A classification performance evaluation measure considering data separability

Lingyan Xue¹, Xinyu Zhang¹*, Weidong Jiang¹ and Kai Huo¹

¹College of Electronic Science and Technology, National University of Defense Technology, Deya District, Changsha, 410000, Hunan, China.

*Corresponding author(s). E-mail(s): zhangxinyu90111@163.com; Contributing authors: xly1524@nudt.edu.cn; jwd2232@vip.163.com; huokai2001@163.com;

Abstract

Machine learning and deep learning classification models are data-driven, and the model and the data jointly determine their classification performance. It is biased to evaluate the model’s performance only based on the classifier accuracy while ignoring the data separability. Sometimes, the model exhibits excellent accuracy, which might be attributed to its testing on highly separable data. Most of the current studies on data separability measures are defined based on the distance between sample points, but this has been demonstrated to fail in several circumstances. In this paper, we propose a new separability measure—the rate of separability (RS), which is based on the data coding rate. We validate its effectiveness as a supplement to the separability measure by comparing it to four other distance-based measures on synthetic datasets. Then, we demonstrate the positive correlation between the proposed measure and recognition accuracy in a multi-task scenario constructed from a real dataset. Finally, we discuss the methods for evaluating the classification performance of machine learning and deep learning models considering data separability.

Keywords: machine learning, deep learning, classification accuracy, data separability, classification difficulty, performance evaluation
A classification performance evaluation measure considering data separability

1 Introduction

As an important branch in data mining, classification aims to construct a classification model to learn a mapping regularity from existing data to class labels. The research of model is essential, yet data also determines the performance [1]. A specific example is the impact of spectral separability on classification accuracy [2]. Numerous classification models have been proposed, including KNN, SVM, logistic regression, neural networks, etc., but studies on data separability are substantially fewer. A recent study in hyperspectral image classification has argued that insufficient data may limit the assessment capability of existing accuracy indexes [3]. That leads to the problem of whether a model performs best on a classification case is unclear or inconclusive [4, 5]. It is acknowledged that a good classification model provides greater generalization potential, which means finding rules consistent with available data that apply widely to predict the class of unknown data [6]. Yet the criteria for assessing the model’s generalization ability remain debated. To simplify the performance evaluation process, researchers generally tend to adopt measures based on the confusion matrix [7], like accuracy, precision, kappa statistic, and F-score. Each measure is represented with a single score number, making it straightforward to compare and analyze classification models quantitatively. Although the result is intuitive, its comparability is invalid when confronted with a multi-task classification situation more representative of the real-world environment.

A contradictory example is that a classifier reaches the highest accuracy in one task but the lowest in another. What causes the problem is that such classifier-oriented measures treat the different instances of a dataset as statistical objects and ignore the classification difficulty of each instance. For the above issue, Yu et al. [8] proposed an instance-oriented measure but only apply to data with few samples due to the computational complexity of classification difficulty for each instance. Therefore, we require a measure to statistically characterize the classification difficulty of datasets. Fortunately, previous research has established that separability is an intrinsic characteristic of a dataset [9] to describe how instances belonging to different classes mix. Measuring the data quality is critical for estimating the problem’s difficulty in advance since a classification model’s accuracy strongly depends on the data quality [10]. Obviously, the more separable the dataset, the simpler the classification. Eventually, we consider data separability as a metric of classification difficulty.

Such a metric would be helpful in the following ways: First, the metric is used to evaluate the dataset. Specifically, experts can use it to build a new standard dataset according to a list of desired levels of classification difficulty. In addition, if necessary, the classification difficulty can be adjusted by modifying the dataset. Second, this metric aids in selecting the dataset most appropriate for the system objectives. And even if only one dataset is available, it is beneficial to know its classification difficulty in advance. Researchers can predict the range of classification accuracy based on difficulty information. If the precision
is lower than the expected value, the algorithm’s utilization may be wrong. Finally, the performance evaluation can be more objective. For instance, if the metric of a personal dataset and its performance results are provided simultaneously, we can fairly judge the model’s performance by comparing it to that of a standard dataset with known classification difficulty. Similarly, the performance on two recognition tasks with varying classification difficulty will be comparable.

There are several measures of data separability that can quantify classification difficulty. Fisher discriminant ratio (FDR) [11] has been used in many studies, which measures the data separability using each class’s mean and standard deviation. Similarly inspired by FDR, Generalized Discrimination Value (GDV) [12], which has dimensional invariance, is also proposed to quantify the separability of data classes in neural networks. But the above feature-based measures are proved to fail in some cases, like two-class circle data in Figure 3(c).

