mathematics

MDPI

Article

Eigenvalue Distributions in Random Confusion Matrices: Applications to Machine Learning Evaluation

Oyebayo Ridwan Olaniran [1,*], Ali Rashash R. Alzahrani [2] and Mohammed R. Alzahrani [3]

[1] Department of Statistics, Faculty of Physical Sciences, University of Ilorin, Ilorin 1515, Nigeria
[2] Mathematics Department, Faculty of Sciences, Umm Al-Qura University, Makkah 24382, Saudi Arabia; arrzahrani@uqu.edu.sa
[3] Department of Psychology, Faculty of Education, Umm Al-Qura University, Al-Abidiyah, Makkah 24382, Saudi Arabia; mrzahrani@uqu.edu.sa
* Correspondence: olaniran.or@unilorin.edu.ng

Abstract: This paper examines the distribution of eigenvalues for a 2×2 random confusion matrix used in machine learning evaluation. We also analyze the distributions of the matrix's trace and the difference between the traces of random confusion matrices. Furthermore, we demonstrate how these distributions can be applied to calculate the superiority probability of machine learning models. By way of example, we use the superiority probability to compare the accuracy of four disease outcomes machine learning prediction tasks.

Keywords: eigenvalue; confusion matrix; random matrix; probability distribution; evaluation metrics

MSC: 60E05; 62H30

Citation: Olaniran, O.R.; Alzahrani, A.R.R.; Alzahrani, M.R. Eigenvalue Distributions in Random Confusion Matrices: Applications to Machine Learning Evaluation. *Mathematics* **2024**, *12*, 1425. https://doi.org/10.3390/math12101425

Academic Editor: Yang Chen

Received: 13 April 2024
Revised: 1 May 2024
Accepted: 3 May 2024
Published: 7 May 2024

1. Introduction

The distribution of eigenvalues of a confusion matrix is an interesting and important concept in machine learning (ML), particularly in the evaluation of classification models [1]. Confusion matrices are widely used to assess the performance of a classification algorithm by providing a detailed breakdown of the predicted and actual class labels [2–4]. The eigenvalues of a confusion matrix offer insights into the underlying structure and characteristics of the classification results [5–9]. Eigenvalues are a mathematical concept used to analyze linear transformations, and in the context of confusion matrices, they can reveal information about the matrix's behaviour [10,11]. The distribution of eigenvalues provides a quantitative measure of the spread and concentration of information in the matrix. Understanding the distribution of eigenvalues of a confusion matrix can be valuable for various purposes, including model assessment, variable selection, high-dimensional analysis, dimension reduction, model comparison, anomaly detection, and generalization or overfitting issues [1,11–15].

For example, in Ref. [1], the significance of eigenvalue analysis for selecting important features in big data was explored. The authors emphasize the importance of understanding the patterns of eigenvalues in covariance matrices for various analytical purposes, such as model comparison and anomaly detection. They highlight how eigenvalues provide insights into the underlying structure of classification results, contributing to an overall understanding of model performance. In a similar study by [11], the authors utilized eigenvalue analysis in conjunction with principal component analysis (PCA) methods to reduce the dimensionality of big data before exploring the performances of several classification methods. The results of their analysis revealed that the outcomes from eigenvalue and PCA are much superior to those from the linear discriminant analysis (LDA) procedure.

In another study by [15], various eigenvalue-based dimension reduction techniques were compared for high-dimensional analysis. Specifically, the authors investigated the performances of PCA, LDA, and singular value decomposition (SVD). The findings from the study validate the utility of eigenvalue-based dimension reduction techniques in handling high-dimensional data. By comparing the effectiveness of PCA, LDA, and SVD, the research underscores the importance of eigenvalue analysis in addressing the challenges posed by high-dimensional datasets. Moreover, ref. [16] utilized eigenvalue analysis to tackle the generalization error problem in two-layered neural networks for high-dimensional analysis. By leveraging eigenvalue properties, the study aimed to enhance the understanding of how neural networks generalize from training data to unseen data. Eigenvalue analysis in this context provides valuable insights into the behaviour and performance of neural networks, particularly in high-dimensional spaces. The approach in [16] highlights the significance of incorporating eigenvalue-based techniques in optimizing and refining machine learning models for complex data analysis tasks.

In a different context within high-dimensional analysis, Sifaou et al. [14] employed eigenvalue and eigenvector analyses to improve the performance of a high-dimensional LDA classifier in the spiked covariance model. The author introduced a modified regularized R-LDA that is based on eigenvalue and eigenvector analyses. Numerical simulations, using both real and simulated data, revealed that the proposed classifier yields better classification performance than the classical R-LDA while requiring lower computational complexity. In a similar context, ref. [13] increased the performance of support vector machine (SVM) by employing eigenvalue analysis of the features covariance matrices and subsequently performing PCA to reduce the dimension of the features. This approach helps to increase the prediction accuracy of hepatitis disease.

In a Bayesian analysis of confusion matrices, Caelen [17] delved into Bayesian methods for analyzing confusion matrices in machine learning. While Bayesian approaches are widely used in various aspects of ML, their application to confusion matrices provides a unique perspective on model evaluation. Ref. [17] provided Bayesian interpretations of various evaluation metrics derived from confusion matrices of machine learning models. The authors presented posterior distributions for these metrics from the confusion matrix and used them to compare the performances of several ML models.

The findings of various studies reviewed indicate a significant body of work on eigenvalue analysis within the context of dimension reduction, particularly in Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). However, there is a notable gap in research concerning the eigenvalue analysis of confusion matrices arising from machine learning (ML) models. In high-dimensional analysis and variable selection, dimension reduction serves as a filtering mechanism wherein techniques like eigenvalue analysis are employed to select important variables before training a classification model. Many authors, including [2,18–25], among others, have criticized this approach. They argue that it eliminates the possibility of interaction effects present in variables. Therefore, embedded and wrapper variable selection methods, which combine selection techniques and ML models, are preferred. In this regard, comparing ML models based on confusion matrices from a trained model will be more beneficial than the covariance matrix of a pre-trained model.

Moreover, by leveraging eigenvalue analysis, researchers can objectively compare different machine learning models, discerning their relative strengths and weaknesses based on the underlying structure of their confusion matrices [26–29]. Hence, this paper presents the distribution of eigenvalues for a 2×2 random confusion matrix arising from a machine learning evaluation scenario. Furthermore, we provide distributions for both the matrix's trace and the difference between the traces of two random confusion matrices. We also demonstrate how these distributions can be utilized to compute the superiority probability of ML models.

