

easyROC: An Interactive Web-tool for ROC Curve Analysis Using R Language Environment

by Dincer Goksuluk, Selcuk Korkmaz, Gokmen Zararsiz and A. Ergun Karaagaoglu

Abstract ROC curve analysis is a fundamental tool for evaluating the performance of a marker in a number of research areas, e.g., biomedicine, bioinformatics, engineering etc., and is frequently used for discriminating cases from controls. There are a number of analysis tools which are used to guide researchers through their analysis. Some of these tools are commercial and provide basic methods for ROC curve analysis while others offer advanced analysis techniques and a command-based user interface, such as the R environment. The R environment includes comprehensive tools for ROC curve analysis; however, using a command-based interface might be challenging and time consuming when a quick evaluation is desired; especially for non-R users, physicians etc. Hence, a quick, comprehensive, free and easy-to-use analysis tool is required. For this purpose, we developed a user-friendly web-tool based on the R language. This tool provides ROC statistics, graphical tools, optimal cutpoint calculation, comparison of several markers, and sample size estimation to support researchers in their decisions without writing R codes. easyROC can be used via any device with an internet connection independently of the operating system. The web interface of easyROC is constructed with the R package **shiny**. This tool is freely available through www.biosoft.hacettepe.edu.tr/easyROC.

Introduction

The receiver operating characteristics (ROC) curve is a graphical approach used to visualize and assess the performance of a binary classifier system. This unique feature of ROC curve analysis makes it one of the most extensively used methods in various fields of science. It was originally developed during World War II to detect whether a signal on the radar screen represented an object or a noise (Egan, 1975; Swets et al., 2000; Fan et al., 2006) and today it is widely used in medicine, radiology, biometrics, bioinformatics and various applications of machine learning and data mining research (Fawcett, 2006; Sonego et al., 2008). ROC curve analysis can be implemented for several reasons: (i) to assess the overall performance of a classifier using several performance measures, (ii) to compare the performances of classifiers, and (iii) to determine the optimal cutpoint for a given classifier, diagnostic test or marker/biomarker. For simplicity of language, we will use the terms classifier and diagnostic test throughout the manuscript. The performance of a classifier can be summarized using the point estimations and confidence intervals of several basic performance measures such as sensitivity, specificity or combined measures of sensitivity and specificity such as likelihood ratios, accuracy, area under the ROC curve (AUC), etc. A ROC curve is basically a plot of a classifier's true positive rates (TPR: sensitivity) versus false positive rates (FPR: $1 - \text{specificity}$) where each point is generated by a different threshold value, i.e., cutpoint. For the simplicity of equations, we will use the terms TPR and FPR in the equations. One of the major tasks is to determine the optimum cutpoint value which corresponds to the reasonable TPR and FPR values. The determination of an optimum value is usually a trade-off between performance measures. The ROC curve is used to find the optimal cutpoint located on the curve which is the closest point to the top-left corner. However, finding the "optimum" cutpoint is not always based on maximizing the sensitivity and specificity. It is reasonable to select an optimum cutpoint value by regarding alternative selection criteria such as maximization of predictive values, diagnostic odds ratio, etc.

There are a number of commercial (e.g., IBM SPSS, MedCalc, Stata, etc.) and open-source (R) software packages which are used to guide researchers through their ROC curve analysis. Some of these software packages provide basic features for ROC curve analysis while others, such as R, offer advanced features but also a command-based user interface. The R environment includes comprehensive tools for ROC curve analysis, such as **ROCR** (Sing et al., 2005), **pROC** (Robin et al., 2011), **ROC** (Carey and Redestig, 2015) and **OptimalCutpoints** (Lopez-Raton et al., 2014).

All of the R packages mentioned above perform ROC curve analysis using the related package functions. Although these packages are comprehensive and flexible, they require a good programming knowledge of the R language. However, working with a command-based interface might be challenging and time consuming when a quick evaluation is desired especially for non-R users, such as physicians and other health care professionalists. Fortunately, an R package **shiny** (Chang et al., 2015) allows users to create interactive web-tools with a nicely designed, user-friendly and easy-to-use user interface. In this context, we developed a web-tool, easyROC, for ROC curve analysis. The

user interface of easyROC is constructed via **shiny** and HTML codes. easyROC combines several R packages for ROC curve analysis. This tool has three main parts including ROC statistics, cutpoint calculations and sample size estimation. Detailed information about easyROC and the related methods together with mathematical background are given in Section **Material and methods**. easyROC is freely available at <http://www.biosoft.hacettepe.edu.tr/easyROC> and all the source codes are on GitHub¹.

Material and methods

Theory behind ROC analysis

Let us consider the binary classification problem where X denotes the value of the classifier for cases and controls. Consider the values of controls distributed as $X_0 \sim G_0(\cdot)$ and cases as $X_1 \sim G_1(\cdot)$. Let $\hat{Y} = \{0, 1\}$ be the estimated class labels of the subjects for a given threshold value c as given in Equation 1.

$$\hat{Y} = \begin{cases} 1, & \text{if } X \geq c \\ 0, & \text{if } X < c \end{cases} \quad (1)$$

Parametric ROC curve. The parametric ROC curve is plotted using the FPR (1 – Specificity) and TPR (Sensitivity) values given in Equation 2 for all possible cutpoints of a classifier.

$$\begin{aligned} FPR_c &= P(X \geq c | Y = 0) = \int_c^\infty G_0(x) dx \\ TPR_c &= P(X \geq c | Y = 1) = \int_c^\infty G_1(x) dx \end{aligned} \quad (2)$$

When the distribution of the classifier is *Normal*, the parametric ROC curve is fitted using *binormal* ROC properties. Suppose $X_0 \sim Normal(\mu_0, \sigma_0^2)$ and $X_1 \sim Normal(\mu_1, \sigma_1^2)$. The ROC curve is the function of FPRs; as in Equation 3.

$$ROC(t) = \Phi\left(a + b\Phi^{-1}(t)\right), \quad (3)$$

where $a = (\mu_1 - \mu_0)/\sigma_1$, $b = \sigma_0/\sigma_1$, $t = FPR_c$ and Φ is the cumulative distribution function of the standard normal distribution (Zhou et al., 2002). The area under the curve is calculated using Equation 4.

$$AUC = \int_0^1 ROC(t) dt = \Phi\left(\frac{a}{\sqrt{1+b^2}}\right) \quad (4)$$

Fitting the ROC curve by using Equation 3 has two major drawbacks: (i) incorrect ROC curves may arise when the underlying distribution is not normal, (ii) ROC lines are improper when within class variations are not similar, i.e., heteroscedasticity. An example of improper ROC curves is given in Figure 1. To overcome these problems, one may nonparametrically fit the ROC curve without considering distributional assumptions or use parametric/semiparametric alternatives to the binormal model (Gönen and Heller, 2010).

Nonparametric ROC curve. Consider the estimated class labels in Equation 1. The FPR and TPR given in Equation 2 are estimated; as given in Equation 5.