A more practical issue is data complexity, which primarily measures the Euclidean distance between intra-class and inter-class data. Ho and Basu [13] conducted a groundbreaking review of data complexity measures. Recently, Lorena et al. [14] reviewed existing methods for measuring classification complexity, showing that some of those may have enormous time costs. The measures above work based on the linearity assumption of data and, therefore, do not identify properly clusters characterized by nonlinear structures. To overcome this limitation, Maria and Paolo [15] summarized several approaches: density-, kernel-, graph- or manifold-based methods. However, such distance-based methods for assessing data separability still leave an open question, namely, in high dimensional space, the boundary conditions of the data to which various distance metrics (such as Euclidean, Manhattan, and Minkowski distance) are applicable [16].

As an alternative to the distance-based criterion, we consider explaining data separability from the perspective of probability theory. According to the Bayesian criterion, when each category has an equal probability of occurrence, the result of sample partitioning depends on the magnitude of the likelihood function corresponding to each class. At this point, the more significant the gap between the likelihood functions of the various categories, the more separable the data. However, due to the complexity of high-dimensional data, the form of the likelihood function and parameter estimation are uncertain, which brings low classification accuracy. As inspired by Cover and Thomas [17], the process of minimizing the data rate distortion is equivalent to the operation of solving the optimal solution of the likelihood function, i.e., the data coding rate has strong consistency with the parameter estimation performance [18]. That means if the data can be fitted with a better distribution model after segmentation, then the data should be effectively encoded with such a model. Ma et al. [19] argued that the coding rate (subject to distortion) provides a natural measure of the goodness of segmentation for real-valued mixed data.
A classification performance evaluation measure considering data separability

Since there is no research verifying the feasibility of using coding rate as a measure of data separability, this is the first study to construct a separability measure based on rate-distortion theory called the rate of separability (RS). The main contributions of this paper are summarized as follows.

- We proposed a data separability measure based on rate-distortion theory, derived a new form applied to non-zero mean data, and verified its effectiveness in theory and experiments.
- We found a positive correlation between classification accuracy and data separability in a multi-task noisy environment.
- In a multi-task noisy environment, we designed a task-oriented classifier performance evaluation method considering data separability as the task difficulty. Unlike the classification accuracy changing with different tasks, this method obtains classifier ability as the classifier’s inherent property under certain assumptions.
- We built a modular classifier performance evaluation model to explain the function of deep learning convolutional blocks using data separability.

The rest of this paper is structured as follows. Section 2 introduces the method of constructing coding-rate-based measure and the theory of distance-based measures. Section 3 provides experimental methods for validating measure validity. Section 4 evaluates the classification model performance; results and analysis are also given in this section. Finally, we conclude in Section 5.

2 Measures of data separability

Whether or not data is challenging to separate depends on its confusion level. For example, if data points from the interclass are mixed, and data points from the intra-class are scattered, the data is difficult to classify. Such a condition appears to be described by the inter and intra-class distance. Specifically, we expect the intraclass data to be closer and the inter-class data to be further away. Guan et al. [20] recently demonstrated that several data separability measures (e.g., DSI, N2, LSC, Density) based on the inter and intra-class distance perform well in a two-class dataset. Due to the calculation of the inter and intra-class distance, these algorithms may have to traverse almost every data point at times, resulting in massive computation.

Another point of view comes from the information theory. A system’s entropy can be used to measure how chaotic it is. Similarly, data’s level of confusion can be defined by data’s entropy. When the data points are scattered in various regions of the space, the entropy of the data can be calculated by the proportion of the data occupying the region [21]. Unfortunately, entropy is not suitable for continuous random variables or high-dimensional space. Another information theory concept that quantifies the ”compactness” of a random distribution was developed to address this problem. It is known as rate-distortion
A classification performance evaluation measure considering data separability

[17]. Notably, it can be accurately and efficiently computed in closed form. Ma et al. [19] provide a precise estimation of rate-distortion on a subspace and a mixture of subspaces. In recent studies, rate-distortion has been used to explain or construct deep networks [22, 23]. Yet there is no research verifying the feasibility of using rate-distortion as a measure of data separability.

Therefore, part of the work of this paper is to verify the effectiveness of the proposed separability measure based on rate-distortion theory and compare it with distance-based measures. In this section, we briefly introduce the distance-based separability measures and then apply rate-distortion theory in constructing a new data separability measure.

2.1 Distance based data separability measures

Here we introduce several distance-based data separability measures which effectively assess typical two-class datasets [1].