2. Distribution of Eigenvalues of Random Confusion Matrix

Suppose we have a learning problem given by data $\mathcal{D} = \{X_i, Y_i\}$, where $i \in 1, 2, \ldots, n$, X_i is the matrix of the features, and Y_i is the response vector which we assume to be categorical with k classes. For simplicity, we consider the binary case with $k = 2$ as the derivation in this paper can be easily generalized to the multiclass k classes. In any binary classification problem, the goal is to predict the Y_i based on new information $x \in \mathcal{X}$ using a classifier $\hat{y} : f(x)$. Consider a testing dataset denoted as $T = \{(X_i, Y_i)\}_{i=1}^{n_T}$, comprising n_T independent samples drawn from an unknown distribution $F(X, Y)$. To assess the accuracy of predictions made by $\hat{y}$ on the samples in T, we introduce a loss function $\mathcal{L} : y \times \hat{y} \to \{a, b, c, d\}$. Let y belong to $\{\theta_0, \theta_1\}$ as the true class, and $\hat{y}$ belong to $\{\theta_0, \theta_1\}$ as the predicted class. Following convention in [17], we define the mapping of the $\mathcal{L}$ function as follows:

$$\mathcal{L} = \begin{cases} a, & \text{if } y = \theta_1 \text{ and } \hat{y} = \theta_1 \\ b, & \text{if } y = \theta_1 \text{ and } \hat{y} = \theta_0 \\ c, & \text{if } y = \theta_0 \text{ and } \hat{y} = \theta_1 \\ d, & \text{if } y = \theta_0 \text{ and } \hat{y} = \theta_0 \end{cases} \tag{1}$$

where a denotes true positive, b denotes false negative, c denotes false positive and d denotes true negative. The elements of vector $\mathcal{L}$ can be presented in a 2×2 matrix often referred to as a confusion matrix. Let $\boldsymbol{A}$ represents the 2×2 confusion matrix obtained from a classification learning problem defined above; $\boldsymbol{A}$ can be defined as

$$\boldsymbol{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{2}$$

The obvious properties of $\boldsymbol{A}$ are that (a.) it is not symmetric (b.) it is square and it is also random. Now, if we assume $\boldsymbol{A}$ is diagonalizable such that there exist a scalar λ and vector $\boldsymbol{\mathcal{V}}$ that we can use to decompose $\boldsymbol{A}$ using

$$\boldsymbol{A}\boldsymbol{\mathcal{V}} = \lambda\boldsymbol{\mathcal{V}} \tag{3}$$

then $\lambda = \{\lambda_1, \lambda_2\}$ and $\boldsymbol{\mathcal{V}} = \begin{bmatrix} v_{11} & v_{12} \\ v_{13} & v_{13} \end{bmatrix}$ are the eigenvalues and eigenvectors of $\boldsymbol{A}$ respectively. One interesting property of eigenvalues of this type of diagonalizable square matrix is that the sum of the eigenvalues equals the trace of the matrix. That is

$$tr(\boldsymbol{A}) = \sum_{j=1}^{2} \lambda_j. \tag{4}$$

The $tr(\boldsymbol{A})$ is very useful in evaluating the accuracy of a classifier in a machine learning problem most especially if the categories of the response variable is balance that is $p_k = 1/k$. In a balance binary classification problem with n_T test cases, the accuracy $(\hat{\phi})$ of a classifier can be computed using

$$\begin{aligned} \hat{\phi} &= n_T^{-1} tr(\boldsymbol{A}) \\ &= n_T^{-1} \sum_{j=1}^{2} \lambda_j. \end{aligned} \tag{5}$$

Note that, since elements of $\boldsymbol{A}$ are resultants of random outcomes of randomly generated test instances used to validate classifier $\hat{y}$, $\boldsymbol{A}$ can be regarded as a random matrix. Also, since only n_T is the only known parameter, the elements of $\boldsymbol{A}$ can be assumed to be multinomially distributed with parameters n_T, π_a, π_b, π_c and π_d. Thus, the joint density function of the elements in the random matrix $\boldsymbol{A}$ can be given as

$$P(a, b, c, d | n_T, \pi_a, \pi_b, \pi_c, \pi_d) = \frac{n_T!}{a! \cdot b! \cdot c! \cdot d!} a^{\pi_a} \cdot b^{\pi_b} \cdot c^{\pi_c} \cdot d^{\pi_d} \cdot I(a + b + c + d = n_T). \tag{6}$$

The last part of the RHS of (6) implies that it is required that the total sum of the four cells be equal to the number of test instances for it to be a proper pdf.

Theorem 1. *The joint probability density function (pdf) of the eigenvalues* (λ_1, λ_2) *of a* 2×2 *confusion matrix is given by:*

$$f(\lambda_1, \lambda_2) = \frac{1}{4s^2\sqrt{\pi}} e^{-\frac{1}{2s^2}(\lambda_1^2+\lambda_2^2-2\bar{A}(\lambda_1+\lambda_2)+2\bar{A}^2)} |\lambda_1 - \lambda_2| \quad -\infty \leq \lambda_1, \lambda_2 \leq \infty. \tag{7}$$

Proof. We begin this proof by standardizing the element of the confusion matrix $\boldsymbol{A}$ as follows:

$$\boldsymbol{z} = \boldsymbol{s}^{-1}(\boldsymbol{A} - \bar{\boldsymbol{A}}), \tag{8}$$

where $\bar{\boldsymbol{A}}$ is the mean of all elements in $\boldsymbol{A}$ and $\boldsymbol{s}$ is the standard deviation of each element from their mean. If the confusion matrix is balanced such that $p_k = 1/4$ for all four elements, the mean $\bar{\boldsymbol{A}}$ and standard deviation $\boldsymbol{s}$ are $\frac{n}{4}$ and $\sqrt{\frac{3n}{16}}$, respectively. If otherwise, the mean $\bar{\boldsymbol{A}}$ and standard deviation $\boldsymbol{s}$ are computed as follows:

$$\begin{aligned} \bar{\boldsymbol{A}} &= \frac{a+b+c+d}{4}, \\ \boldsymbol{s} &= \sqrt{\frac{(a-\bar{\boldsymbol{A}})^2 + (b-\bar{\boldsymbol{A}})^2 + (c-\bar{\boldsymbol{A}})^2 + (d-\bar{\boldsymbol{A}})^2}{3}}. \end{aligned} \tag{9}$$

The next step involves the symmetrization of $\boldsymbol{z}$ to achieve symmetry as expected for a Gaussian Orthogonal Ensemble (GOE) [30,31].

$$\boldsymbol{z}_s = (\boldsymbol{z} + \boldsymbol{z}^T)/2. \tag{10}$$

where $\boldsymbol{z}_s$ is the standardized symmetrized confusion matrix, $\boldsymbol{z}$ is the standardized confusion matrix and $\boldsymbol{z}^T$ is it transpose. The elements of $\boldsymbol{z}_s$ are explicitly defined as

$$\boldsymbol{z}_s = \begin{bmatrix} a' & b' \\ b' & d' \end{bmatrix}. \tag{11}$$