¹<http://www.github.com/dncR/easyROC>

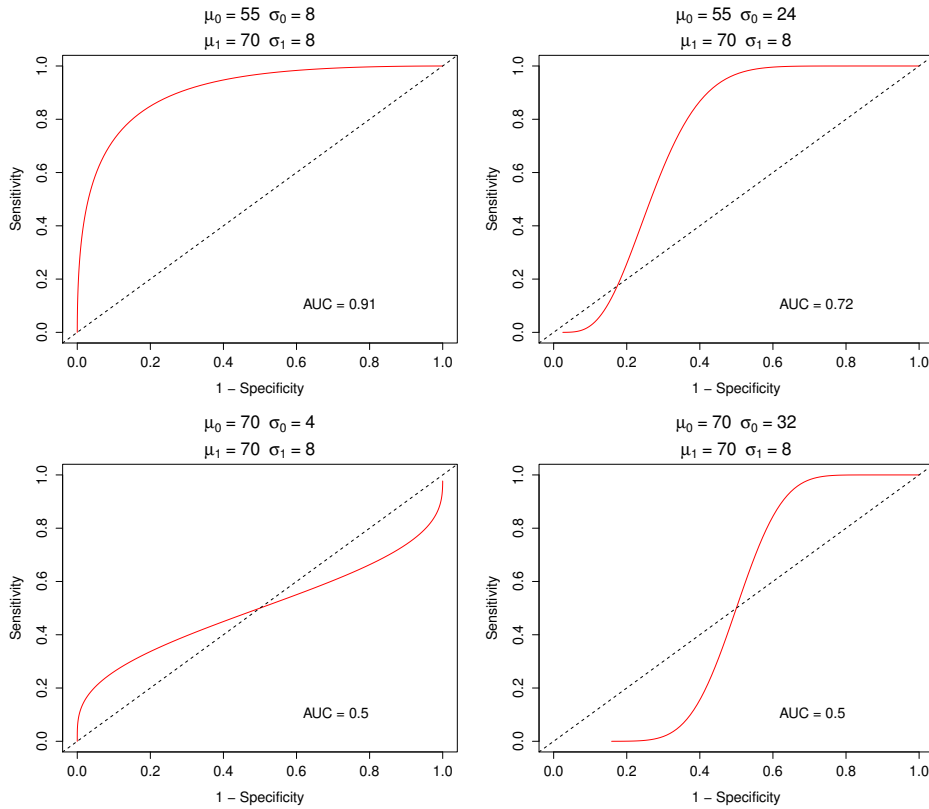


Figure 1: Parametric ROC curves.

$$\widehat{FPR}_c = \frac{1}{n_0} \sum_{j=1}^{n_0} I [X_{0j} \geq c]$$

$$\widehat{TPR}_c = \frac{1}{n_1} \sum_{i=1}^{n_1} I [X_{1i} \geq c] \tag{5}$$

The empirical ROC curve is plotted using \widehat{FPR}_c and \widehat{TPR}_c and the area under the curve, given in Equation 6, is estimated by summing the trapezoids enclosed by the points of the ROC curve. The nonparametric AUC is related to the Mann-Whitney statistic of the rank-sum test (Bamber, 1975; Hanley and McNeil, 1982).

$$\widehat{AUC} = \frac{1}{n_0 n_1} \sum_{j=1}^{n_0} \sum_{i=1}^{n_1} \Psi (X_{1i}, X_{0j}), \tag{6}$$

where $\Psi = 0$ if $X_0 > X_1$, $\Psi = 1$ if $X_1 > X_0$ and $\Psi = 1/2$ if $X_0 = X_1$.

Performance measures and optimal cutpoints. The predicted and actual classes, i.e., gold standard test results, can be shown with a 2×2 cross table; as seen in Table 1. The performance of a classifier is basically measured using the total proportion of true positive (TP) and true negative (TN) cases. By using Table 1, several performance measures are also calculated. Among these performance measures, we focused on the measures given in Table 1 which are widely used and well-known. The optimal cutpoint is determined by using one or more performance measures together. An ideal cutpoint, for example, might be selected by maximizing the sensitivity and specificity of a classifier. A classifier with perfect discriminative ability would have sensitivity and specificity measures equal to 1. Hence, the area under the curve for a perfect separation will be equal to 1.

Although researchers are usually interested in the overall diagnostic performance of a classifier, it is sometimes useful to focus on a portion of the ROC curve to compute the partial AUCs (pAUC). pAUC is an extension of the AUC measure which considers the trapezoids within a given interval of

Predicted Labels	Actual Labels		Total
	Positive ($Y = 1$)	Negative ($Y = 0$)	
Positive ($\hat{Y} = 1$)	TP	FP	TP + FP
Negative ($\hat{Y} = 0$)	FN	TN	FN + TN
Total	TP + FN	FP + TN	n

TP: True positive
 FP: False positive
 TN: True negative
 FN: False negative
 NPV: Negative predictive value
 PPV: Positive predictive value
 PLR: Positive likelihood ratio
 NLR: Negative likelihood ratio

Sensitivity = $TP / (TP + FN)$
 Specificity = $TN / (FP + TN)$
 PPV = $TP / (TP + FP)$
 NPV = $TN / (TN + FN)$
 PLR = $Sensitivity / (1 - Specificity)$
 NLR = $(1 - Sensitivity) / Specificity$

Table 1: A 2×2 classification table and performance measures.

sensitivity and/or specificity. Let us consider the pAUC where specificity (or sensitivity) lies within the interval $[t_1, t_2]$. The pAUC is calculated by taking the integral (parametric) as given in Equation 7 or by summing the trapezoids within the interval (nonparametric).

$$pAUC(t_1, t_2) = \int_{t_1}^{t_2} ROC(x) dx \tag{7}$$

As the interval $[t_1, t_2]$ converges to $[0, 1]$, the pAUC will converge to the overall AUC. The best classifier can be selected using either AUC or pAUC values.

Identification of the optimal cutpoint is an important task to avoid incorrect conclusions. Various methods are available in the literature to determine the optimal cutpoint. Most of these methods are based on the sensitivity and specificity measures. However, other methods are also available based on cost-benefit, prevalence, predictive values and diagnostic likelihood ratios. Two popular methods are, for example, the Youden index and the minimization of the distance of the point on the curve to the top-left corner, i.e., the point indicating perfect discrimination.

$$Youden(c) = \max\{TPR_c - FPR_c\} \tag{8}$$

Table 1 gives the list of optimal cutpoint methods we consider in easyROC. For detailed information and mathematical background, see Lopez-Raton et al. (2014).

Statistical inference. A common subject of interest in ROC analysis is to compare the performances of several classifiers to select the best one to discriminate cases from controls. For a classifier with random chance discrimination ability, the equation $TPR = FPR$ holds. In that case, the area under the curve is 0.50. Hence, the discrimination ability of a classifier is mostly tested against the value 0.50.

$$\begin{aligned}
 H_0 : AUC &= 0.50 \\
 H_1 : AUC &\neq 0.50
 \end{aligned}$$

Under the large sample theory, the significance of AUC is tested using the *Wald* test statistic as given in Equation 9.

$$z = \frac{\widehat{AUC} - AUC}{\text{Var}(\widehat{AUC})^{1/2}} \tag{9}$$

When the parametric approach is used, the variance of AUC is estimated using Equation 10 (McClish, 1989; Zhou et al., 2002).

$$\text{Var}(\widehat{AUC}) = f^2 \text{Var}(\hat{a}) + g^2 \text{Var}(\hat{b}) + 2fg \text{Cov}(\hat{a}, \hat{b}), \tag{10}$$

where

$$f = \frac{e^{-a^2/2(1+b^2)}}{\sqrt{2\pi(1+b^2)}} \quad \text{and} \quad g = -\frac{abe^{-a/2(1+b^2)}}{\sqrt{2\pi(1+b^2)^3}} \tag{11}$$

and the estimated variances for a and b as follows:

$$\begin{aligned} \widehat{\text{Var}}(\hat{a}) &= \frac{n_1(\hat{a}^2 + 2) + 2n_0\hat{b}^2}{2n_0n_1}, \\ \widehat{\text{Var}}(\hat{b}) &= \frac{(n_1 + n_0)\hat{b}^2}{2n_0n_1}, \\ \widehat{\text{Cov}}(\hat{a}, \hat{b}) &= \frac{\hat{a}\hat{b}}{2n_0}. \end{aligned} \tag{12}$$

The estimated values of a and b are used in Equation 11. A number of methods have been proposed for the estimation of the variance of AUC when the nonparametric approach is used. In this paper, we will focus on the methods described below:

1. Mann-Whitney version of rank-sum test:

Hanley and McNeil (1982) propose the variance estimation given in Equation 13. This method estimates the variance using an approximation based on exponential distribution as

$$\text{Var}(\widehat{AUC}) = \frac{1}{n_0n_1} \left\{ AUC(1 - AUC) + (n_1 - 1)(Q_1 - AUC^2) + (n_0 - 1)(Q_2 - AUC^2) \right\}, \tag{13}$$

where $Q_1 = \widehat{AUC}/(2 - \widehat{AUC})$ and $Q_2 = 2\widehat{AUC}^2/(1 + \widehat{AUC})$. The Mann-Whitney version might underestimate the variance when the area is nearly 0.5 and overestimate it when the area is close to 1 (Hanley and McNeil, 1982; Hanley and Hajian-Tilaki, 1997; Obuchowski, 1994). This estimate is mostly used in sample-size estimation.

