> False T+ proportion for T+ *          0.17 (0.10, 0.28)
#> False T- proportion for T- *          0.14 (0.07, 0.24)
#> Correctly classified proportion *     0.84 (0.78, 0.89)
#> -------------------------------------------------------------
#> * Exact CIs
```

Furthermore, if the diagnostic test results and performance measures of the combination score are compared with the results of the single markers, it can be observed that the TN value of the combination score is higher than that of the single markers, and the combination of markers has higher specificity and positive-negative predictive value than the log-transformed leukocyte counts and D-dimer level (Table 5). Conversely, D-dimer has a

higher sensitivity than the others. Optimal cut-off values for both markers and the combined approach are also given in this table.

Table 5: Statistical diagnostic measures with 95% confidence intervals for each marker and the combination score

Diagnostic Measures (95% CI)	**D-dimer level (> 1.6)**	**Log(leukocyte count) (> 4.16)**	**Combination score (> 0.437)**
TP	66	61	66
TN	53	60	67
FP	28	21	14
FN	11	16	11
Apparent prevalence	0.59 (0.51-0.67)	0.52 (0.44-0.60)	0.51 (0.43-0.59)
True prevalence	0.49 (0.41-0.57)	0.49 (0.41-0.57)	0.49 (0.41-0.57)
Sensitivity	0.86 (0.76-0.93)	0.79 (0.68-0.88)	0.86 (0.76-0.93)
Specificity	0.65 (0.54-0.76)	0.74 (0.63-0.83)	0.83 (0.73-0.90)
Positive predictive value	0.70 (0.60-0.79)	0.74 (0.64-0.83)	0.82 (0.72-0.90)
Negative predictive value	0.83 (0.71-0.91)	0.79 (0.68-0.87)	0.86 (0.76-0.93)
Positive likelihood ratio	2.48 (1.81-3.39)	3.06 (2.08-4.49)	4.96 (3.05-8.06)
Negative likelihood ratio	0.22 (0.12-0.39)	0.28 (0.18-0.44)	0.17 (0.10-0.30)
False T+ proportion for true D-	0.35 (0.24-0.46)	0.26 (0.17-0.37)	0.17 (0.10-0.27)
False T- proportion for true D+	0.14 (0.07-0.24)	0.21 (0.12-0.32)	0.14 (0.07-0.24)
False T+ proportion for T+	0.30 (0.21-0.40)	0.26 (0.17-0.36)	0.17 (0.10-0.28)
False T- proportion for T-	0.17 (0.09-0.29)	0.21 (0.13-0.32)	0.14 (0.07-0.24)
Accuracy	0.75 (0.68-0.82)	0.77 (0.69-0.83)	0.84 (0.78-0.89)

For a comprehensive analysis, the `plotComb` function in **dtComb** can be used to generate plots of the kernel density and individual-value of combination scores of each group and the specificity and sensitivity corresponding to different cut-off point values Figure 4. This function requires the result of the `nonlinComb` function, which is an object of the "dtComb" class and `status` which is of factor type.

```
# draw distribution, dispersion, and specificity and sensitivity plots
p <- plotComb(fit.nonlin.splines, status)
p$all
```

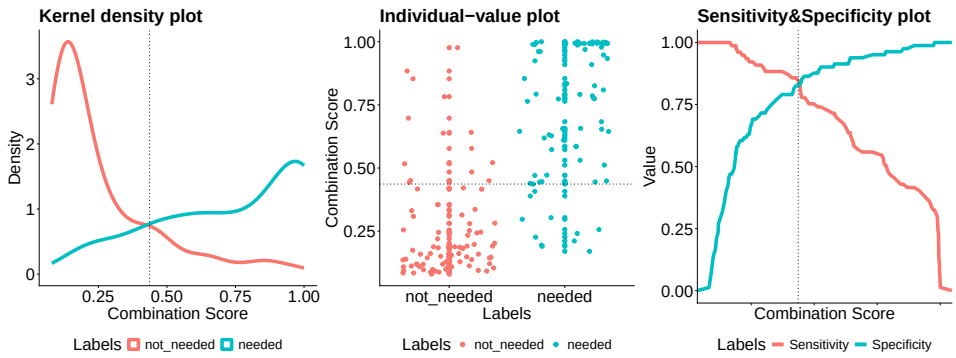

Figure 4: Kernel density, individual-value, and sens&spe plots of the combination score acquired with the training model. Kernel density of the combination score for two groups: needed and not needed. Individual-value graph with classes on the x-axis and combination score on the y-axis. Sensitivity and specificity graph of the combination score. While colors show each class in the density and individual-value plots, in the sensitivity and specificity plot, the colors represent the sensitivity and specificity of the combination score.