Now that we have established that $\boldsymbol{z}_s$ is a GOE with joint pdf of (a', b', d') given by

$$f(a', b', d') = \frac{1}{2\pi} e^{-\frac{tr(z_s^2)}{2}}; \quad -\infty \leq a', b', d' \leq \infty, \tag{12}$$

we can proceed to derive the distribution of eigenvalues of $\boldsymbol{z}_s$ and subsequently the distribution of eigenvalues of $\boldsymbol{A}$. Note that by using the change of variable rule, the distribution of eigenvalues (η_1, η_2) of $\boldsymbol{z}_s$ is given by

$$f(\eta_1, \eta_2) = f(a', b', d')|det(J)|, \tag{13}$$

where J is the change of variable Jacobian matrix. Thus, since $\boldsymbol{z}_s$ matrix is invariant under orthogonal transformation such that

$$\boldsymbol{z}_s = \boldsymbol{P}^T \boldsymbol{z}_s^{\eta} \boldsymbol{P} \tag{14}$$

where $\boldsymbol{P} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}$ is an orthogonal matrix and $z_s^{\eta} = \begin{bmatrix} \eta_1 & 0 \\ 0 & \eta_2 \end{bmatrix}$ is a diagonal matrix of the eigenvalues of matrix z_s, we have

$$\begin{aligned} \begin{bmatrix} a' & b' \\ b' & d' \end{bmatrix} &= \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \eta_1 & 0 \\ 0 & \eta_2 \end{bmatrix} \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \\ &= \begin{bmatrix} \eta_1 \cos^2(\theta) + \eta_2 \sin^2(\theta) & (\eta_1 - \eta_2)\sin(\theta) \\ (\eta_1 - \eta_2)\sin(\theta) & \eta_1 \sin^2(\theta) + \eta_2 \cos^2(\theta) \end{bmatrix}. \end{aligned} \tag{15}$$

As we are moving from z_s to z_s^{η}, it is required to normalized the resultant pdf of (η_1, η_2) using the Jacobian determinant $\det(J)$. The change of variable Jacobian J is given as

$$\begin{aligned} J &= \begin{bmatrix} \frac{\partial a'}{\partial \eta_1} & \frac{\partial a'}{\partial \eta_2} & \frac{\partial a'}{\partial \theta} \\ \frac{\partial d'}{\partial \eta_1} & \frac{\partial d'}{\partial \eta_2} & \frac{\partial d'}{\partial \theta} \\ \frac{\partial b'}{\partial \eta_1} & \frac{\partial b'}{\partial \eta_2} & \frac{\partial b'}{\partial \theta} \end{bmatrix} \\ &= \begin{bmatrix} \cos^2(\theta) & \sin^2(\theta) & (\eta_2 - \eta_1)\sin(2\theta) \\ \sin^2(\theta) & \cos^2(\theta) & (\eta_1 - \eta_2)\sin(2\theta) \\ \frac{1}{2}\sin(2\theta) & -\frac{1}{2}\sin(2\theta) & (\eta_1 - \eta_2)\cos(2\theta) \end{bmatrix}. \end{aligned} \tag{16}$$

Subsequently, the determinant of the Jacobian is given by

$$\begin{aligned} \det(J) &= \det \begin{bmatrix} \cos^2(\theta) & \sin^2(\theta) & (\eta_2 - \eta_1)\sin(2\theta) \\ \sin^2(\theta) & \cos^2(\theta) & (\eta_1 - \eta_2)\sin(2\theta) \\ \frac{1}{2}\sin(2\theta) & -\frac{1}{2}\sin(2\theta) & (\eta_1 - \eta_2)\cos(2\theta) \end{bmatrix} \\ &= \cos^2(\theta) \begin{bmatrix} \cos^2(\theta) & (\eta_1 - \eta_2)\sin(2\theta) \\ -\frac{1}{2}\sin(2\theta) & (\eta_1 - \eta_2)\cos(2\theta) \end{bmatrix} \\ &\quad - \sin^2(\theta) \begin{bmatrix} \sin^2(\theta) & (\eta_1 - \eta_2)\sin(2\theta) \\ \frac{1}{2}\sin(2\theta) & (\eta_1 - \eta_2)\cos(2\theta) \end{bmatrix} \\ &\quad + (\eta_2 - \eta_1)\sin(2\theta) \begin{bmatrix} \sin^2(\theta) & \cos^2(\theta) \\ \frac{1}{2}\sin(2\theta) & -\frac{1}{2}\sin(2\theta) \end{bmatrix} \\ &= (\eta_1 - \eta_2)(\cos^2(2\theta) + \sin^2(2\theta)) \\ &= \eta_1 - \eta_2. \end{aligned} \tag{17}$$

Therefore, the corresponding joint pdf of (η_1, η_2) for matrix z_s is given by

$$f(\eta_1, \eta_2) = \frac{1}{4\sqrt{\pi}} e^{-\frac{1}{2}(\eta_1^2 + \eta_2^2)} |\eta_1 - \eta_2|; \quad -\infty \leq \eta_1, \eta_2 \leq \infty. \tag{18}$$

Now that we have the distribution of the eigenvalues for the transformed matrix z_s, we can obtain the distribution of eigenvalues for the required confusion matrix $\boldsymbol{A}$ as follows

$$\boldsymbol{A} = \boldsymbol{s} z_s + \bar{\boldsymbol{A}}. \tag{19}$$

From (19), it can be seen that there is a one-one correspondence between matrices $\boldsymbol{A}$ and z_s, thus, we can define the eigenvalues of $\boldsymbol{A}$ as a function of eigenvalues of z_s. This implies

$$\boldsymbol{\lambda} = \boldsymbol{s}\boldsymbol{\eta} + \bar{\boldsymbol{A}}. \tag{20}$$

where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$ and $\boldsymbol{\eta} = (\eta_1, \eta_2)$. Therefore, the joint pdf of eigenvalues of $\boldsymbol{A}$ is given by

$$\begin{aligned} f(\lambda_1, \lambda_2) &= f_{\eta_1,\eta_2}(\lambda_1, \lambda_2)\left|\frac{d\boldsymbol{\eta}}{d\boldsymbol{\lambda}}\right| \\ &= \frac{1}{4\sqrt{\pi}} e^{-\frac{1}{2s^2}(\lambda_1^2+\lambda_2^2-2\bar{A}(\lambda_1+\lambda_2)+2\bar{A}^2)}\left|\frac{\lambda_1-\lambda_2}{s}\right|\left|\frac{1}{s}\right| \\ f(\lambda_1, \lambda_2) &= \frac{1}{4s^2\sqrt{\pi}} e^{-\frac{1}{2s^2}(\lambda_1^2+\lambda_2^2-2\bar{A}(\lambda_1+\lambda_2)+2\bar{A}^2)}|\lambda_1-\lambda_2|; \quad -\infty \le \lambda_1, \lambda_2 \le \infty, \end{aligned} \tag{21}$$

where $\eta_1^2 + \eta_2^2 = \left(\frac{\lambda_1-\bar{A}}{s}\right)^2 + \left(\frac{\lambda_2-\bar{A}}{s}\right)^2$, $\eta_1 - \eta_2 = \left(\frac{\lambda_1-\bar{A}}{s}\right) - \left(\frac{\lambda_2-\bar{A}}{s}\right)$ and $\frac{d\boldsymbol{\eta}}{d\boldsymbol{\lambda}} = \frac{1}{s}$. □

Remark 1. *Equation (21) implies $f(\lambda_1, \lambda_2)$ is a shifted GOE with mean and variance $\bar{\boldsymbol{A}}$ and s^2 respectively.*