1). DSI

DSI was built by measuring the Kolmogorov-Smirnov similarity between intra-class distance distribution \( \{d_{\mathbf{x}_i}\} \) and inter-class distance distribution \( \{d_{\mathbf{x}_i, \bar{\mathbf{x}}_i}\} \) :

\[
DSI = \frac{1}{k} \sum_{j=1}^{k} KS(\{d_{\mathbf{x}_j}\}, \{d_{\mathbf{x}_j, \bar{\mathbf{x}}_j}\}).
\] (1)

2). N2

To calculate N2, we first compute the Euclidean distance between each point \( \mathbf{x}_i \), its nearest neighbor \( NN(\mathbf{x}_i) \) and its nearest enemy \( NE(\mathbf{x}_i) \). Then we have \( d(\mathbf{x}_i, NN(\mathbf{x}_i)) \) and \( d(\mathbf{x}_i, NE(\mathbf{x}_i)) \), take the sum of all the distances to \( d(\mathbf{x}_i, NN(\mathbf{x}_i)) \) and \( d(\mathbf{x}_i, NE(\mathbf{x}_i)) \), the ratio of two sum results is defined as N2:

\[
N2 = \frac{\sum_{i=1}^{m} d(\mathbf{x}_i, NN(\mathbf{x}_i))}{\sum_{i=1}^{m} d(\mathbf{x}_i, NE(\mathbf{x}_i))}.
\] (2)

3). LSC

The Local Set (LS) includes the points \( \mathbf{x}_j (j \neq i) \) whose distance to \( \mathbf{x}_i \) is smaller than the distance from \( \mathbf{x}_i \) to \( \mathbf{x}_i \)’s nearest enemy \( NE(\mathbf{x}_i) \).

\[
LS(\mathbf{x}_i) = \{ \mathbf{x}_j | d(\mathbf{x}_i, \mathbf{x}_j) < d(\mathbf{x}_i, NE(\mathbf{x}_i)) \}
\] (3)

This measure is to count the number of \( \mathbf{x}_i \).

\[
LSC = 1 - \frac{1}{m^2} \sum_{i=1}^{m} |LS(\mathbf{x}_i)|
\] (4)

4). Density
A classification performance evaluation measure considering data separability

To use this measure, it is necessary to represent the classification dataset as a graph \( G = (V, E) \) where \( |V| = m \), \( 0 \leq |E| \leq \frac{m(m-1)}{2} \). Two nodes \( x_i \) and \( x_j \) are connected only if \( d(x_i, x_j) < \varepsilon \), then the measure is given by:

\[
Density = 1 - \frac{2|E|}{m(m - 1)}.
\] (5)

5). transform all measures into \([0,1]\)

To facilitate comparability, we transform the values of some of the measures so that all range in \([0, 1]\) with low values meaning high separability. Since RS, LSC, and Density have fulfilled the conditions, the modifications of DSI and N2 are as follows.

The DSI values are already in the range \([0, 1]\), but the low values indicate low separability. Hence, we modify it as

\[
DSI \leftarrow 1 - DSI.
\] (6)

originally, the N2 values are in range \([0, \infty)\). Hence, we shall use

\[
N2 \leftarrow 1 - \frac{1}{1 + N2}.
\] (7)

2.2 Coding-rate-based data separability measure

2.2.1 definition and computation of the coding rate

According to Cover and Thomas’ [17] definition of rate-distortion: given a data \( X = [x_1, ..., x_m] \in \mathbb{R}^{d \times m} \) with \( m \) samples of \( d \) dimension and a encoding precision \( \varepsilon > 0 \), the rate-distortion \( R(X, \varepsilon) \) is the minimal number of binary bits needed to encode \( X \) and the expected decoding error is less than \( \varepsilon \). The actual estimation coding rate of \( X \) with zero mean is as follows:

\[
R(X, \varepsilon) = \frac{m}{2} \log \det(I + \frac{d}{m \varepsilon^2} XX^T).
\] (8)

Furthermore, suppose \( X \) has \( k \)-class samples, then \( X = X^1 \cup X^2 \cup ... \cup X^k \) . the data \( X^j \) in each class \( j \) also occupy a certain volume in its low dimensional subspace. For each subset, the above coding rate (8) is applied. To determine the per-class sample’s overall coding rate \( R_C(X, \varepsilon|\Pi) \), let \( \Pi = \{\Pi^j \in \mathbb{R}^{m \times m}\}_{j=1}^k \) be the label matrix of the \( X \) in the \( k \) classes, and \( \Pi^j(i, i) \) is the probability of \( x_i \) belonging to class \( j \), then \( R_C(X, \varepsilon|\Pi) \) is given by

\[
\sum_{j=1}^{k} R(X^j, \varepsilon) = R_C(X, \varepsilon|\Pi)
= \sum_{j=1}^{k} \frac{\text{tr}(\Pi^j)}{2} \log \det \left( I + \frac{d}{\text{tr}(\Pi^j)\varepsilon^2} X\Pi^j X^T \right) \] (9)

The difference between \( R(X^1, \varepsilon) + R(X^2, \varepsilon) \) and \( R(X^1 \cup X^2, \varepsilon) \) can be described as Figure 1 in a two-class dataset.
A classification performance evaluation measure considering data separability

2.2.2 Derivation of coding rate for non-zero mean data

The equation for the coding rate in Section 2.2.1 is for the scenario where the mean value of the given data is zero mean. More generally, in this section, we derive the formula for the coding rate when the data’s mean is non-zero.