If the model trained with Splines is to be tested, the generically written predict function is used. This function requires the test set and the result of the `nonlinComb` function, which is an object of the "dtComb" class. As a result of prediction, the output for each observation consisted of the combination score and the predicted label determined by the cut-off value derived from the model.

```
# To predict the test set
pred <- predict(fit.nonlin.splines, testData)
head(pred)
```

```
#>    comb.score labels
#> 1   0.6133884 needed
#> 7   0.9946474 needed
#> 10  0.9972347 needed
#> 11  0.9925040 needed
#> 13  0.9257699 needed
#> 14  0.9847090 needed
```

Above, it can be seen that the estimated combination scores for the first six observations in the test set were labelled as **needed** because they were higher than the cut-off value of 0.448.

3.3 Web interface for the dtComb package

The primary goal of developing the **dtComb** package is to combine numerous distinct combination methods and make them easily accessible to researchers. Furthermore, the package includes diagnostic statistics and visualization tools for diagnostic tests and the combination score generated by the chosen method. Nevertheless, it is worth noting that using R code may pose challenges for physicians and those unfamiliar with R programming. We have also developed a user-friendly web application for **dtComb** using **shiny** (Chang et al., 2025) to address this. This web-based tool is publicly accessible and provides an interactive interface with all the functionalities found in the **dtComb** package.

To initiate the analysis, users must upload their data by following the instructions outlined in the "Data upload" tab of the web tool. For convenience, we have provided three example datasets on this page to assist researchers in practicing the tool's functionality and to guide them in formatting their own data (as illustrated in Figure 5.a). We also note that ROC analysis for a single marker can be performed within the 'ROC Analysis for Single Marker(s)' tab in the data upload section of the web interface.

In the "Analysis" tab, one can find two crucial subpanels:

- **Plots** (Figure 5.b): This section offers various visual representations, such as ROC curves, kernel density plots, individual-value plots, and sensitivity and specificity plots. These visualizations help users assess single diagnostic tests and the combination score generated using user-defined combination methods.
- **Results** (Figure 5.c): In this subpanel, one can access a range of statistics. It provides insights into the combination score and single diagnostic tests, AUC statistics, and comparisons to evaluate how the combination score fares against individual diagnostic tests, and various diagnostic measures. One can also predict new data based on the model parameters set previously and stored in the "Predict" tab (Figure 5.d). If needed, one can download the model created during the analysis to keep the parameters of the fitted model. This lets users make new predictions by reloading the model from the "Predict" tab. Additionally, all the results can easily be downloaded using the dedicated download buttons in their respective tabs.

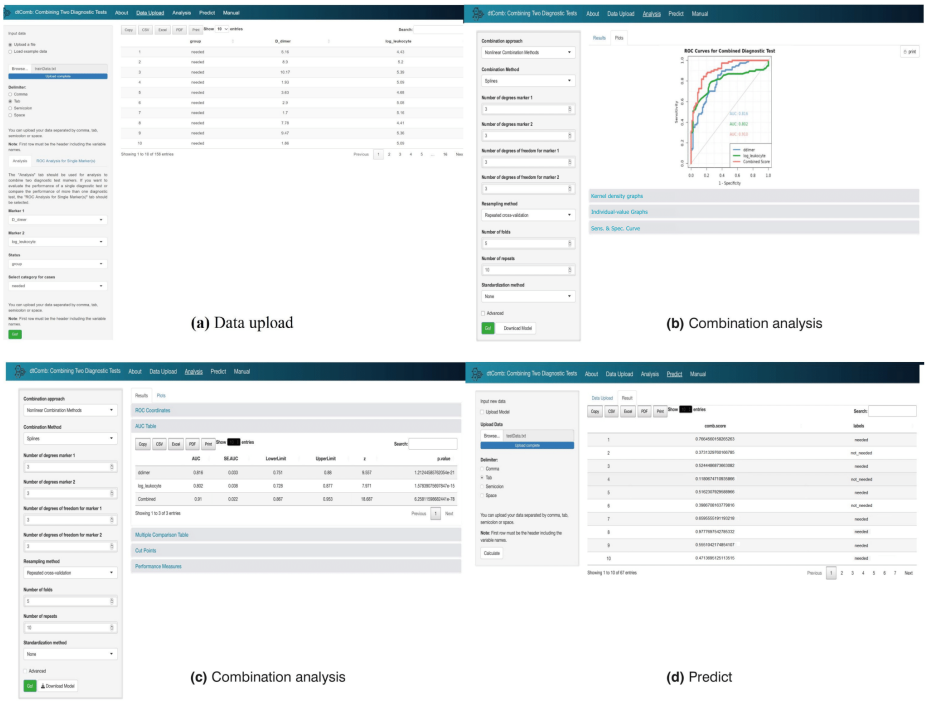

Figure 5: Web interface of the dtComb package. The figure illustrates the web interface of the dtComb package, which demonstrates the steps involved in combining two diagnostic tests. a) Data Upload: The user is able to upload the dataset and select relevant markers, a gold standard test, and an event factor for analysis. b) Combination Analysis: This panel allows the selection of the combination method, method-specific parameters, and resampling options to refine the analysis. c) Combination Analysis Output: Displays the results generated by the selected combination method, providing the user with key metrics and visualizations for interpretation. d) Predict: Displays the prediction results of the trained model when applied to the test set.