2.1. Distribution of Trace of a Random Confusion Matrix

Theorem 2. *The probability density function (pdf) of the trace $t = tr(\boldsymbol{A})$ of a 2×2 random confusion matrix $\boldsymbol{A}$ is given by:*

$$f(t) = \frac{1}{\sqrt{4\pi s^2}} e^{-\frac{1}{4s^2}(t-2\bar{A})^2}; \quad -\infty \le t \le \infty, \tag{22}$$

Lemma 1. *Suppose matrix $\boldsymbol{z}_s$ is a GOE, thus its elements (a', d') are independent and identically distributed as normal, $N(0, 1)$, and b' is distributed normally as $N(0, 1/2)$.*

Remark 2. *Lemma 1 implies that the distribution of the trace of the standardized symmetrized matrix $\boldsymbol{z}_s$ is the sum of two normal distributions denoted by $N(0, 2)$. Thus,*

$$f(w) = \frac{1}{2\sqrt{\pi}} e^{-w^2/4}; \quad -\infty \le w \le \infty. \tag{23}$$

Proof. Again, considering the standardized symmetrized confusion matrix $\boldsymbol{z}_s$ defined in (11). The eigenvalues (η_1, η_2) of $\boldsymbol{z}_s$ can be estimated from the characteristics equation

$$\eta^2 - (a' + d')\eta + (a'd' - b'^2) = 0. \tag{24}$$

Solving (24) gives

$$\begin{aligned} \eta_1 &= \frac{(a'+d') + \sqrt{(a'+d')^2 - 4b'^2}}{2}, \\ \eta_2 &= \frac{(a'+d') - \sqrt{(a'+d')^2 - 4b'^2}}{2}. \end{aligned} \tag{25}$$

Recall that the trace (w) for matrix $\boldsymbol{z}_s$ is given by

$$\begin{aligned} tr(\boldsymbol{z}_s) &= \eta_1 + \eta_2 \\ w &= \eta_1 + \eta_2 \\ &= \frac{(a'+d') + \sqrt{(a'+d')^2 - 4b'^2}}{2} + \frac{(a'+d') - \sqrt{(a'+d')^2 - 4b'^2}}{2} \\ w &= a' + d' \end{aligned} \tag{26}$$

Again, by change of variable, we can derive the distribution of the trace of matrix $\boldsymbol{A}$ as follows

$$\begin{aligned} f(t) &= f_w(t)\left|\frac{dw}{dt}\right| \\ &= \frac{1}{2\sqrt{\pi}}e^{-(t-2\boldsymbol{A})^2\bar{/}4s^2}\left|\frac{dw}{dt}\right| \\ &= \frac{1}{2\sqrt{\pi}}e^{-(t-2\bar{\boldsymbol{A}})^2/4s^2}\left|\frac{1}{s}\right| \\ f(t) &= \frac{1}{\sqrt{4\pi s^2}}e^{-(t-2\bar{\boldsymbol{A}})^2/4s^2}; \quad -\infty \leq t \leq \infty \end{aligned} \tag{27}$$

□

Remark 3. *Equation (27) implies $f(t)$ is normally distributed with mean and variance $2\bar{\boldsymbol{A}}$ and $2s^2$ respectively and it is denoted by $N(2\bar{\boldsymbol{A}}, 2s^2)$.*

Lemma 2. *The cumulative distribution function $F(t)$ for the trace of matrix $\boldsymbol{A}$ is given by*

$$\begin{aligned} F(t) &= \int_{-\infty}^{t} f(t)dt \\ &= \int_{-\infty}^{t} \frac{1}{\sqrt{4\pi s^2}}e^{-(t-2\bar{\boldsymbol{A}})^2/4s^2}dt \\ F(t) &= \Phi\left(\frac{t-2\bar{\boldsymbol{A}}}{\sqrt{2s^2}}\right), \end{aligned} \tag{28}$$

where Φ is the cdf of standardized normal distribution with mean 0 and variance 1.

Figure 1 illustrates the graph of the probability density function for the trace of a 2×2 random matrix, showcasing various diagonal probabilities π_1 and π_4. The plot highlights that the distribution closely resembles a normal distribution when the diagonal cell probabilities are equal or nearly equal. However, it is noticeably peaked when the confusion matrix stems from a highly unbalanced machine learning task. Figure 2 supports the observations made in Figure 1, displaying a consistently increasing cumulative distribution function when cell probabilities are approximately equal, contrasted with a vertical line around 1 in cases of unbalanced data.

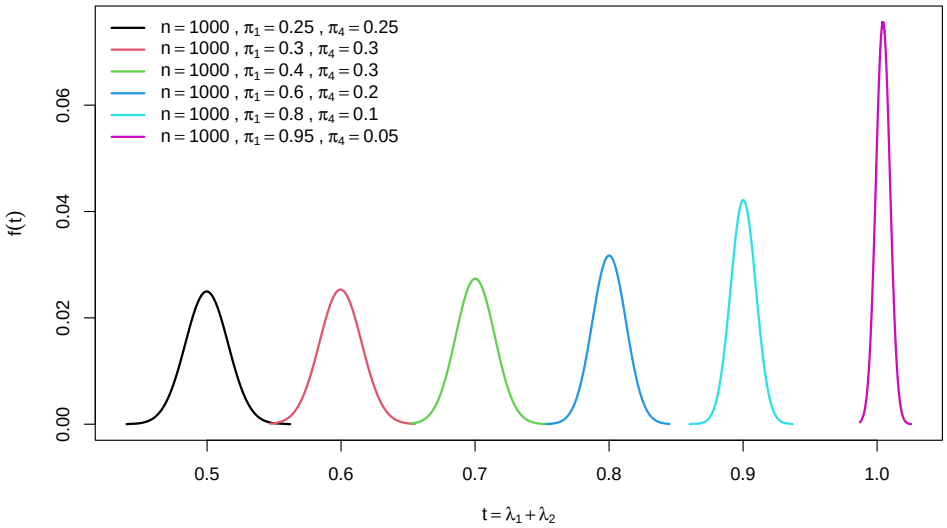

Figure 1. Graphs of the pdf of the trace of a random 2×2 confusion matrix for different diagonal probabilities π_1 and π_4.

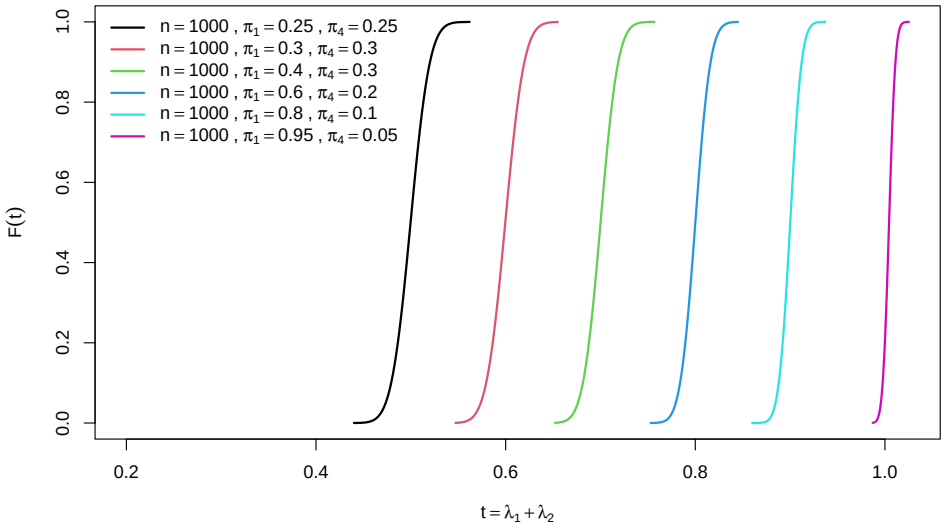

Figure 2. Graphs of the cdf of the trace of a random 2×2 confusion matrix for different diagonal probabilities π_1 and π_4.