When \( X = [x_1, \ldots, x_m] \in \mathbb{R}^{d \times m} \) is not zero mean, we have \( \mu = \frac{1}{m} \sum_{i=1}^{m} x_i \in \mathbb{R}^d \). Here we disassemble the data into two components: the zero mean part of the data and the mean of the data, that is \( X = \bar{X} + V \). And we define \( V = \mu \cdot 1_{1 \times m} = [\mu, \mu, \ldots, \mu] \in \mathbb{R}^{d \times m} \). Then, the error induced on the matrix \( X \) is

\[
\delta X = \delta \bar{X} + \delta V = \delta \bar{X} + \delta \mu \cdot 1_{1 \times m}. \tag{10}
\]

Supposing that \( \delta \bar{X} \) and \( \delta V \) are zero-mean independent random variables, the expected total squared error is

\[
E(\text{tr}(\delta X\delta X^T)) = E(\text{tr}(\delta \bar{X}\delta \bar{X}^T)) + mE(\text{tr}(\delta \mu^T \delta \mu)) \tag{11}
\]

Now, we define the precision \( \varepsilon' = \frac{\varepsilon}{\sqrt{d}} \) encoding each entry: \( \bar{x}_{ij} \) and \( \mu_i \). And the error \( \delta \bar{x}_{ij} \) and \( \delta \mu_i \) are uniformly distributed in the interval \([\varepsilon / \sqrt{d}, \varepsilon / \sqrt{d}]\). We can compute the expected total squared error of \( \bar{X} \) and \( \mu \) are

\[
E(\text{tr}(\delta \bar{X}\delta \bar{X}^T)) = m \times d \times \frac{(\frac{2\varepsilon}{\sqrt{d}})^2}{12} = \frac{m\varepsilon^2}{3}
\]

\[
E(\text{tr}(\delta \mu^T \delta \mu)) = d \times \frac{(\frac{2\varepsilon}{\sqrt{d}})^2}{12} = \frac{\varepsilon^2}{3}. \tag{12}
\]

The total squared error satisfies
A classification performance evaluation measure considering data separability

\[
E(\text{tr}(\delta X \delta X^T)) = \frac{m \varepsilon^2}{3} + \frac{m \varepsilon^2}{3} = \frac{2m \varepsilon^2}{3} < m \varepsilon^2. \quad (13)
\]

Then, the mean squared error per vector in \(X\) is

\[
\frac{1}{m} E(\text{tr}(\delta X \delta X^T)) < \varepsilon^2. \quad (14)
\]

Since \(\varepsilon' = \frac{\varepsilon}{\sqrt{d}}\) is zero mean, we apply the conclusion in formula (8), its coding rate with precision \(\varepsilon' = \frac{\varepsilon}{\sqrt{d}}\) is

\[
R(\bar{X}) = \frac{1}{2} \log \det(I + \frac{d}{m \varepsilon^2} \bar{X} \bar{X}^T). \quad (15)
\]

And the number of bits to encoding the mean vector \(\mu\) with precision \(\varepsilon' = \frac{\varepsilon}{\sqrt{d}}\) is

\[
R(\mu) = \sum_{i=1}^{d} \frac{1}{2} \log_2 \left(1 + \frac{(\mu_i)^2}{\varepsilon^2}\right)
= \frac{1}{2} \sum_{i=1}^{d} \log_2 \left(1 + \frac{d \mu_i^2}{\varepsilon^2}\right) \leq \frac{d}{2} \log_2 \left(1 + \frac{\mu^T \mu}{\varepsilon^2}\right). \quad (16)
\]

The above inequality is from the following formula (17)

\[
\frac{\log_2(1 + \mu_1) + \log_2(1 + \mu_2) + \cdots + \log_2(1 + \mu_d)}{d}
\leq \log_2 \left(1 + \frac{\mu_1 + \mu_2 + \cdots + \mu_d}{d}\right),
\]

for nonnegative real numbers \(\mu_1, \mu_2, \ldots, \mu_d \geq 0\). Thus, the total coding rate of \(X\) with non-zero mean is

\[
R(X) = R(\bar{X}) + R(\mu)
= \frac{m}{2} \log \det(I + \frac{d}{m \varepsilon^2} \bar{X} \bar{X}^T) + \frac{d}{2} \log_2 \left(1 + \frac{\mu^T \mu}{\varepsilon^2}\right). \quad (18)
\]

2.2.3 correlation between coding rate and data separability

This section discusses the connection between data encoding rate and data separability. Under the condition that the data follows a Gaussian distribution, we prove Theorem 1. Theorem 1 gives the lower bound of the data coding rate and the necessary and sufficient conditions for it to reach the lower bound. This condition illustrates that if and only if every class \(X\) has the same distribution, the total coding rate of \(X\) is identical to the sum of \(X^j\)’s coding rate.