2. DeLong et al. (1988)'s estimate:

Since the exponential distribution approximation in Equation 13 gives biased variance estimates, DeLong et al. (1988) suggest an alternative method which is free from distributional assumptions. Define the components T_{1i} for the i th subject from cases and T_{0j} for the j th subject from controls as follows:

$$\begin{aligned} \psi(T_{1i}) &= \frac{1}{n_0} \sum_{j=1}^{n_0} \Psi(X_{1i}, X_{0j}) \quad i = 1, 2, \dots, n_1 \\ \psi(T_{0j}) &= \frac{1}{n_1} \sum_{i=1}^{n_1} \Psi(X_{1i}, X_{0j}) \quad j = 1, 2, \dots, n_0 \end{aligned} \tag{14}$$

Using the Equation 14 the variance of AUC is estimated as

$$\text{Var}(\widehat{AUC}) = \frac{1}{n_1} S_{T_1}^2 + \frac{1}{n_0} S_{T_0}^2, \tag{15}$$

where $S_{T_1}^2$ and $S_{T_0}^2$ are variance estimates of T_1 and T_0 as in Equation 16.

$$S_{T_i}^2 = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} [\psi(T_{ij}) - \widehat{AUC}]^2 \quad i = 0, 1 \tag{16}$$

3. Normal approximation of binomial proportion:

Another alternative for variance estimation is to use binomial approximation under the large sample theory, as given in Equation 17. For small samples, this method may give biased estimates.

$$\text{Var}(\widehat{AUC}) = \frac{AUC(1 - AUC)}{n_0 + n_1} \tag{17}$$

The estimated variance derived from one of the methods described above is used to construct the confidence intervals of the AUC. A common method is to use large sample approximation as below:

$$\widehat{AUC} - z_{1-\alpha/2}\text{Var}(\widehat{AUC})^{1/2} < AUC < \widehat{AUC} + z_{1-\alpha/2}\text{Var}(\widehat{AUC})^{1/2}. \tag{18}$$

When the area under the curve is close to 1 or the sample size is relatively small, the large sample approximation in Equation 18 produces improper confidence intervals since the upper limit exceeds 1. To solve this problem, Agresti and Coull (1998) proposed the *score confidence interval* that guarantees the upper limit is less than or equal to 1. Another alternative is to construct the binomial exact confidence intervals given in Equation 19 using the relationship between binomial and *F*-distribution (Morisette and Khorram, 1998)

$$\frac{1}{1 + \frac{n-x+1}{x} F_{2(n-x+1), 2x, \alpha/2}} \leq p \leq \frac{\frac{x+1}{n-x} F_{2(x+1), 2(n-x), \alpha/2}}{1 + \frac{x+1}{n-x} F_{2(x+1), 2(n-x), \alpha/2}}, \tag{19}$$

where $p = x/n$ is the binomial proportion such as sensitivity, specificity and AUC.

Sample size calculation. In most studies, determining the required sample size is an important step for the research to be able to detect significant results. Sample size determination is required for both constructing the confidence interval of the unknown population parameter and testing a research hypothesis. Obuchowski (1998) reviewed sample size determination for several study designs. In this paper, we cover the sample size determination for three types of studies based on AUCs. In addition, the following sample size calculations can be extended to other performance measures such as sensitivity, specificity, etc.

The variance estimates of AUCs can be obtained using one of the Equations 13, 15 and 17. While Equation 13 is a good approximation for a variety of underlying distributions, the estimated variance will be underestimated if the test results are in a discrete rating format. To overcome this problem, Obuchowski (1998) and Obuchowski et al. (2004) suggest an alternative variance estimation method for rating data using the variance function as given in Equation 20 which is based on an underlying binormal distribution. In this section, we focused on sample size calculation for discrete scale data. However, the same formulas are valid for continuous scale diagnostic tests since the only difference is about estimating the variance of diagnostic test accuracy.

$$V(\widehat{AUC}) = 0.0099 e^{-a^2/2} \times [(5a^2 + 8) + (a^2 + 8)/R], \tag{20}$$

where $a = \sqrt{2} \Phi^{-1}(AUC)$ and $R = n_0/n_1$ is the allocation ratio, i.e., the ratio of the number of controls to the number of cases. The estimated variance is then $\text{Var}(\widehat{AUC}) = V(\widehat{AUC})/n_1$. The total sample size is equal to $n = n_1(1 + R)$. One of the variance estimations from Equations 13, 15, 17 and 20 is used for the sample size calculations. The selection of the appropriate variance estimation method is based on the variable type of the test results and underlying distributions.

1. Hypothesis test to determine the AUC of a single classifier:

In most of the studies with a single classifier, the aim of the study is to determine whether the diagnostic test performs well for discriminating diseased patients from controls. Consider the hypotheses $H_0 : AUC = 0.5$ versus $H_1 : AUC > 0.5$ (i.e, one-sided test). The required number of cases is determined using Equation 21 (Obuchowski et al., 2004).

$$n_1 = \frac{\left[z_{1-\alpha} \sqrt{\text{Var}_0(\widehat{AUC})} + z_{1-\beta} \sqrt{\text{Var}_1(\widehat{AUC})} \right]^2}{(AUC - 0.5)^2} = \frac{\left[z_{1-\alpha} \sqrt{0.0792 \times (1 + 1/R)} + z_{1-\beta} \sqrt{\text{Var}_1(\widehat{AUC})} \right]^2}{(AUC - 0.5)^2}, \tag{21}$$

where Var_0 and Var_1 are the variance estimations under the null and alternative hypotheses using Equation 20. $z_{1-\alpha}$ and $z_{1-\beta}$ are lower-tailed percentile values of the cumulative standard normal distribution. Finally, the total sample size is obtained using $n = n_1 + n_1 \times R$.

2. Comparing the AUCs of two classifiers:

When the aim of a study is to compare two classifiers, one may consider the hypotheses $H_0 : AUC_1 = AUC_2$ versus $H_1 : AUC_1 \neq AUC_2$. The two classifiers will be equally performing under the null hypothesis. The required number of cases is calculated using Equation 22.

$$n_1 = \frac{\left[z_{1-\alpha/2} \sqrt{\text{Var}_0(\widehat{AUC}_1 - \widehat{AUC}_2)} + z_{1-\beta} \sqrt{\text{Var}_1(\widehat{AUC}_1 - \widehat{AUC}_2)} \right]^2}{(AUC_1 - AUC_2)^2}, \tag{22}$$

where Var_0 and Var_1 are the variance estimations under the null and alternative hypotheses; as given in Equation 23 (Zhou et al., 2002; Obuchowski et al., 2004).

$$\text{Var}(\widehat{AUC}_1 - \widehat{AUC}_2) = \text{Var}(\widehat{AUC}_1) + \text{Var}(\widehat{AUC}_2) - 2\text{Cov}(\widehat{AUC}_1, \widehat{AUC}_2) \tag{23}$$

The total sample size is calculated using the allocation ratio. When two classifiers are performed on the same subjects, the design will be *paired* yielding the covariance term to be a nonzero (usually positive) quantity. However, the covariance term will be zero (i.e., independent classifiers) if each test is performed on different subjects. Detailed information on the calculation of the covariance term can be found in Zhou et al. (2002).