2.2. Distribution of Difference of Two Traces of Random Confusion Matrices

In machine learning, it is often valuable to compare the confusion matrices of two algorithms, such as decision trees and random forests [2,18,19]. Understanding the distribution of differences is crucial because it quantifies the degree of superiority one algorithm holds over the other. Therefore, in this section, we have developed the distribution of differences between two sets of 2×2 random confusion matrices.

Theorem 3. *The probability density function (pdf) of the difference of two traces of 2×2 random confusion matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ denoted by $m = tr(\boldsymbol{A}) - tr(\boldsymbol{B})$ is given by*

$$f(m) = \frac{1}{\sqrt{4\pi S^2_{\boldsymbol{A}+\boldsymbol{B}}}} e^{-\frac{1}{4S^2_{\boldsymbol{A}+\boldsymbol{B}}}(m-2\bar{\boldsymbol{A}}+2\bar{\boldsymbol{B}})^2}; \quad -\infty \leq m \leq \infty, \tag{29}$$

where $S^2_{\boldsymbol{A}+\boldsymbol{B}} = S^2_{\boldsymbol{A}} + S^2_{\boldsymbol{B}}$.

Lemma 3. *Suppose the traces of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are independently distributed normal $N(2\bar{\boldsymbol{A}}, 2s^2)$, then the distribution of their difference is also normal with mean $2\bar{\boldsymbol{A}} - 2\bar{\boldsymbol{B}}$ and variance $S^2_{\boldsymbol{A}+\boldsymbol{B}} = S^2_{\boldsymbol{A}} + S^2_{\boldsymbol{B}}$.*

Remark 4. *Lemma (3) implies that the pdf of the difference of two traces of 2×2 random confusion matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is $N(2\bar{\boldsymbol{A}} - 2\bar{\boldsymbol{B}}, S^2_{\boldsymbol{A}+\boldsymbol{B}})$.*

Proof. This proof follows from the earlier distribution of t which follows $N(2\bar{\boldsymbol{A}}, 2s^2)$. Thus, the pdf of the difference of two traces of 2×2 random confusion matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ denoted by $m = tr(\boldsymbol{A}) - tr(\boldsymbol{B})$ is given by

$$f(m) = \frac{1}{\sqrt{4\pi S^2_{\boldsymbol{A}+\boldsymbol{B}}}} e^{-\frac{1}{4S^2_{\boldsymbol{A}+\boldsymbol{B}}}(m-2\bar{\boldsymbol{A}}+2\bar{\boldsymbol{B}})^2}; \quad -\infty \leq m \leq \infty, \tag{30}$$

□

Lemma 4. *The cumulative distribution function $F(m)$ for the difference of two traces of 2×2 random confusion matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ denoted by $m = tr(\boldsymbol{A}) - tr(\boldsymbol{B})$ is given by*

$$
\begin{aligned}
F(m) &= \int_{-\infty}^{m} f(m)dm \\
&= \int_{-\infty}^{m} \frac{1}{\sqrt{4\pi S^2_{\bar{A}+\bar{B}}}} e^{-\frac{1}{4S^2_{\bar{A}+\bar{B}}}(m-2\bar{A}+2\bar{B})^2} dm \\
F(m) &= \Phi\left(\frac{m-2\bar{A}+2\bar{B}}{\sqrt{S^2_{\bar{A}+\bar{B}}}}\right),
\end{aligned} \tag{31}
$$

where Φ *is the* cdf *of standardized normal distribution with mean* 0 *and variance* 1.

Figure 3 displays the distributions of the differences between 2×2 random confusion matrices at various effect sizes. The plot illustrates that as the effect size increases, the spread of the distribution decreases, and conversely, as the effect size decreases, the spread increases. Similarly, the cumulative distribution function in Figure 4 supports these findings.

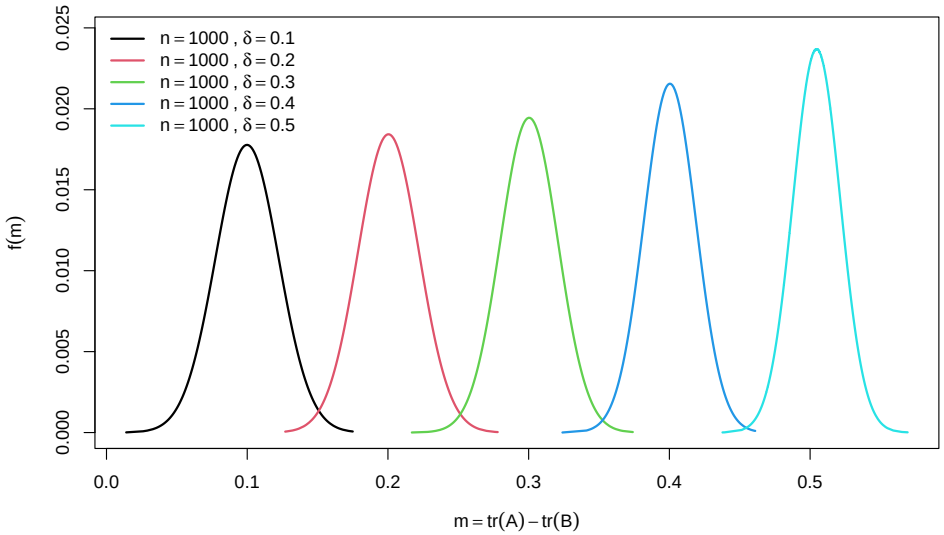

Figure 3. Graphs of the pdf of for the difference of two traces of 2×2 random confusion matrices for different effect size $\delta = tr(\boldsymbol{A}) - tr(\boldsymbol{B})$.