**Theorem 1** For any \(\left\{X^j \in \mathbb{R}^{d \times m_j}\right\}_{j=1}^k\) and any \(\varepsilon > 0\), let \(X = [x_1, \ldots, x_m] = [X^1, \ldots, X^k] \in \mathbb{R}^{d \times m}\) with \(m = \sum_{j=1}^{k} m_j\) and \(\mu = \frac{1}{m} \sum_{i=1}^{m} x_i \in \mathbb{R}^d\), then we define
A classification performance evaluation measure considering data separability

the zero mean part $\bar{X} = X - \mu \cdot 1_{1 \times m}$. Let $X^j = [x^j_1, ..., x^j_m] \in \mathbb{R}^{d \times m_j}$ with $\mu^j = \frac{1}{m_j} \sum_{i=1}^{m_j} x^j_i \in \mathbb{R}^d$ and $\bar{X}^j = X^j - \mu^j \cdot 1_{1 \times m_j}$. We have

$$\frac{m}{2} \log \det(I + \frac{d}{m_j} \bar{X}^j \bar{X}^j^T) + \frac{d}{2} \log_2 \left(1 + \frac{\mu^T \mu}{\varepsilon^2}\right) \geq \sum_{j=1}^{k} \frac{m_j}{2} \log \det(I + \frac{d}{m_j} \bar{X}^j \bar{X}^j^T) + \frac{d}{2k} \log_2 \left(1 + \frac{(\mu^j)^T \mu^j}{\varepsilon^2}\right). \tag{19}$$

where the equality holds if and only if

$$\frac{\bar{X}^1(\bar{X}^1)^T}{m_1} = \frac{\bar{X}^2(\bar{X}^2)^T}{m_2} = \cdots = \frac{\bar{X}^k(\bar{X}^k)^T}{m_k} = \frac{\bar{X}(\bar{X})^T}{m} \tag{20}$$

Since the covariance matrix of the vector $x_i^j$ is

$$\sum^j = \mathbb{E} \left[ \frac{1}{m_j} \sum_{i=1}^{m_j} (x^j_i - \mu^j)(x^j_i - \mu^j)^T \right] = \frac{\bar{X}^j(\bar{X}^j)^T}{m_j}, \tag{21}$$

this means that the lower bound is tight when all the components $\{X^j\}_{j=1}^k$ have the same covariance and mean. In particular, for data with a Gaussian distribution, it means that every $X^j$ has the same distribution, which corresponds to the most inseparable case of data.

The Proof of Theorem 1 is based on the concave property of the $\log \det(\cdot)$ and $\log(\cdot)$ functions, and they satisfy Jensen’s inequality.

**Proof** Since $\log \det(\cdot)$ and $\log(\cdot)$ is strictly concave, The Jensen’s inequality is satisfied. We have

$$f(\sum^k_{j=1} \beta_j S^j) \geq \sum^k_{j=1} \beta_j f(S^j). \tag{22}$$

for all $\{\beta_j > 0\}_{j=1}^k$, $\sum^k_{j=1} \beta_j = 1$ and $\{S^j \in \mathbb{S}_{++}^d\}_{j=1}^k$, where equality holds if and only if $S^1 = S^2 = \cdots = S^k$.

For function $\log \det(\cdot)$, take $\beta^j = \frac{m_j}{m}$ and $S^j = I + \frac{d}{m_j \varepsilon^2} \bar{X}^j(\bar{X}^j)^T$, we get

$$\log \det(I + \frac{d}{m_j \varepsilon^2} \bar{X}^j \bar{X}^j^T) \geq \sum_{j=1}^{k} \frac{m_j}{m} \log \det(I + \frac{d}{m_j \varepsilon^2} \bar{X}^j \bar{X}^j^T). \tag{23}$$

with equality holds if and only if $\frac{\bar{X}^1(\bar{X}^1)^T}{m_1} = \frac{\bar{X}^2(\bar{X}^2)^T}{m_2} = \cdots = \frac{\bar{X}^k(\bar{X}^k)^T}{m_k} = \frac{\bar{X}(\bar{X})^T}{m}.

For function $\log(\cdot)$, take $\beta^j = \frac{1}{k}$ and $S^j = 1 + \frac{d}{\varepsilon^2} (\mu^j)^T \mu^j$, we get

$$\log(1 + \frac{\mu^T \mu}{\varepsilon^2}) \geq \sum_{j=1}^{k} \frac{1}{k} \log(1 + \frac{(\mu^j)^T \mu^j}{\varepsilon^2}). \tag{24}$$
A classification performance evaluation measure considering data separability

with equality holds if and only if $\mu^1 = \mu^2 = \cdots = \mu^k = \mu$, and the last equality is from $\sum_{j=1}^{k} m_j \mu^j = m \mu$.