4 Summary and further research

In clinical practice, multiple diagnostic tests are possible for disease diagnosis (Yu and Park, 2015). Combining these tests to enhance diagnostic accuracy is a widely accepted approach (Su and Liu, 1993; Pepe and Thompson, 2000; Liu et al., 2011; Sameera et al., 2016; Pepe et al., 2006; Todor et al., 2014). As far as we know, the tools in Table 4 have been designed to combine diagnostic tests but only contain at most five different combination methods. As a

result, despite the existence of numerous advanced combination methods, there is currently no extensive tool available for integrating diagnostic tests.

In this study, we presented **dtComb**, a comprehensive R package designed to combine diagnostic tests using various methods, including linear, non-linear, mathematical operators, and machine learning algorithms. The package integrates 142 different methods for combining two diagnostic markers to improve the accuracy of diagnosis. The package also provides ROC curve analysis, various graphical approaches, diagnostic performance scores, and binary comparison results. In the given example, one can determine whether patients with abdominal pain require laparotomy by combining the D-dimer levels and white blood cell counts of those patients. Various methods, such as linear and non-linear combinations, were tested, and the results showed that the Splines method performed better than the others, particularly in terms of AUC and accuracy compared to single tests. This shows that diagnostic accuracy can be improved with combination methods.

Future work can focus on extending the capabilities of the **dtComb** package. While some studies focus on combining multiple markers, our study aimed to combine two markers using nearly all existing methods and develop a tool and package for clinical practice (Kang et al., 2016).

5 R software

The R package **dtComb** is now available on the CRAN website `https://cran.r-project.org/web/packages/dtComb/index.html` (Yerlitas et al., 2025).

6 Acknowledgment

We would like to thank the Proofreading & Editing Office of the Dean for Research at Erciyes University for the copyediting and proofreading service for this manuscript.