3. Non-inferiority of a new classifier to a standard one:

In addition to comparing two classifiers, some studies are designed to explore the performance of a new classifier to that of a standard one. The new classifier should perform as well as but not necessarily better than the standard test (Obuchowski et al., 2004). The hypotheses are $H_0 : AUC_{std} - AUC_{new} \geq \Delta$ versus $H_1 : AUC_{std} - AUC_{new} < \Delta$. The required number of cases is calculated using Equation 24

$$n_1 = \frac{(z_{1-\alpha} + z_{1-\beta})^2 \text{Var}_1(\widehat{AUC}_{std} - \widehat{AUC}_{new})}{(AUC_{std} - AUC_{new} - \Delta)^2}, \tag{24}$$

where Δ is the non-inferiority margin, i.e., the minimum acceptable difference between the AUCs of the standard and new classifiers.

Current ROC analysis tools and easyROC

ROC curve analysis is one of the standard procedures included in most statistical analysis tools such as IBM SPSS, Stata, MedCalc and R. Each tool offers different features within ROC curve analysis. Among commercial software packages, IBM SPSS, which is one of the most widely used commercial software packages, plots the ROC curve and computes some basic statistics such as AUC and its standard error, confidence interval and statistical significance. However, it does not provide any method for sample size calculation or cutpoint determination. Stata offers a variety of calculations for ROC curve analysis including partial AUC, multiple comparisons of ROC curves, optimal cutpoint determination using the Youden index and several performance measures. Another commercial software alternative for ROC curve analysis is MedCalc, which has comprehensive features compared to most of the other available commercial software packages and is especially developed for biomedical research. MedCalc provides sample size estimation for a single diagnostic test, but it does not have an option for pAUC calculation.

Unlike commercial software packages, R is an open source and free software package that includes all the features of commercial software packages and more through several packages such as **ROC**,

	IBM SPSS	Stata	MedCalc	ROC	ROCR	pROC	easyROC
Plots	Yes	Yes	Yes*	Yes	Yes*	Yes*	Yes*
Conf. intervals	Yes	Yes*	Yes	Yes	Yes	Yes*	Yes*
pAUC	No	Yes	No	Yes	Yes	Yes*	Yes*
Statistical tests	No	Yes	No	Yes	Yes	Yes*	Yes*
Diagnostic measures	No	Yes	Yes	No	Yes*	Yes	Yes
Multiple comp.	No	Yes	Yes*	No	No	Yes*	Yes
Cutpoints	No	Yes	Yes	No	No	Yes	Yes*
Sample size	No	No	Yes	No	No	Yes	Yes*
Free license	No	No	No	Yes*	Yes*	Yes*	Yes*
Open source	No	No	No	Yes*	Yes*	Yes*	Yes*
Web-tool access	No	No	No	No	No	No	Yes*
User interface	Yes	Yes*	Yes*	No	No	Yes*	Yes*

* Comprehensive ones.

Table 2: Comparison of easyROC with other tools.

ROCR, pROC and OptimalCutpoints. **ROC** is an R/Bioconductor package which can plot the ROC curve and calculate the AUC. It also calculates pAUCs based on false positive rates. This package is originally developed to be used for the ROC analysis with DNA microarrays. **ROCR** is a comprehensive R package providing over 25 different performance measures (based on package version 1.0-7). It allows users to create two dimensional performance curves. Although **ROCR** is one of the most comprehensive packages for assessing the performance measures, it provides limited options to select the optimum cutpoint. One may use any of the two-dimensional performance graphs to determine the optimal cutpoint graphically. It computes the AUC and its confidence interval, however, it does not provide a statistical test for performance measures.

pROC, on the other hand, offers more comprehensive and flexible features than its free and commercial counterparts. It performs statistical tests for the comparison of ROC curves using [DeLong et al. \(1988\)](#), [Venkatraman and Begg \(1996\)](#) and [Venkatraman \(2000\)](#) for AUC, and [Hanley and McNeil \(1983\)](#) and [Pepe et al. \(2009\)](#) for both AUC and pAUC. It also calculates the confidence intervals for the sensitivity, specificity, ROC curves, pAUC, and smoothed ROC curves. The confidence intervals are computed using [DeLong et al. \(1988\)](#)'s method for AUCs and using bootstrap for pAUCs, sensitivity and specificity at given threshold(s). Bootstrap confidence intervals and pAUC regions are shown in the ROC curve plot. Several diagnostic measures, such as sensitivity, specificity, negative and positive predictive values, are computed for a given threshold. Like **ROCR**, **pROC** also offers limited features for detecting the optimal cutpoint. Two methods, i.e., Youden index and closest point to the top-left corner, are available to find the optimal cutpoint. In addition, **pROC** is an alternative among the ROC packages on CRAN to find the required sample size for a single diagnostic test or the comparison of two diagnostic tests. Two versions of **pROC** are available: (i) for the R programming language and (ii) with a graphical user interface for the S-PLUS statistical software package.

There are several packages providing optimal cutpoint calculations through R. **OptimalCutpoints** is a sophisticated R package specifically developed to determine the optimal cutpoint of a test or biomarker ([Lopez-Raton et al., 2014](#)). It includes 34 different cutpoint calculation methods based on sensitivity/specificity measures, cost-benefit analysis, predictive values, diagnostic likelihood ratios, prevalences and *p*-values. A brief description of these methods is given in Supplementary 1. Although these R packages, especially **pROC**, seem to be a perfect match for ROC curve analysis, none of them has a graphical user interface and all require coding knowledge, which makes them hard to use; especially for non-R users.

Another R package worth mentioning is **plotROC** ([Sachs, 2016](#)) which is available on CRAN and also for **shiny** platforms. **plotROC** is a flexible and sophisticated R package which can be used to create nice-looking and interactive ROC graphs. Unlike the packages described above, **plotROC** has a web-based user interface which is very useful for non-R users. Researchers can use its web service to create ROC graphics and download the figures to their local computer. However, it does not provide any statistical tests or sample size calculations.

easyROC aims to extend the features of several ROC packages in R and allows researchers to conduct their ROC curve analysis through a single and easy-to-use interface without writing any R code. This tool is a web-based application created via **shiny** and HTML programming. easyROC makes use of the R packages **plyr** ([Wickham, 2011](#)), **pROC** and **OptimalCutpoints** for conducting ROC analysis. **plyr** is used for manipulating data while **pROC** is used for estimation and hypothesis testing of pAUCs. easyROC has comprehensive options for ROC curve analysis which other tools do not have

Modules (Tab panels)	Features
ROC curve	<ul style="list-style-type: none"> • Parametric/Nonparametric ROC • AUC, pAUC <ul style="list-style-type: none"> – Confidence interval (Exact and Asymptotic) – Significance test (Wald) • Standard error estimation <ul style="list-style-type: none"> – DeLong (1988) – Mann-Whitney – Binomial approximation • Multiple comparison of AUCs <ul style="list-style-type: none"> – Bonferroni – False discovery rate • ROC plot (customizable)
Cutpoints	<ul style="list-style-type: none"> • 34 different methods for optimum cutpoints (Lopez-Raton et al., 2014) • Performance measures with confidence intervals <ul style="list-style-type: none"> – Exact CIs – Asymptotic CIs • Cutpoint graphs (fully customizable) <ul style="list-style-type: none"> – ROC curve – Sensitivity & specificity plot – Density plot – Scatter plot
Sample size	<ul style="list-style-type: none"> • Single diagnostic test • Comparison of two diagnostic tests • Noninferiority of a new test to a standard test

Table 3: Features of easyROC.