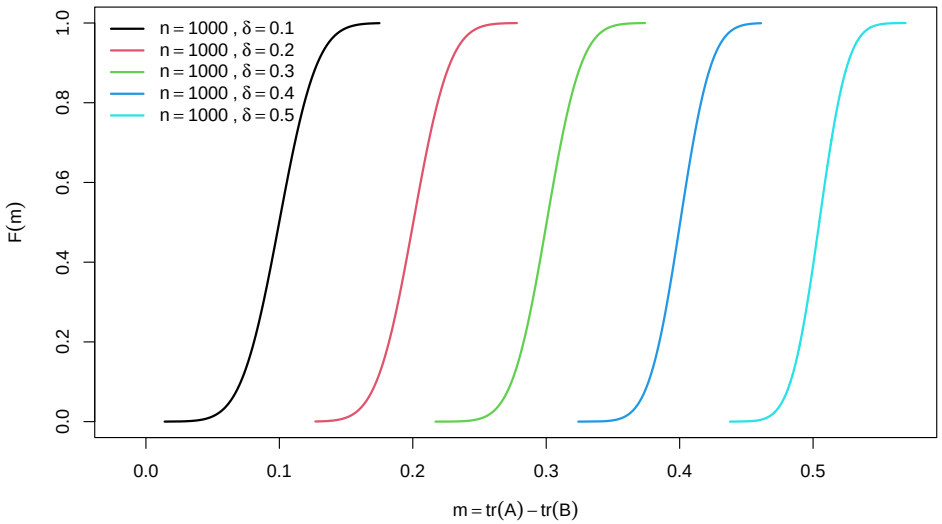

Figure 4. Graphs of the cdf of for the difference of two traces of 2×2 random confusion matrices for different effect size $\delta = tr(\boldsymbol{A}) - tr(\boldsymbol{B})$

3. Example

To demonstrate our approach, let us examine an example featuring two classifiers, $\boldsymbol{A}$ and $\boldsymbol{B}$, generating the following confusion matrices on the identical testing dataset T, where the size of T is $n_T = 200$:

$$\begin{aligned} \boldsymbol{A} &= \begin{bmatrix} 62 & 36 \\ 51 & 51 \end{bmatrix} \\ \boldsymbol{B} &= \begin{bmatrix} 50 & 53 \\ 50 & 47 \end{bmatrix}. \end{aligned} \tag{32}$$

The eigenvalues of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are denoted by $(\lambda_1^A, \lambda_2^A)$ and $(\lambda_1^B, \lambda_2^B)$ respectively. Correspondingly, the traces of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ can be computed as follows:

$$\begin{aligned} tr(\boldsymbol{A}) &= \lambda_1^A + \lambda_2^A \\ tr(\boldsymbol{B}) &= \lambda_1^B + \lambda_2^B. \end{aligned} \tag{33}$$

The estimates for the eigenvalues and traces of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are as follows: $(\lambda_1^A = 99.7, \lambda_2^A = 13.3, \mathrm{tr}(\boldsymbol{A}) = 113)$ and $(\lambda_1^B = 100, \lambda_2^B = -3, \mathrm{tr}(\boldsymbol{B}) = 97)$, respectively. With these trace values, we can compute the accuracies of the two classifiers: $(\phi^A = 0.57, \phi^B = 0.49)$. According to this criterion, it seems that classifier A outperforms B. However, without sufficient information, we cannot conclusively determine whether this superiority is genuine or merely a result of chance. By analyzing the distribution of the difference between the two traces, as shown in (30) and (31), we can quantify the extent to which classifier A is superior to classifier B. Therefore, the probability that classifier A genuinely outperforms B is given by

$$\begin{aligned} P[(\phi^A = 0.57 - \phi^B = 0.49) > 0] &= 1 - P[(\phi^A = 0.57 - \phi^B = 0.49) < 0] \\ &= 1 - F(m) \\ &= 1 - \Phi\left(\frac{0.08}{\sqrt{0.0775}}\right) \\ &= 0.8492. \end{aligned} \tag{34}$$

This estimated probability value suggests a strong likelihood that model A significantly surpasses model B in terms of accuracy performance.

4. Applications

We utilize the following datasets to demonstrate the practical application of analyzing the distribution of differences between two traces of random confusion matrices in machine learning, particularly within the field of medicine and health:

1. Heart disease [32]: This dataset comprises information from 303 patients with heart disease at Cleveland Hospital, including 14 features. The objective is to determine the presence or absence of heart disease.
2. Breast cancer [33]: Originating from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia, this dataset contains data from 286 patients with breast cancer, encompassing 9 features. The goal is to predict the presence or absence of breast cancer recurrence.
3. Liver disease [34]: This dataset consists of 584 patient records from the NorthEast region of Andhra Pradesh, India, across 10 features. The objective is to predict whether a patient has liver disease using various biochemical markers.

The aim of this section is to implement and compare four baseline machine learning algorithms applied to these datasets: logistic regression (LR), decision trees (DT), random forest classification (RF) and XGboost classification (XG) [35]. The evaluation criterion utilized to compare the ML algorithms is accuracy. In addition, the supremacy of each of the algorithms is computed by computing the probability distribution in (31). Note that

this probability can empirically be computed by bootstrapping the dataset L times and then obtaining the empirical distribution of the difference between two accuracies or traces. Thus, the approximate bootstrap estimate of (31) is

$$\hat{F}(m) = L^{-1} \sum_{l=1}^{L} (m_l < m), \tag{35}$$

where $L = 5000$ is set as the bootstrap sample size. The significance of the approach presented in this study lies in its provision of a closed-form solution for this distribution. This solution offers a faster and more accurate method for calculating the distribution of differences between two accuracies. All analyses were carried out using R statistical software version 4.3.1.

Table 1 presents bootstrap accuracy estimates, denoted as $\hat{\phi}_L$, along with their standard errors, $SE(\hat{\phi}_L)$, and accuracy estimates for eigenvalue distribution, denoted as $\hat{\phi}_\lambda$, along with their standard errors, $SE(\hat{\phi}_\lambda)$, for the three datasets using the four baseline ML methods. The results indicate that the accuracy estimates and associated standard errors using both the bootstrap and eigenvalue distribution approaches are similar across the machine learning (ML) methods and datasets. This finding empirically validates the eigenvalue distribution approach for estimating the accuracy of an ML method based on the eigenvalue of a confusion matrix.

Table 1. The bootstrap accuracy estimate $\hat{\phi}_L$ along with its standard error $SE(\hat{\phi}_L)$, and the accuracy estimate for eigenvalue distribution $\hat{\phi}_\lambda$ along with its standard error $SE(\hat{\phi}_\lambda)$, for the three datasets using the four baseline methods.

	Heart Disease		**Breast Cancer**		**Liver Disease**	
Method	$\hat{\phi}_L$ $(SE(\hat{\phi}_L))$	$\hat{\phi}_\lambda$ $(SE(\hat{\phi}_\lambda))$	$\hat{\phi}_L$ $(SE(\hat{\phi}_L))$	$\hat{\phi}_\lambda$ $(SE(\hat{\phi}_\lambda))$	$\hat{\phi}_L$ $(SE(\hat{\phi}_L))$	$\hat{\phi}_\lambda$ $(SE(\hat{\phi}_\lambda))$
LR	0.83 (0.031)	0.88 (0.032)	0.70 (0.043)	0.72 (0.031)	0.71 (0.029)	0.72 (0.029)
DT	0.77 (0.042)	0.76 (0.025)	0.71 (0.031)	0.73 (0.024)	0.72 (0.031)	0.67 (0.021)
RF	0.82 (0.031)	0.84 (0.029)	0.82 (0.025)	0.79 (0.030)	0.82 (0.025)	0.80 (0.030)
XG	0.77 (0.036)	0.82 (0.029)	0.74 (0.030)	0.70 (0.031)	0.75 (0.030)	0.72 (0.030)

Table 2 presents pairwise comparison results of the accuracies of the four ML methods using both bootstrap and eigenvalue distribution approaches. Again, the estimates of the pairwise differences are similar in most cases in terms of direction (positive or negative). However, significant differences exist in the estimates of the superiority probability between the bootstrap and eigenvalue distribution approaches. On average, the results are approximately similar for positive differences but exhibit distinct differences for negative differences. The bootstrap tends to be conservative on average when the difference between the accuracies of two ML methods is negative but restrictive when the difference is positive. It is worth noting that bootstrap estimates are approximations to the distribution of the difference of ML accuracy, while the eigenvalue distribution provides the actual distribution of the difference based on Theorem 3. Thus, the results of the superiority probability obtained using the eigenvalue distribution are more reliable than bootstrap estimates, which have been reported in previous studies to have potentially biased estimates [36,37].