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org. [p81]

S. Agarwal, A. S. Yadav, V. Dinesh, K. S. S. Vatsav, K. S. S. Prakash, and S. Jaiswal. By artificial intelligence algorithms and machine learning models to diagnosis cancer. *Materials Today: Proceedings*, 80:2969–2975, 2023. URL `https://doi.org/10.1016/j.matpr.2021.07.088`. [p81]

M. Ahsan, A. Khan, K. R. Khan, B. B. Sinha, and A. Sharma. Advancements in medical diagnosis and treatment through machine learning: A review. *Expert Systems*, 41(3):e13499, 2024. URL `https://doi.org/10.1111/exsy.13499`. [p81]

H. Y. Akyildiz, E. Sozuer, A. Akcan, C. Kuçuk, T. Artis, İ. Biri, and N. Yılmaz. The value of d-dimer test in the diagnosis of patients with nontraumatic acute abdomen. *Turkish Journal of Trauma and Emergency Surgery*, 16(1):22–26, 2010. URL `https://jag.journalagent.com/z4/jvi.asp?pdir=travma&plng=eng&un=UTD-21703&look4=`. [p90]

M. Alzyoud, R. Alazaidah, M. Aljaidi, G. Samara, M. Qasem, M. Khalid, and N. Al-Shanableh. Diagnosing diabetes mellitus using machine learning techniques. *International*

Journal of Data and Network Science, 8(1):179–188, 2024. URL https://doi.org/10.5267/j.ijdns.2023.10.006. [p81]

T. W. Anderson and R. R. Bahadur. Classification into two multivariate normal distributions with different covariance matrices. *The annals of mathematical statistics*, pages 420–431, 1962. URL https://doi.org/10.1214/aoms/1177704568. [p83]

R. Aznar-Gimeno, L. M. Esteban, R. del Hoyo-Alonso, Á. Borque-Fernando, and G. Sanz. A stepwise algorithm for linearly combining biomarkers under youden index maximization. *Mathematics*, 10(8):1221, 2022. URL https://doi.org/10.3390/math10081221. [p80]

R. Aznar-Gimeno, L. Esteban, G. Sanz, and R. del Hoyo-Alonso. Comparing the min–max–median/iqr approach with the min–max approach, logistic regression and xgboost, maximising the youden index. *Symmetry (Basel)*, 15, 2023. doi: 10.3390/sym15030756. URL https://doi.org/10.3390/sym15030756. [p90]

A. Bansal and M. Sullivan Pepe. When does combining markers improve classification performance and what are implications for practice? *Statistics in medicine*, 32(11):1877–1892, 2013. URL https://doi.org/10.1002/sim.5736. [p80]

S.-H. Cha. Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2):1, 2007. URL https://api.semanticscholar.org/CorpusID:15506682. [p86]

W. Chang, J. Cheng, J. Allaire, C. Sievert, B. Schloerke, Y. Xie, J. Allen, J. McPherson, A. Dipert, and B. Borges. *shiny: Web Application Framework for R*, 2025. URL https://CRAN.R-project.org/package=shiny. R package version 1.11.1. [p96]

D. Cox and E. Snell. *Analysis of Binary Data*. Chapman and Hall/CRC, London, 2nd edition, 1989. URL https://doi.org/10.1201/9781315137391. [p82]

W. DeGroat, H. Abdelhalim, K. Patel, D. Mendhe, S. Zeeshan, and Z. Ahmed. Discovering biomarkers associated and predicting cardiovascular disease with high accuracy using a novel nexus of machine learning techniques for precision medicine. *Scientific reports*, 14 (1):1, 2024. URL https://doi.org/10.1038/s41598-023-50600-8. [p81]

Z. Du, P. Du, and A. Liu. Likelihood ratio combination of multiple biomarkers via smoothing spline estimated densities. *Statistics in Medicine*, 43(7):1372–1383, 2024. URL https://doi.org/10.1002/sim.10026. [p80]

B. Efron. The efficiency of logistic regression compared to normal discriminant analysis. *Journal of the American Statistical Association*, 70(352):892–898, 1975. URL https://doi.org/10.2307/2285453. [p82]

A. Elhakeem, R. A. Hughes, K. Tilling, D. L. Cousminer, S. A. Jackowski, T. J. Cole, A. S. Kwong, Z. Li, S. F. Grant, A. D. Baxter-Jones, et al. Using linear and natural cubic splines, sitar, and latent trajectory models to characterise nonlinear longitudinal growth trajectories in cohort studies. *BMC Medical Research Methodology*, 22(1):68, 2022. URL https://doi.org/10.1186/s12874-022-01542-8. [p86]