(or partially shares some features). The ROC curve can be estimated using parametric or nonparametric approaches. It offers four different methods for the calculation of the standard error and confidence interval of the AUC. Researchers can calculate the pAUCs based on sensitivity and specificity, if necessary. One may perform pairwise comparisons to find the classifiers which have similar or different discrimination ability. However, the pairwise comparison should be carried out carefully since the type I error increases with the increasing number of comparisons. easyROC offers multiple test corrections in order to keep type I error at a given level. Multiple comparisons of diagnostic tests can be applied using either Bonferroni or false discovery rate correction. Furthermore, the optimal cutpoints are determined using the methods from **OptimalCutpoints** and the corresponding measures at a given cutpoint, including sensitivity, specificity, positive and negative predictive values, and positive and negative likelihood ratios are also returned. One can determine the desired sample size for ROC curve analysis using this tool for three different cases. All these comprehensive features are accessible through a graphical user interface, which makes the analysis process easier for all users. The comparison with other tools is given in Table 2 and the features of each module are given in Table 3.

Results

Case study on non-alcoholic fatty liver disease

To illustrate our application, we used the non-alcoholic fatty liver disease (NAFLD) dataset of [Celikbilek et al. \(2014\)](#). This study was designed to identify the non-invasive miRNA biomarkers of NAFLD. The authors obtained the serum samples of 20 healthy and 20 NAFLD observations and quantified the expression levels of eight miRNAs using quantitative Real-Time PCR (qPCR) technology. After performing the necessary statistical analysis, the authors revealed that miR-197, miR-146b, miR-181d and miR-99a may be potential biomarkers in identifying NAFLD. The normalized expression values

easyROC: a web-tool for ROC curve analysis (ver. 1.3)

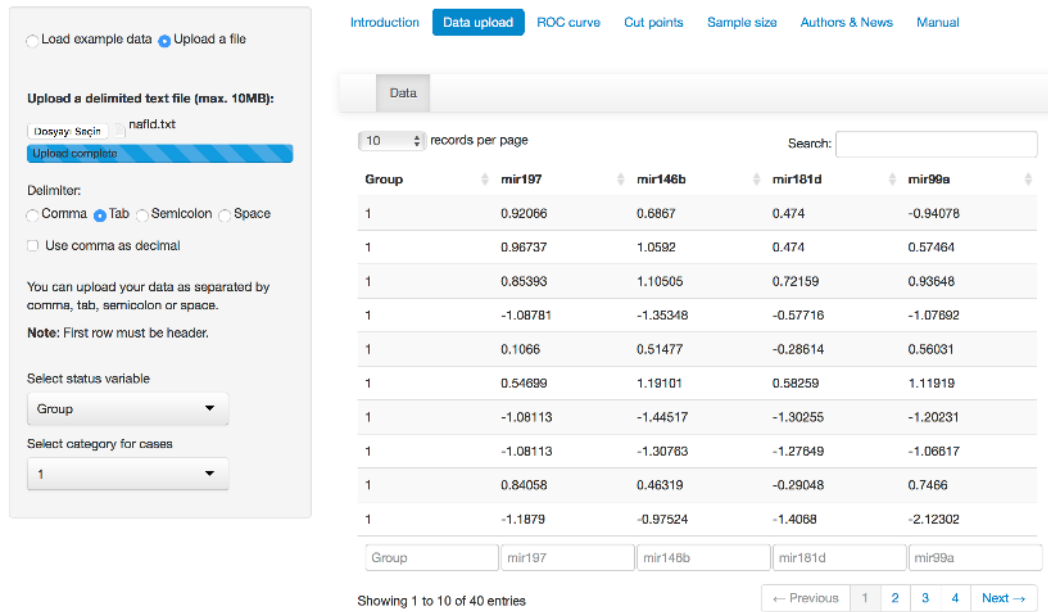


Figure 2: Uploading data into easyROC.

of these miRNAs and the class information (the column named “Group”, where 0 refers to controls and 1 refers to cases) of each observation are given in Supplementary 2. This file can be directly used as input to the easyROC web-tool and users can arrange their own data based on this file. Two example datasets, Mayo and PBC (Murtaugh et al., 1994), are also available in the web-tool for users to practice the application. In our example, the aim is to investigate the discriminative performances of each miRNA, compare each other and identify the optimal cutpoints for each miRNA in identifying NAFLD.

Implementation of easyROC web-tool

The data are uploaded to the easyROC interface using the *Data upload* tab (Figure 2). easyROC accepts a delimited text file with variable names in the first row. The status variable is also set by the same tab panel. easyROC automatically detects the variable names and exports them into related fields. When data are correctly uploaded, researchers may proceed with ROC curve analysis, cutpoint estimations or sample size calculations. The area under the curve, confidence intervals and significance tests for AUC, multiple comparisons (if multiple markers are selected) and pAUCs are calculated with the *ROC curve* tab (Figures 3 and 4). The ROC curve is estimated using the nonparametric approach. The advanced option allows researchers to select a method for standard error estimation and confidence intervals. easyROC selects the DeLong et al. (1988) method by default.

Here, we select mir197, mir146b, mir181d and mir99a miRNAs to assess their performances and to compare them with each other in identifying NAFLD. Since the expression levels of all miRNAs are underexpressed in the NAFLD group, lower values will indicate higher risk and therefore we should uncheck the “Higher values indicate higher risks” box. Using DeLong et al. (1988) standard error estimations, we obtained the ROC curves for each miRNA biomarker and AUC values as 0.86 (0.75–0.97), 0.77 (0.61–0.92), 0.76 (0.60–0.93) and 0.75 (0.59–0.91) for mir181d, mir197, mir99a and mir146b, respectively. The results revealed that all miRNAs’ predictive performances are significant and higher than random chance in identifying NAFLD (Figure 3). By controlling the type I error using Bonferroni correction, all pairwise comparisons showed non-significant results ($p > 0.05$). This may be due to the small sample size of the data. Increasing the sample size, thus the statistical power of the test, may concretize the predictive ability of mir181d as compared to other miRNAs.

Finding a suitable cutpoint is one of the aims of ROC curve analysis. We made use of the **OptimalCutPoints** package from R (Lopez-Raton et al., 2014), which has 34 different methods, to calculate cutpoints for each marker. An optimal cutpoint can be computed via the *Cut point* tab by selecting a marker and a method. Then, the application will calculate an optimal cutpoint and