From formula (23) and (24), Theorem 1 can be proved. \hfill \square

We can now conclude that the sum of the various classes of data coding rate is a lower bound on the overall data coding rate. When the overall data coding rate reaches the lower bound, its necessary and sufficient condition indicate: for the Gaussian distributed data, each category of data has the same distribution, which also means that the feature vectors of each class have a high degree of coincidence, corresponding to the most inseparable situation. In a two-class problem, Figure 2 shows the different separability of two datasets and vividly reflects the above point of view. We can suppose that as $X$ space expands, each $X_j$ is compressed, making it easier to classify.

Fig. 2 Different separability of two datasets. Compared with (b), (a) has a larger total space volume and a smaller intra-class space volume, which seems to have better separability.

According to the preceding analysis, our proposed data separability measure based on rate-distortion is:

$$R_S(X) = \frac{R_C(X, \varepsilon | \Pi)}{R(X, \varepsilon)}.$$  \hfill (25)

Unlike Yu et al. [24], who utilized $\Delta R(X) = R(X, \varepsilon) - R_C(X, \varepsilon | \Pi)$ as the optimization problem’s objective function subjecting to $\|X_j\|_F^2 = \text{tr}(\Pi^j)$, here we discard the constraint and adopt a ratio form between $R(X, \varepsilon | \Pi)$ and $R(X, \varepsilon)$, resulting in a data separability measure $R_S(X)$ in the range of [0,1] with low values indicating high separability.
A classification performance evaluation measure considering data separability

3 Experiments

We first verify the proposed measure RS’s effectiveness using a two-class synthetic dataset\(^1\) with adjustable separability, and contrast its separability evaluation results with DSI, N2, LSC, and Density.

3.1 datasets with different feature shapes

The most basic type of data separability analysis is the linear separability of data [25]. Linear separability for a two-class dataset is defined as: There exists a hyperplane, which can correctly divide the positive and negative instance points to both sides of the hyperplane. An effective measure must first be capable of measuring data linear separability. Therefore, we created two-class datasets with six distinct feature shapes, showing clearly discernible linear separability. Each class consists 1000 samples of 2 dimensions. Figure 3 displays datasets feature plots and Table 1 lists the measures’ performance.

\(^1\)These datasets are created by the Samples Generator in sklearn.datasets https://scikit-learn.org/stable/modules/classes.html#samples-generator

Fig. 3 Typical two-class datasets: separability is decreasing from Figure (a) to Figure (f).

According to Table 1, we can conclude that RS can correctly represent the relative gap of linear separability among datasets while maintaining excellent comparability accuracy to four decimal places. In general, the other measures can also roughly reflect relative separability. But in terms of details, DSI deviated from the separability analysis of the Circle data. When assessing the Random data, N2 has increased significantly. LSC confuses the separability of XOR and Spirals, And the Density’s evaluation results of Circle, XOR, and Spirals are unclear.
A classification performance evaluation measure considering data separability

Table 1 Separability measures results for the two-class datasets (Figure 3)

| Measures | Blobs | Mooms | Circles | XOR | Spirals | Random |
|----------|-------|-------|---------|-----|---------|--------|
| DSI      | 0.0008| 0.6413| 0.4483  | 0.7775| 0.9468  | 0.9992 |
| N2       | 0.0007| 0.0014| 0.0011  | 0.0127| 0.0789  | 0.8140 |
| LSC      | 0.5137| 0.9083| 0.8400  | 0.9997| 0.9997  | 0.9999 |
| Density  | 0.0758| 0.6707| 0.5580  | 0.7714| 0.7406  | 0.8565 |
| RS(Proposed) | 0.1402| 0.6560| 0.8571  | 0.8700| 0.9847  | 0.9982 |

A low value indicates high separability.

3.2 datasets with different overlap regions

Now consider another case in which the feature distribution exhibits a high degree of coincidence, making it challenging to categorize the data. Can data separability indicators measure this situation? To answer the question, we experiment on the Blobs data. Since the features of the Blobs data follow a Gaussian distribution, the region of feature overlap can be adjusted by changing the feature standard deviation (SD). We set the SD parameter from 1 to 9; four instances are depicted in Figure 4.

![Fig. 4](image_url) The Blobs data with different cluster standard deviations (SD). A high SD value denotes a significant overlap area.
Then, for the nine Blobs datasets, we compute DSI, N2, LSC, Density, and the proposed measure (RS) and present them together with SVM classification accuracy as a baseline for separability in Figure 5.

![Figure 5](image_url)

**Fig. 5** Comparison of data separability evaluation results with varying degrees of feature overlap. For measures, a high value on the y-axis indicates low separability.