G. Ertürk Zararsız. Linear combination of leukocyte count and d-dimer levels in the diagnosis of patients with non-traumatic acute abdomen. *Med. Rec.*, 5:84–90, 2023. URL https://doi.org/10.37990/medr.1166531. [p80]

S. S. Faria, P. C. Fernandes Jr, M. J. B. Silva, V. C. Lima, W. Fontes, R. Freitas-Junior, A. K. Eterovic, and P. Forget. The neutrophil-to-lymphocyte ratio: a narrative review. *ecancermedicalscience*, 10, 2016. URL http://doi.org/10.3332/ecancer.2016.702. [p80]

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010. URL https://doi.org/10.18637/jss.v033.i01. [p85]

S. Ganapathy, H. KT, B. Jindal, P. S. Naik, and S. Nair N. Comparison of diagnostic accuracy of models combining the renal biomarkers in predicting renal scarring in pediatric population with vesicoureteral reflux (vur). *Irish Journal of Medical Science (1971-)*, 192(5): 2521–2526, 2023. URL https://doi.org/10.1007/s11845-023-03275-z. [p81]

D. Ghosh and A. M. Chinnaiyan. Classification and selection of biomarkers in genomic data using lasso. *BioMed Research International*, 2005(2):147–154, 2005. URL https://doi.org/10.1155/JBB.2005.147. [p80]

J. Grau, I. Grosse, and J. Keilwagen. Prroc: computing and visualizing precision-recall and receiver operating characteristic curves in r. *Bioinformatics*, 31(15):2595–2597, 2015. URL https://doi.org/10.1093/bioinformatics/btv153. [p81]

T. Hastie. *gam: Generalized Additive Models*, 2025. URL https://CRAN.R-project.org/package=gam. R package version 1.22-6. [p86]

G. James, D. Witten, T. Hastie, R. Tibshirani, et al. *An introduction to statistical learning*, volume 112. Springer, 2013. URL https://doi.org/10.1007/978-1-4614-7138-7. [p84, 85, 86]

L. Kang, A. Liu, and L. Tian. Linear combination methods to improve diagnostic/prognostic accuracy on future observations. *Statistical methods in medical research*, 25(4):1359–1380, 2016. URL https://doi.org/10.1177/0962280213481053. [p83, 98]

A. Kramar, D. Faraggi, A. Fortuné, and B. Reiser. mroc: a computer program for combining tumour markers in predicting disease states. *Computer methods and programs in biomedicine*, 66(2-3):199–207, 2001. URL https://doi.org/10.1016/S0169-2607(00)00129-2. [p80, 90]

M. Kuhn. Building predictive models in r using the caret package. *Journal of statistical software*, 28:1–26, 2008. URL https://doi.org/10.18637/jss.v028.i05. [p81, 87, 89]

C. León, S. Ruiz-Santana, P. Saavedra, B. Almirante, J. Nolla-Salas, F. Álvarez-Lerma, J. Garnacho-Montero, M. Á. León, E. S. Group, et al. A bedside scoring system ("candida score") for early antifungal treatment in nonneutropenic critically ill patients with candida colonization. *Critical care medicine*, 34(3):730–737, 2006. URL https://doi.org/10.1097/01.CCM.0000202208.37364.7D. [p82]

C. Liu, A. Liu, and S. Halabi. A min–max combination of biomarkers to improve diagnostic accuracy. *Statistics in medicine*, 30(16):2005–2014, 2011. URL https://doi.org/10.1002/sim.4238. [p83, 97]

M. López-Ratón, M. X. Rodríguez-Álvarez, C. C. Suárez, and F. G. Sampedro. OptimalCutpoints: An R package for selecting optimal cutpoints in diagnostic tests. *Journal of Statistical Software*, 61(8):1–36, 2014. doi: 10.18637/jss.v061.i08. [p88, 89]

J. Luo, F. Yu, H. Zhou, X. Wu, Q. Zhou, Q. Liu, and S. Gan. Ast/alt ratio is an independent risk factor for diabetic retinopathy: A cross-sectional study. *Medicine*, 103(26):e38583, 2024. URL https://doi.org/10.1097/MD.0000000000038583. [p81]

G. Minaev, R. Piché, and A. Visa. Distance measures for classification of numerical features. 2018. URL https://trepo.tuni.fi/handle/10024/124353. [p86]

E. G. Müller, T. H. Edwin, C. Stokke, S. S. Navelsaker, A. Babovic, N. Bogdanovic, A. B. Knapskog, and M. E. Revheim. Amyloid-β pet—correlation with cerebrospinal fluid biomarkers and prediction of alzheimer´ s disease diagnosis in a memory clinic. *PloS one*, 14(8):e0221365, 2019. URL https://doi.org/10.1371/journal.pone.0221365. [p80]