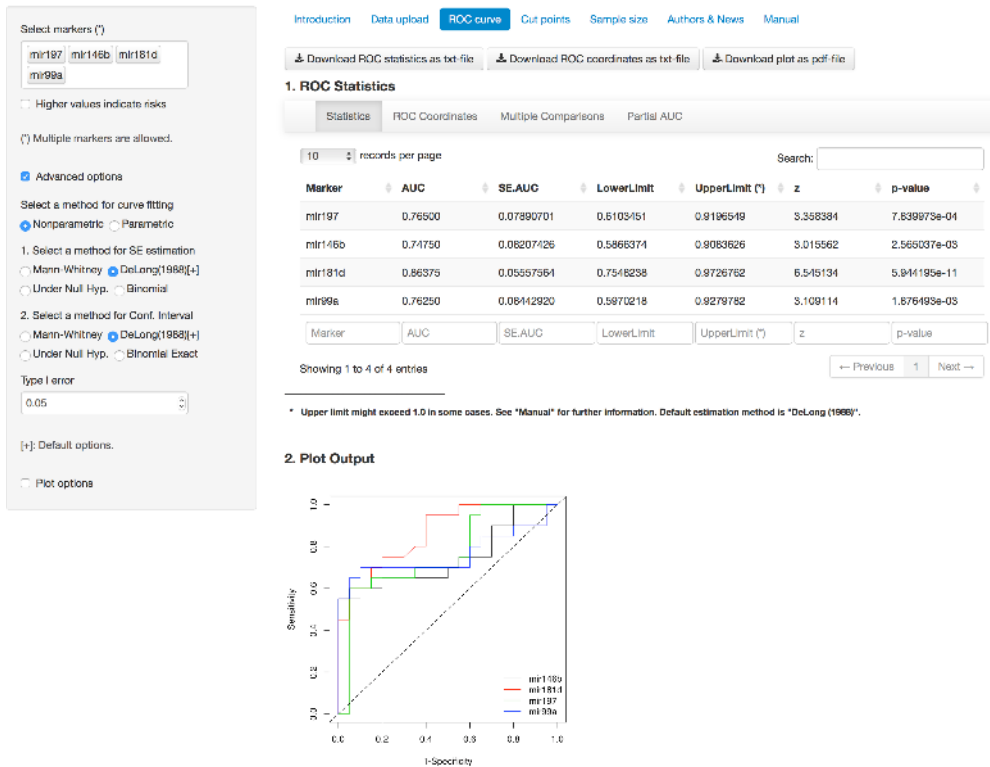


Figure 3: ROC curve analysis results.

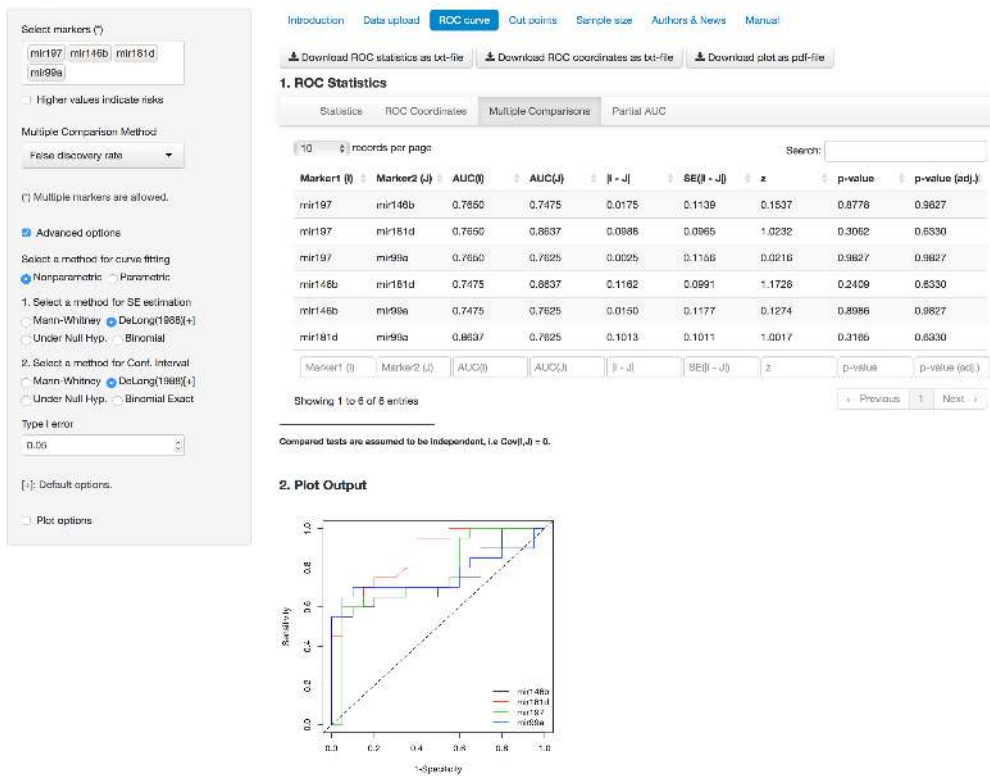


Figure 4: Multiple comparison of the diagnostic tests.

performance measures such as sensitivity, specificity, positive and negative predictive value, and positive and negative likelihood ratio based on the corresponding cutpoint value. The “ROC01” method, for example, determines the optimal cutpoint as -0.12977 for mir181d. Using this cutpoint, a new test observation with a mir181d expression level lower than this value can be assigned as an

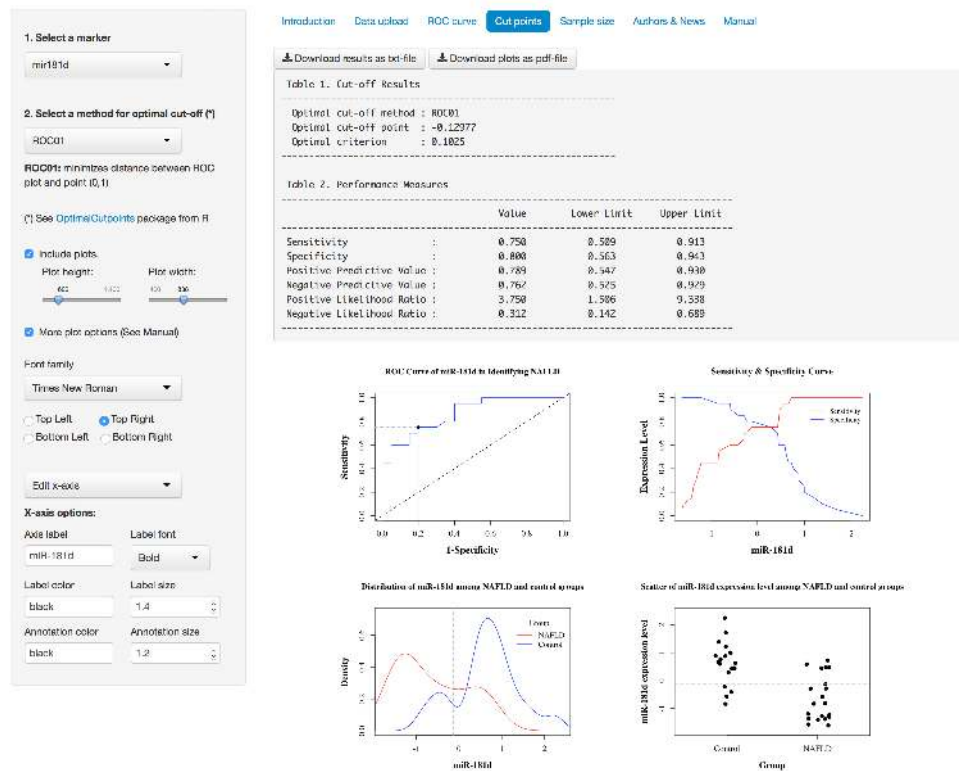


Figure 5: Determination of optimal cutpoint(s).

NAFLD patient. Based on the identified cutpoint, we obtained statistical diagnostic measures with 95% confidence intervals (Figure 5). We obtain a sensitivity of 0.75 (0.51–0.91) and specificity of 0.80 (0.56–0.94). If users select the “Include plots” option, four plots will appear under the statistics results. The first plot in the upper-left corner displays the optimal cutpoint on the ROC curve. Users can observe the change of sensitivity and specificity measures based on the value of the marker on the plot placed in the upper-right corner. The density and scatter of the expression values in each group are displayed in the bottom-left and bottom-right corners. The plots can be modified through the “More plot options” section. All the results and figures can be downloaded using the related “Download” buttons in each tab panel.