In terms of ML performance based on superiority probability, XG is on average better than LR and DT, while RF is on average better than XG. Thus, RF emerges as the best among the four ML methods across the three datasets in terms of the prediction accuracy and superiority of accuracy across several replications of the experiment.

Table 2. Estimates of the difference between pairwise accuracies ($\hat{m} = \phi_A - \phi_B$) and their respective superiority probabilities ($1 - F(\hat{m})$) using both bootstrap and eigenvalue distribution approaches across the three datasets.

	Heart Disease		**Breast Cancer**		**Liver Disease**	
Pair	$\hat{m}_L$ $(1 - \hat{F}(\hat{m}_L))$	$\hat{m}_\lambda$ $(1 - F(\hat{m}_\lambda))$	$\hat{m}_L$ $(1 - \hat{F}(\hat{m}_L))$	$\hat{m}_\lambda$ $(1 - F(\hat{m}_\lambda))$	$\hat{m}_L$ $(1 - \hat{F}(\hat{m}_L))$	$\hat{m}_\lambda$ $(1 - F(\hat{m}_\lambda))$
XG - LR	−0.05 (0.058)	−0.06 (0.408)	0.04 (0.855)	−0.02 (0.473)	0.03 (0.835)	0.01 (0.509)
XG - RF	−0.05 (0.055)	−0.02 (0.468)	−0.08 (0.001)	−0.08 (0.367)	−0.08 (0.002)	−0.08 (0.375)
XG - DT	0.00 (0.481)	0.06 (0.598)	0.02 (0.728)	−0.03 (0.453)	0.02 (0.719)	0.05 (0.588)
LR - RF	0.01 (0.536)	0.04 (0.561)	−0.12 (0.000)	−0.07 (0.393)	−0.11 (0.000)	−0.08 (0.365)
LR - DT	0.06 (0.895)	0.11 (0.685)	−0.02 (0.300)	−0.01 (0.481)	−0.01 (0.316)	0.04 (0.579)
RF - DT	0.05 (0.888)	0.08 (0.629)	0.10 (0.999)	0.06 (0.595)	0.10 (0.999)	0.13 (0.715)

5. Conclusions

This paper introduces eigenvalue distributions for random confusion matrices obtained from a machine learning (ML) evaluation. Additionally, we derived distributions for the traces and the difference between traces from two ML methods. Our key finding is that the eigenvalues from a 2×2 random confusion matrix, denoted as $\boldsymbol{A}$, follow a shifted Gaussian Orthogonal Ensemble (GOE) with a mean of $\bar{\boldsymbol{A}}$ and a variance of s^2. Furthermore, the distribution of the trace of $\boldsymbol{A}$ follows a normal distribution with a mean and variance of $2\bar{\boldsymbol{A}}$ and $2s^2$, respectively. Similarly, the distribution of the difference of traces between two random confusion matrices, $\boldsymbol{A}$ and $\boldsymbol{B}$, is also normal with a mean and variance of $2(\bar{\boldsymbol{A}} + \bar{\boldsymbol{B}})$ and $2(s_{\boldsymbol{A}}^2 + s_{\boldsymbol{B}}^2)$, respectively. By way of illustration, our study presents bootstrap accuracy estimates and accuracy estimates for eigenvalue distribution across various ML methods and datasets. The findings suggest that both approaches yield similar accuracy estimates and standard errors, validating the effectiveness of the eigenvalue distribution method for ML accuracy estimation based on confusion matrix eigenvalues. Pairwise comparison results reveal consistent estimates of differences between ML models, yet significant variations exist in superiority probability estimates between bootstrap and eigenvalue distribution approaches. Notably, the bootstrap method tends to be conservative for negative differences and restrictive for positive ones. This underscores the importance of considering the actual distribution provided by the eigenvalue approach for more reliable superiority probability assessments.