M. Neumann, H. Kothare, and V. Ramanarayanan. Combining multiple multimodal speech features into an interpretable index score for capturing disease progression in amyotrophic lateral sclerosis. *Interspeech*, 2353, 2023. URL https://doi.org/10.21437/interspeech.2023-2100. [p80]

H. Nyblom, E. Björnsson, M. Simrén, F. Aldenborg, S. Almer, and R. Olsson. The ast/alt ratio as an indicator of cirrhosis in patients with pbc. *Liver International*, 26(7):840–845, 2006. URL http://doi.org/10.1111/j.1478-3231.2006.01304.x. [p80]

S. Pandit, S. Gupta, et al. A comparative study on distance measuring approaches for clustering. *International journal of research in computer science*, 2(1):29–31, 2011. URL https://doi.org/10.7815/ijorcs.21.2011.011. [p86]

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011. URL http://jmlr.org/papers/v12/pedregosa11a.html. [p81]

M. S. Pepe. *The statistical evaluation of medical tests for classification and prediction*. Oxford university press, 2003. URL https://doi.org/10.1093/oso/9780198509844.001.0001. [p83]

M. S. Pepe and M. L. Thompson. Combining diagnostic test results to increase accuracy. *Biostatistics*, 1(2):123–140, 2000. URL https://doi.org/10.1093/biostatistics/1.2.123. [p80, 82, 83, 97]

M. S. Pepe, T. Cai, and G. Longton. Combining predictors for classification using the area under the receiver operating characteristic curve. *Biometrics*, 62(1):221–229, 2006. URL https://doi.org/10.1111/j.1541-0420.2005.00420.x. [p83, 97]

S. Pérez-Fernández, P. Martínez-Camblor, P. Filzmoser, and N. Corral. Visualizing the decision rules behind the roc curves: understanding the classification process. *AStA Advances in Statistical Analysis*, 105(1):135–161, 2021. URL https://doi.org/10.1007/s10182-020-00385-2. [p80, 90]

F. Prinzi, C. Militello, N. Scichilone, S. Gaglio, and S. Vitabile. Explainable machine-learning models for covid-19 prognosis prediction using clinical, laboratory and radiomic features. *IEEE Access*, 11:121492–121510, 2023. URL https://doi.org/10.1109/ACCESS.2023.3327808. [p81]

X. Robin, N. Turck, A. Hainard, N. Tiberti, F. Lisacek, J.-C. Sanchez, and M. Müller. proc: an open-source package for r and s+ to analyze and compare roc curves. *BMC Bioinformatics*, 12:77, 2011. [p81]

S. Ruiz-Velasco. Asymptotic efficiency of logistic regression relative to linear discriminant analysis. *Biometrika*, 78(2):235–243, 1991. URL https://doi.org/10.2307/2337248. [p82]

M. C. Sachs. plotroc: a tool for plotting roc curves. *Journal of statistical software*, 79, 2017. URL https://doi.org/10.18637/jss.v079.c02. [p81]

N. Salvetat, F. J. Checa-Robles, V. Patel, C. Cayzac, B. Dubuc, F. Chimienti, J.-D. Abraham, P. Dupré, D. Vetter, S. Méreuze, et al. A game changer for bipolar disorder diagnosis using rna editing-based biomarkers. *Translational Psychiatry*, 12(1):182, 2022. URL https://doi.org/10.1038/s41398-022-01938-6. [p81]

N. Salvetat, F. J. Checa-Robles, A. Delacrétaz, C. Cayzac, B. Dubuc, D. Vetter, J. Dainat, J.-P. Lang, F. Gamma, and D. Weissmann. Ai algorithm combined with rna editing-based blood biomarkers to discriminate bipolar from major depressive disorders in an external validation multicentric cohort. *Journal of Affective Disorders*, 356:385–393, 2024. URL https://doi.org/10.1016/j.jad.2024.04.022. [p81]

G. Sameera, R. V. Vardhan, and K. Sarma. Binary classification using multivariate receiver operating characteristic curve for continuous data. *Journal of biopharmaceutical statistics*, 26 (3):421–431, 2016. URL https://doi.org/10.1080/10543406.2015.1052479. [p81, 84, 97]

D. Serban, N. Papanas, A. M. Dascalu, P. Kempler, I. Raz, A. A. Rizvi, M. Rizzo, C. Tudor, M. Silviu Tudosie, D. Tanasescu, et al. Significance of neutrophil to lymphocyte ratio (nlr) and platelet lymphocyte ratio (plr) in diabetic foot ulcer and potential new therapeutic targets. *The International Journal of Lower Extremity Wounds*, 23(2):205–216, 2024. URL https://doi.org/10.1177/15347346211057742. [p81]