Conclusion

Since ROC curve analysis is one of the principal statistical analysis methods, it is used by a wide range of the scientific community. Both commercial and free software tools are available for users to perform it. Generally, easy-to-use and nicely-designed interfaces are offered by commercial software packages whereas flexible and comprehensive tools are available in free, open-access, code-based software packages, such as R. The first novelty of our tool is that it allows the user to use free and open-access software with an easy-to-use interface. In other words, we combine the power of an open-source and free language with a nicely designed and easily accessible interface. This tool offers more comprehensive features and a wide variety of implementations for ROC curve analysis than its commercial and free counterparts, which is another novelty of this application. It is specifically constructed for ROC curve analysis, unlike the commercial software packages, such as IBM SPSS, Stata and MedCalc.

This web-based application is intended for research purposes only, not for clinical or commercial use. Since it is a non-profit service to the scientific community, it comes with no warranty and no data security. However, since this web server uses the R package *shiny*, each user performs his/her analyses in a new R session. After uploading data, the application only saves responses within its R session and prints the results instantly. After a user has quit the application, the corresponding R session will be closed and any uploaded data, responses or outputs will not be saved locally or remotely.

This tool is freely available through <http://www.biosoft.hacettepe.edu.tr/easyROC/> and all the source codes are available at <http://www.github.com/dncR/easyROC> under GPL version 3. It will

be regularly updated upon the dependent R packages used in this application, including **shiny** and **OptimalCutpoints**, and new features will be continually added as they are developed.

Bibliography

- A. Agresti and B. A. Coull. Approximate is better than “exact” for interval estimation of binomial proportions. *The American Statistician*, 52(2):119–126, 1998. doi: 10.2307/2685469. [p218]
- D. Bamber. The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *Journal of Mathematical Psychology*, 12(4):387–415, 1975. doi: 10.1016/0022-2496(75)90001-2. [p215]
- V. Carey and H. Redestig. *ROC: Utilities for ROC, with Uarray Focus*, 2015. URL <https://www.bioconductor.org>. R package version 1.44.0. [p213]
- M. Celikbilek, M. Baskol, S. Taheri, K. Deniz, S. Dogan, G. Zararsiz, S. GURSOY, K. Guven, O. Ozbakir, M. Dundar, and M. Yucesoy. Circulating microRNAs in patients with non-alcoholic fatty liver disease. *World Journal of Hepatology*, 6(8):613–620, 2014. [p221]
- W. Chang, J. Cheng, J. Allaire, Y. Xie, and J. McPherson. *shiny: Web Application Framework for R*, 2015. URL <https://CRAN.R-project.org/package=shiny>. R package version 0.12.1. [p213]
- E. R. DeLong, D. M. DeLong, and D. L. Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. *Biometrics*, 44(3):837–845, 1988. doi: 10.2307/2531595. [p217, 220, 222]
- J. P. Egan. *Signal Detection Theory and ROC Analysis*. Series in Cognition and Perception. Academic Press, New York, NY, 1975. [p213]
- J. Fan, S. Upadhye, and A. Worster. Understanding receiver operating characteristic (ROC) curves. *Canadian Journal of Emergency Medicine*, 8:19–20, 2006. doi: 10.1017/s1481803500013336. [p213]
- T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006. doi: 10.1016/j.patrec.2005.10.010. [p213]
- M. Gönen and G. Heller. Lehmann family of ROC curves. *Medical Decision Making*, 30(4):509–517, 2010. doi: 10.1177/0272989x09360067. [p214]
- J. A. Hanley and K. O. Hajian-Tilaki. Sampling variability of nonparametric estimates of the areas under receiver operating characteristic curves: An update. *Academic Radiology*, 4(1):49–58, 1997. doi: 10.1016/s1076-6332(97)80161-4. [p217]
- J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982. doi: 10.1148/radiology.143.1.7063747. [p215, 217]
- J. A. Hanley and B. J. McNeil. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148(3):839–843, 1983. doi: 10.1148/radiology.148.3.6878708. [p220]
- M. Lopez-Raton, M. X. Rodriguez-Alvarez, C. Cadarso-Suárez, and F. Gude-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. [p213, 216, 220, 221, 222]
- D. K. McClish. Analyzing a portion of the ROC curve. *Medical Decision Making*, 9(3):190–195, 1989. doi: 10.1177/0272989x8900900307. [p216]
- J. T. Morissette and S. Khorram. Exact binomial confidence interval for proportions. *Photogrammetric Engineering and Remote Sensing*, 64(4):281–283, 1998. [p218]
- P. A. Murtaugh, E. R. Dickson, G. M. Van Dam, M. Malinchoc, P. M. Grambsch, A. L. Langworthy, and C. H. Gips. Primary biliary cirrhosis: Prediction of short-term survival based on repeated patient visits. *Hepatology*, 20(1):126–134, 1994. doi: 10.1002/hep.1840200120. [p222]
- N. A. Obuchowski. Computing sample size for receiver operating characteristic studies. *Investigative Radiology*, 29(2):238–243, 1994. doi: 10.1097/00004424-199402000-00020. [p217]
- N. A. Obuchowski. Sample size calculations in studies of test accuracy. *Statistical Methods in Medical Research*, 7(4):371–392, 1998. [p218]

- N. A. Obuchowski, M. L. Lieber, and F. H. Wians. ROC curves in clinical chemistry: Uses, misuses, and possible solutions. *Clinical Chemistry*, 50(7):1118–1125, 2004. doi: 10.1373/clinchem.2004.031823. [p218, 219]
- M. Pepe, G. Longton, and H. Janes. Estimation and comparison of receiver operating characteristic curves. *The Stata Journal*, 9(1):1, 2009. [p220]
- 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(1): 77, 2011. doi: 10.1186/1471-2105-12-77. [p213]
- M. C. Sachs. *plotROC: Generate Useful ROC Curve Charts for Print and Interactive Use*, 2016. URL <http://sachsmc.github.io/plotROC>. R package version 2.0.1. [p220]
- T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer. ROCR: Visualizing classifier performance in R. *Bioinformatics*, 21(20):3940–3941, 2005. [p213]
- P. Sonogo, A. Kocsor, and S. Pongor. ROC analysis: Applications to the classification of biological sequences and 3D structures. *Briefings in Bioinformatics*, 9(3):198–209, 2008. [p213]
- J. A. Swets, R. M. Dawes, and J. Monahan. Better decisions through science. *Scientific American*, 283(4): 82–87, 2000. doi: 10.1038/scientificamerican1000-82. [p213]
- E. Venkatraman. A permutation test to compare receiver operating characteristic curves. *Biometrics*, 56 (4):1134–1138, 2000. doi: 10.1111/j.0006-341x.2000.01134.x. [p220]
- E. Venkatraman and C. B. Begg. A distribution-free procedure for comparing receiver operating characteristic curves from a paired experiment. *Biometrika*, 83(4):835–848, 1996. doi: 10.1093/biomet/83.4.835. [p220]
- H. Wickham. The split-apply-combine strategy for data analysis. *Journal of Statistical Software*, 40(1): 1–29, 2011. URL <http://www.jstatsoft.org/v40/i01/>. [p220]
- X.-H. Zhou, D. K. McClish, and N. A. Obuchowski. *Statistical Methods in Diagnostic Medicine*, volume 569. John Wiley & Sons, 2002. doi: 10.1002/9780470317082. [p214, 216, 219]

Dincer Goksuluk
Department of Biostatistics, School of Medicine, Hacettepe University
Sihhiye Campus, 06100 - Ankara
Turkey
dincer.goksuluk@hacettepe.edu.tr

Selcuk Korkmaz
Department of Biostatistics, School of Medicine, Hacettepe University
Sihhiye Campus, 06100 - Ankara
Turkey
selcukorkmaz@hotmail.com

Gokmen Zararsiz
Department of Biostatistics, School of Medicine, Erciyes University
Genome and Stem Cell Center, Erciyes University
Talas, 38039 - Kayseri
Turkey
gokmenzararsiz@hotmail.com

A. Ergun Karaagaoglu
Department of Biostatistics, School of Medicine, Hacettepe University
Sihhiye Campus, 06100 - Ankara
Turkey
ekaraaga@gmail.com

Supplementary material

Supplementary 1: A brief description of optimal cutpoint methods.