In this condition, both N2 and LSC fail to assess the data separability. Among them, LSC can only distinguish the case of features with or without overlap and is not sensitive to the change of feature overlap area. At the same time, N2 fluctuates with the deterioration of data separability, suggesting a lower evaluation precision. Besides, RS, DSI, and Density can correctly reflect the trend of data separability, i.e., a high SD value corresponds to a high measure value. Furthermore, both DSI and RS have an extensive dynamic change range.

### 3.3 datasets with different preprocessing methods

Feature preprocessing is one of the most prevalent approaches to ensure the classification performance of the classifier. Some types of preprocessing are employed before feeding data to the classifier or before the network extracts latent features from the data [26]. Therefore, the classifier’s data is often pre-processed data, not the original data. The purpose of this experiment is to examine the impact of preprocessing operations on the performance of the separability measure. Take the Blobs data (SD=1) as an example; Table 2 summarizes the common feature preprocessing methods.
A classification performance evaluation measure considering data separability

Table 2 Feature preprocessing methods

| Preprocessing methods     | Operations                                                                 | Examples |
|---------------------------|-----------------------------------------------------------------------------|----------|
| Min-max normalization     | $\mathbf{x} \leftarrow \frac{\mathbf{x} - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}$ | ![Image](image1) |
| Mean normalization        | $\mathbf{x} \leftarrow \frac{\mathbf{x} - \text{mean}(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}$ | ![Image](image2) |
| L2 normalization          | $\mathbf{x} \leftarrow \frac{\mathbf{x}_{\mathbf{||x\mathbf{||}}^2}}{||\mathbf{x}||^2}$ | ![Image](image3) |
| Centralization            | $\mathbf{x} \leftarrow \mathbf{x} - \text{mean}(\mathbf{x})$              | ![Image](image4) |
| Standardization           | $\mathbf{x} \leftarrow \frac{\mathbf{x} - \text{mean}(\mathbf{x})}{\text{std}(\mathbf{x})}$ | ![Image](image5) |

From Table 2, we can see that the Min-max, Mean normalization, Centralization, and Standardization change the range of feature values without changing the cluster shape. L2 normalization differs in that it alters the feature shape by normalizing the vector length to a unit circle. Except for L2 normalization, the data after each preprocessing shows no difference in separability from the feature maps. Hence, we conjecture that data preprocessing does not affect the separability of the data. So are the separability measures sensitive to the preprocessing methods? We tested RS, DSI, and density on the Blob (SD = 1, 2,...,9), and the results are shown in Figure 6.
A classification performance evaluation measure considering data separability

Figure 6 presents commonalities that RS, Density, and DSI can all indicate the relative gap in the separability of each dataset when exposed to various preprocessing methods. Yet the influence of preprocessing techniques on measures performance is different. With the same data separability (fixed SD), the value of RS is more sensitive to the preprocessing methods. It has the most significant variance in the L2 normalization group as the SD value increases. Density also exhibits a greater range of dynamic variation on the L2-normalization. DSI shows strong consistency in all preprocessing methods. Consequently, we use the preprocessing process of L2 normalization without modifying data separability to improve the proposed measure of RS’s evaluation performance.

3.4 datasets with different complexity decision boundaries

Including the constraint situation Guan et al. mentioned [20], we last examine RS’s performance when measuring the complexity of data points on or near the
A classification performance evaluation measure considering data separability
decision boundary. In this experiment, we construct a two-class dataset with
2000 samples of 2 features for each class. The feature maps and the expected
decision boundaries are shown in Figure 7.

We use SVM classification accuracy to reflect classification difficulty. Figure
8 compares the evaluation performance of RS, Density, and DSI on the datasets
(Figure 7).

![Fig. 8](image)

**Fig. 8** The complexity of boundary evaluation results. High measure value and low
classification accuracy correspond to high complexity levels.

Figure 8 shows that only DSI fails, but RS and Density achieve the neces-
sary progress, a growing trend as complexity levels. Although the increment
of measure value is slight with a whole range of 0.006, it explains that the tiny
change in RS value is meaningful.

4 Discussion

After thoroughly verifying the effectiveness of RS, we apply it to real noisy
datasets with unknown structures. Considering that the recognition accuracy
of the classifier on the dataset can reflect the complexity of real data to a cer-
tain extent, we use three machine learning classifiers as additional verification
tools: SVM, SGD, and KNN. Classifiers are tested in a noisy environment, i.e.,
the original dataset is retained as training data, and Gaussian white noise with
varying variances is added to it to generate a series of recognition tasks as test
data. Eventually, we confirm the correlation between the recognition accuracy
of the classifier in the recognition task and the separability evaluation results
of RS on noisy data.