A. Sewak, S. Siegfried, and T. Hothorn. Construction and evaluation of optimal diagnostic tests with application to hepatocellular carcinoma diagnosis. *arXiv preprint arXiv:2402.03004*, 2024. URL https://doi.org/10.48550/arXiv.2402.03004. [p81]

P. Shiralkar. Programming validation: Perspectives and strategies. *PharmaSUG 2010—paper IB09*, 2010. URL https://www.lexjansen.com/pharmasug/2010/IB/IB09.pdf. [p89]

T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer. Rocr: visualizing classifier performance in r. *Bioinformatics*, 21(20):7881, 2005. URL http://rocr.bioinf.mpi-sb.mpg.de. [p81]

M. Stevenson and E. Sergeant. *epiR: Tools for the Analysis of Epidemiological Data*, 2025. URL https://CRAN.R-project.org/package=epiR. R package version 2.0.86. [p88]

J. Q. Su and J. S. Liu. Linear combinations of multiple diagnostic markers. *Journal of the American Statistical Association*, 88(424):1350–1355, 1993. URL https://doi.org/10.2307/2291276. [p80, 83, 97]

K. Svart, J. J. Korsbæk, R. H. Jensen, T. Parkner, C. S. Knudsen, S. G. Hasselbalch, S. M. Hagen, E. A. Wibroe, L. D. Molander, and D. Beier. Neurofilament light chain is elevated in patients with newly diagnosed idiopathic intracranial hypertension: A prospective study. *Cephalalgia*, 44(5):03331024241248203, 2024. URL https://doi.org/10.1177/03331024241248203. [p81]

N. Todor, I. Todor, and G. Săplăcan. Tools to identify linear combination of prognostic factors which maximizes area under receiver operator curve. *Journal of clinical bioinformatics*, 4: 1–7, 2014. URL https://doi.org/10.1186/2043-9113-4-10. [p81, 84, 97]

H. Wickham. testthat: Get started with testing. *The R Journal*, 3:5–10, 2011. URL https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf. [p89]

H. Wickham, J. Hester, W. Chang, and J. Bryan. *devtools: Tools to Make Developing R Packages Easier*, 2022. URL https://CRAN.R-project.org/package=devtools. R package version 2.4.5. [p89]

H. Wickham, P. Danenberg, G. Csárdi, and M. Eugster. *roxygen2: In-Line Documentation for R*, 2025. URL https://CRAN.R-project.org/package=roxygen2. R package version 7.3.3. [p89]

S. I. Yerlitas, S. B. Gengec, N. Kochan, G. E. Zararsiz, S. Korkmaz, and G. Zararsiz. *dtComb: Statistical Combination of Diagnostic Tests*, 2025. URL https://github.com/gokmenzararsiz/dtComb. R package version 1.0.7. [p98]

W. Yu and T. Park. Two simple algorithms on linear combination of multiple biomarkers to maximize partial area under the roc curve. *Computational Statistics & Data Analysis*, 88: 15–27, 2015. URL https://doi.org/10.1016/j.csda.2014.12.002. [p80, 90, 97]

G. Zararsiz, H. Y. Akyildiz, D. GÖKSÜLÜK, S. Korkmaz, and A. ÖZTÜRK. Statistical learning approaches in diagnosing patients with nontraumatic acute abdomen. *Turkish Journal of Electrical Engineering and Computer Sciences*, 24(5):3685–3697, 2016. URL https://doi.org/10.3906/elk-1501-181. [p81, 87, 90]

S. Ilayda Yerlitaş Taştan
Erciyes University

Department Biostatistics
Kayseri, Türkiye
ORCiD: 0000-0003-2830-3006
ilaydayerlitas340@gmail.com

Serra Bersan Gengeç
Erciyes University
Department Biostatistics
Kayseri, Türkiye
serrabersan@gmail.com

Necla Koçhan
Izmir University of Economics
Department of Mathematics
İzmir, Türkiye
ORCiD: 0000-0003-2355-4826
necla.kayaalp@gmail.com

Gözde Ertürk Zararsız
Erciyes University
Department Biostatistics
Kayseri, Türkiye
ORCiD: 0000-0002-5495-7540
gozdeerturk9@gmail.com

Selcuk Korkmaz
Trakya University
Department Biostatistics
Edirne, Türkiye
ORCiD: 0000-0003-4632-6850
selcukkorkmaz@trakya.edu.tr

Gökmen Zararsız
Erciyes University
Department Biostatistics
Kayseri, Türkiye
ORCiD: 0000-0001-5801-1835
gokmen.zararsiz@gmail.com