Method	Description
Youden	Youden index identifies the cutpoint that maximizes the sum of <i>Sensitivity</i> and <i>Specificity</i> .
CB	CB is a measure based on the cost and benefit method, and is calculated from the slope of the ROC curve.
MinValueSp MinValueSe	For a given minimum value for <i>Specificity</i> , MinValueSp identifies the optimal value as the one that gives the maximum <i>Sensitivity</i> . In contrast, for a given minimum value for <i>Sensitivity</i> , MinValueSe identifies the optimal value as the one that gives the maximum <i>Specificity</i> .
ValueSp ValueSe	For a given particular value for <i>Specificity</i> , ValueSp identifies the optimal value as the one that gives the maximum <i>Sensitivity</i> . In contrast, for a given particular value for <i>Sensitivity</i> , ValueSe identifies the optimal value as the one that gives the maximum <i>Specificity</i> .
MinValueSpSe	For given minimum values for <i>Specificity</i> and <i>Sensitivity</i> measures, MinValueSpSe identifies the optimal value as the one that gives the maximum <i>Sensitivity</i> or <i>Specificity</i> (user-defined).
MaxSp MaxSe	MaxSp and MaxSe are two measures based on the maximization of <i>Specificity</i> and <i>Sensitivity</i> , respectively.
MaxSpSe	MaxSpSe is a measure based on the simultaneous maximization of both <i>Specificity</i> and <i>Sensitivity</i> measures.
MaxProdSpSe	MaxProdSpSe is a measure based on the maximization of the product of <i>Sensitivity</i> and <i>Specificity</i> .
ROC01	ROC01 identifies the optimal cutpoint that is closest to the upper-left corner (0,1) of the ROC graph.
SpEqualSe	SpEqualSe is a measure based on the minimization of the absolute difference between <i>Sensitivity</i> and <i>Specificity</i> .
MaxEfficiency	MaxEfficiency is a measure based on the minimization of the misclassification error, $(FP + FN)/n$.

Continued on next page

Supplementary 1 – Continued from previous page

Method	Description
Minimax	Minimax is a measure based on the minimization of the most frequent error. Minimax is computed using the equation $\text{Minimax}_c = \min_c(\max(p(1 - \text{Sensitivity}) + (1 - p)(1 - \text{Specificity})))$ where c is the cutpoint and p is the prevalence.
MaxDOR	MaxDOR is a measure based on the maximization of the diagnostic odds ratio, calculated using the equation $\text{MaxDOR}_c = \max_c[(\text{Sensitivity} \times \text{Specificity}) / ((1 - \text{Sensitivity})(1 - \text{Specificity}))]$.
MinValueNPV MinValuePPV	For a given minimum value for <i>NPV</i> , MinValueNPV identifies the optimal value as the one that gives the maximum <i>PPV</i> . In contrast, for a given minimum value for <i>PPV</i> , MinValuePPV identifies the optimal value as the one that gives the maximum <i>NPV</i> .
ValueNPV ValuePPV	For a given particular value for <i>NPV</i> , ValueNPV identifies the optimal cutpoint as the one that gives the maximum <i>PPV</i> . In contrast, for a given particular value for <i>PPV</i> , ValuePPV identifies the optimal cutpoint as the one that gives the maximum <i>NPV</i> .
MinValueNPVPPV	For given minimum values for predictive values, MinValueNPVPPV identifies the optimal value as the one that gives the maximum <i>NPV</i> or <i>PPV</i> (user-defined).
PROC01	PROC01 identifies the optimal cutpoint that is closest to the upper-left corner (0,1) of the partial ROC (pROC) graph.
NPVEqualPPV	NPVEqualPPV is a measure based on the minimization of the absolute difference between <i>NPV</i> and <i>PPV</i> .
MaxNPVPPV	MaxNPVPPV is a measure based on the simultaneous maximization of both <i>NPV</i> and <i>PPV</i> measures.
MaxSumNPVPPV	MaxSumNPVPPV is a measure based on the maximization of the sum of <i>NPV</i> and <i>PPV</i> measures.
MaxProdNPVPPV	MaxProdNPVPPV is a measure based on the maximization of the product of <i>NPV</i> and <i>PPV</i> .
ValueDLR.Negative ValueDLR.Positive	These two measures are based on setting particular values for negative and positive diagnostic likelihood ratios, respectively.
MinPvalue	MinPvalue is a measure based on the minimization of the p -value of the Chi-square test on assessing the independence between the diagnostic and gold standard test.

Continued on next page

Supplementary 1 – Continued from previous page

Method	Description
ObservedPrev	ObservedPrev is a measure which identifies the optimal cutpoint closest to the observed prevalence by minimizing the quantity $ c - p $. This method is valid when the diagnostic test takes values within the interval $[0, 1]$.
MeanPrev	MeanPrev is a measure which identifies the optimal cutpoint closest to the average of the diagnostic test values. It is suggested to use this measure if the diagnostic test takes values between 0 and 1.
PrevalenceMatching	PrevalenceMatching is a measure based on the equality of actual and predicted prevalence. The cutpoint minimizes the absolute quantity $ p(1 - \text{Sensitivity}) - (1 - p)(1 - \text{Specificity}) $. This method is valid when the diagnostic test takes values within the interval $[0, 1]$.

For details, see Lopez-Raton et al. (2014).

Supplementary 2: Non-alcoholic fatty liver disease (NAFLD) data (Çelikkilek et al., 2014).

Grup	mir197	mir146b	mir181d	mir99a	Grup	mir197	mir146b	mir181d	mir99a
1	0.921	0.687	0.474	-0.941	0	1.214	1.122	0.882	1.610
1	0.967	1.059	0.474	0.575	0	1.401	0.148	0.444	0.625
1	0.854	1.105	0.722	0.936	0	0.494	-0.179	1.386	0.134
1	-1.088	-1.353	-0.577	-1.077	0	1.608	1.386	2.242	0.926
1	0.107	0.515	-0.286	0.560	0	1.274	1.609	0.769	1.108
1	0.547	1.191	0.583	1.119	0	0.827	1.128	0.452	0.374
1	-1.081	-1.445	-1.303	-1.202	0	-0.147	-0.545	0.878	0.044
1	-1.081	-1.308	-1.276	-1.066	0	0.353	0.320	-0.225	0.367
1	0.841	0.463	-0.290	0.747	0	-1.635	-0.677	-0.838	-0.543
1	-1.188	-0.975	-1.407	-2.123	0	1.848	1.523	1.712	0.940
1	-1.014	-0.649	-1.194	-1.786	0	0.987	0.606	0.626	0.542
1	-1.081	-1.256	-1.229	-0.679	0	0.020	0.503	0.600	0.367
1	-1.295	-1.204	-1.607	-2.216	0	1.061	1.518	1.217	0.209
1	-1.081	-1.268	-0.829	-0.658	0	0.474	0.572	0.292	0.786
1	-1.081	-1.365	-1.376	-1.457	0	-0.868	-0.505	-0.408	-0.117
1	-1.081	-1.371	-0.812	-1.804	0	-0.414	-0.259	0.665	0.363
1	-1.081	-0.769	-1.359	-0.156	0	0.394	0.417	1.000	0.130
1	0.854	1.243	0.444	1.460	0	0.941	0.543	0.431	1.083
1	-1.074	-1.365	-1.572	-0.339	0	-0.387	-0.202	-0.568	0.345
1	-0.634	-0.276	-0.130	-0.081	0	-0.674	-0.689	0.995	0.893