Author Contributions: Conceptualization, O.R.O., A.R.R.A. and M.R.A.; methodology, O.R.O. and A.R.R.A.; software, O.R.O.; validation, M.R.A., O.R.O. and A.R.R.A.; formal analysis, O.R.O.; investigation, M.R.A., O.R.O. and A.R.R.A.; resources, M.R.A. and A.R.R.A.; data curation, O.R.O.; writing—original draft preparation, O.R.O.; writing—review and editing, M.R.A., O.R.O. and A.R.R.A.; visualization, O.R.O.; supervision, O.R.O.; project administration, O.R.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Chen, R.C.; Dewi, C.; Huang, S.W.; Caraka, R.E. Selecting critical features for data classification based on machine learning methods. *J. Big Data* **2020**, *7*, 52. [CrossRef]
2. Olaniran, O.R.; Abdullah, M.A.A. Bayesian weighted random forest for classification of high-dimensional genomics data. *Kuwait J. Sci.* **2023**, *50*, 477–484. [CrossRef]
3. Valero-Carreras, D.; Alcaraz, J.; Landete, M. Comparing two SVM models through different metrics based on the confusion matrix. *Comput. Oper. Res.* **2023**, *152*, 106131. [CrossRef]
4. Larner, A. *The 2 × 2 Matrix: Contingency, Confusion and the Metrics of Binary Classification*; Springer Nature: Cham, Switzerland, 2024.
5. Koço, S.; Capponi, C. On multi-class classification through the minimization of the confusion matrix norm. In Proceedings of the Asian Conference on Machine Learning. PMLR, Canberra, ACT, Australia, 13–15 November 2013; pp. 277–292.
6. García-Balboa, J.L.; Alba-Fernández, M.V.; Ariza-López, F.J.; Rodríguez-Avi, J. Analysis of thematic similarity using confusion matrices. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 233. [CrossRef]
7. Übeyli, E.D.; Güler, İ. Features extracted by eigenvector methods for detecting variability of EEG signals. *Pattern Recognit. Lett.* **2007**, *28*, 592–603. [CrossRef]
8. Božić, D.; Runje, B.; Lisjak, D.; Kolar, D. Metrics related to confusion matrix as tools for conformity assessment decisions. *Appl. Sci.* **2023**, *13*, 8187. [CrossRef]
9. Freeman, V. Production and perception of prevelar merger: Two-dimensional comparisons using Pillai scores and confusion matrices. *J. Phon.* **2023**, *97*, 101213. [CrossRef] [PubMed]
10. Sayyad, S.; Shaikh, M.; Pandit, A.; Sonawane, D.; Anpat, S. Confusion matrix-based supervised classification using microwave SIR-C SAR satellite dataset. In Proceedings of the Recent Trends in Image Processing and Pattern Recognition: Third International Conference, RTIP2R 2020, Aurangabad, India, 3–4 January 2020; Revised Selected Papers, Part II 3; Springer: Singapore, 2021; pp. 176–187.
11. Reddy, G.T.; Reddy, M.P.K.; Lakshmanna, K.; Kaluri, R.; Rajput, D.S.; Srivastava, G.; Baker, T. Analysis of dimensionality reduction techniques on big data. *IEEE Access* **2020**, *8*, 54776–54788. [CrossRef]
12. Golub, G.H.; Van Loan, C.F. *Matrix Computations*; JHU Press: Baltimore, MD, USA, 2013.
13. Alamsyah, A.; Fadila, T. Increased accuracy of prediction hepatitis disease using the application of principal component analysis on a support vector machine. *J. Phys. Conf. Ser.* **2021**, *1968*, 012016.
14. Sifaou, H.; Kammoun, A.; Alouini, M.S. High-dimensional linear discriminant analysis classifier for spiked covariance model. *J. Mach. Learn. Res.* **2020**, *21*, 1–24.
15. Hasan, S.N.S.; Jamil, N.W. A Comparative Study of Hybrid Dimension Reduction Techniques to Enhance the Classification of High-Dimensional Microarray Data. In Proceedings of the 2023 IEEE 11th Conference on Systems, Process & Control (ICSPC), Malacca, Malaysia, 16 December 2023; pp. 240–245.
16. Lu, J.; Lu, Y. A priori generalization error analysis of two-layer neural networks for solving high dimensional Schrödinger eigenvalue problems. *Commun. Am. Math. Soc.* **2022**, *2*, 1–21. [CrossRef]
17. Caelen, O. A Bayesian interpretation of the confusion matrix. *Ann. Math. Artif. Intell.* **2017**, *81*, 429–450. [CrossRef]
18. Olaniran, O.R.; Alzahrani, A.R.R. On the Oracle Properties of Bayesian Random Forest for Sparse High-Dimensional Gaussian Regression. *Mathematics* **2023**, *11*, 4957. [CrossRef]
19. Olaniran, O.; Abdullah, M. Subset selection in high-dimensional genomic data using hybrid variational Bayes and bootstrap priors. *J. Phys. Conf. Ser.* **2020**, *1489*, 012030.
20. Pudjihartono, N.; Fadason, T.; Kempa-Liehr, A.W.; O'Sullivan, J.M. A review of feature selection methods for machine learning-based disease risk prediction. *Front. Bioinform.* **2022**, *2*, 927312. [CrossRef] [PubMed]
21. Mehmood, T.; Sæbø, S.; Liland, K.H. Comparison of variable selection methods in partial least squares regression. *J. Chemom.* **2020**, *34*, e3226. [CrossRef]
22. Chen, C.W.; Tsai, Y.H.; Chang, F.R.; Lin, W.C. Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. *Expert Syst.* **2020**, *37*, e12553. [CrossRef]
23. Wang, G.; Sarkar, A.; Carbonetto, P.; Stephens, M. A simple new approach to variable selection in regression, with application to genetic fine mapping. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2020**, *82*, 1273–1300. [CrossRef] [PubMed]
24. Sauerbrei, W.; Perperoglou, A.; Schmid, M.; Abrahamowicz, M.; Becher, H.; Binder, H.; Dunkler, D.; Harrell, F.E.; Royston, P.; Heinze, G.; et al. State of the art in selection of variables and functional forms in multivariable analysis—Outstanding issues. *Diagn. Progn. Res.* **2020**, *4*, 1–18. [CrossRef] [PubMed]
25. Chowdhury, M.Z.I.; Turin, T.C. Variable selection strategies and its importance in clinical prediction modelling. *Fam. Med. Community Health* **2020**, *8*, e000262. [CrossRef] [PubMed]
26. Peyrache, A.; Rose, C.; Sicilia, G. Variable selection in data envelopment analysis. *Eur. J. Oper. Res.* **2020**, *282*, 644–659. [CrossRef]
27. Montoya, A.K.; Edwards, M.C. The poor fit of model fit for selecting number of factors in exploratory factor analysis for scale evaluation. *Educ. Psychol. Meas.* **2021**, *81*, 413–440. [CrossRef]
28. Greenacre, M.; Groenen, P.J.; Hastie, T.; d'Enza, A.I.; Markos, A.; Tuzhilina, E. Principal component analysis. *Nat. Rev. Methods Primers* **2022**, *2*, 100. [CrossRef]

29. Popoola, J.; Yahya, W.B.; Popoola, O.; Olaniran, O.R. Generalized self-similar first order autoregressive generator (gsfo-arg) for internet traffic. *Stat. Optim. Inf. Comput.* **2020**, *8*, 810–821. [CrossRef]
30. Sarkar, A.; Kothiyal, M.; Kumar, S. Distribution of the ratio of two consecutive level spacings in orthogonal to unitary crossover ensembles. *Phys. Rev. E* **2020**, *101*, 012216. [CrossRef] [PubMed]
31. Grimm, U.; Römer, R.A. Gaussian orthogonal ensemble for quasiperiodic tilings without unfolding: R-value statistics. *Phys. Rev. B* **2021**, *104*, L060201. [CrossRef]
32. Janosi, A.S.W.P.M.; Detrano, R. Heart Disease. UCI Machine Learning Repository. 1988. Available online: https://archive.ics.uci.edu/dataset/45/heart+disease (accessed on 1 March 2024).
33. Zwitter, M.; Soklic, M. Breast Cancer. UCI Machine Learning Repository. 1988. Available online: https://archive.ics.uci.edu/dataset/14/breast+cancer (accessed on 1 March 2024).
34. Ramana, B.; Venkateswarlu, N. ILPD (Indian Liver Patient Dataset). UCI Machine Learning Repository. 2012. Available online: https://archive.ics.uci.edu/dataset/225/ilpd+indian+liver+patient+dataset (accessed on 1 March 2024).
35. Ding, N.; Sadeghi, P. A submodularity-based agglomerative clustering algorithm for the privacy funnel. *arXiv* **2019**, arXiv:1901.06629.
36. Navarro, C.L.A.; Damen, J.A.; Takada, T.; Nijman, S.W.; Dhiman, P.; Ma, J.; Collins, G.S.; Bajpai, R.; Riley, R.D.; Moons, K.G.; et al. Risk of bias in studies on prediction models developed using supervised machine learning techniques: Systematic review. *BMJ* **2021**, *375*, n2281. [CrossRef]
37. Tantithamthavorn, C.; McIntosh, S.; Hassan, A.E.; Matsumoto, K. An empirical comparison of model validation techniques for defect prediction models. *IEEE Trans. Softw. Eng.* **2016**, *43*, 1–18. [CrossRef]