Classifiers’ accuracy indicates data’s separability. And try another perspec-
tive, since the classifier is trained and tested based on data, its recognition
accuracy essentially relies on how good the data separability is. The part of the
discussion is intended to explain this dependence. Similarly, using a real noisy
dataset, we investigate the relationship between data separability and the trend
of classifier generalization ability in noisy environments. The key is to quantify
the classifier’s generalization ability. Specifically, under the premise of meeting
certain assumptions, we construct a mapping model from the known classifica-
tion accuracy to the classifier’s anti-noise ability, using RS as the parameter.
Furthermore, the mapping model can evaluate the anti-noise performance of
particular classifiers within certain limits.
In the last part, we discuss the deep learning model recognition performance evaluation method. As is known, increasing the depth of a neural network improves its ability to extract high-dimensional features. But so far, it is difficult to explain the substantive role of each module in the network. Consequently, we consider a modular evaluation method. The separability measure RS, in particular, is used to judge whether each convolution module works by evaluating the variation in the separability of the features extracted by each convolution layer in the deep convolution neural network (CNN).

4.1 Correlation between classification accuracy and data separability

After thoroughly verifying the validity of RS as a measure of data separability, we characterize the data separability using RS values. This section discusses the experimental procedures used to investigate the correlation between classification accuracy and data separability. In this study, a noisy environment with 16 sets of recognition tasks was designed based on real data. And to verify the rationality of constructed scenario, the T-SNE tool is used to visualize the influence of noise on data separability. And then, with the original data as a training set and the noisy data as a test set, the recognition accuracy of the four classifiers in noisy environments is obtained. Next, we compute noisy data’s RS value and analyze the classification accuracy correlation. Here we apply the Monte Carlo simulation to average the randomness of the results due to noise. The experiment framework is shown in Figure 9.

Step I is to add Gaussian white noise with a specific variance to the original data to create a test set with a signal-to-noise ratio (SNR) of 5-20 dB, and the formula involving the SNR computation is equation (26)

\[
SNR = 10 \log \frac{P_X}{N} = 10 \log \frac{\sum_{d=1}^{D} P_d}{N},
\]

where \(P_X\) is the average power of the original data \(X\), \(N\) is the noise’s average power, \(D\) is the \(X\) ’s feature dimension, and \(P_d\) is the power on the \(d\) dimension.

In Step II, we deploy four standard machine learning classifiers. Nonlinear classifiers such as K-Nearest Neighbor (KNN) and Support Vector Machine with Radial Basis Function (SVM with RBF) can generate nonlinear decision boundaries. Linear classifiers include linear SVM and logistic regression (LR). The UCI datasets [27] are utilized in the experiment. Table 3 and 4 present the detailed parameter setting of the classifiers and datasets².

Steps III and IV submit the final analysis results, as seen in Figure 10.

Since real data are usually high-dimensional, we apply the T-SNE tool to observe the variation of data separability with SNR. When the SNR is lower than 5 dB, the test data separability becomes extremely poor relative to the

²The URL for downloading the dataset:https://archive.ics.uci.edu/
A classification performance evaluation measure considering data separability

Fig. 9 Experiment procedure to verify the correlation between classification accuracy and data separability.

Table 3 Machine learning models

| Classifier     | Parameter setting                                      |
|----------------|-------------------------------------------------------|
| KNN            | number of nearest neighbor nodes $K = 5$             |
| SVM (RBF)      | regular coefficient $C = 1$                          |
| SVM (Linear)   | regular coefficient $C = 1$                          |
| LR             | regular coefficient $\alpha = 0.0001$                |

Table 4 UCI datasets

| Datasets | Instances | Features | Classes |
|----------|-----------|----------|---------|
| Mice     | 1080      | 77       | 8       |
| Iris     | 150       | 4        | 3       |
| Cancer   | 569       | 30       | 2       |

training data. In this condition, all kinds of sample points are scattered, the volume of the subspace in which they are located is expanded, and the proportion of the coding rate of the whole data space is increased. This phenomenon also well corresponds to our definition of RS. When the SNR is high as 20dB, its separability is equivalent to the training data, and the two have almost the same feature distribution. Referring to the separability evaluation and correlation analysis, as SNR grows, the separability of test data gradually improves,
A classification performance evaluation measure considering data separability

and the classification accuracy increases along with it. Finally, we can conclude a strong positive correlation between data separability and recognition accuracy.

4.2 Classifier’s ability evaluated by classification accuracy under data separability

We verify the positive correlation between classification accuracy and data separability in the previous discussion. For the same type of classifier with fixed parameters, the better the separability of the test data provided by the recognition task, the higher the recognition accuracy the classifier exhibits.
For different classifiers, we can explicitly compare the recognition accuracy of each classifier on an unknown separability recognition task, implying that each classifier’s recognition ability is directly given by its recognition accuracy. This conclusion applies only to the circumstance of classifier recognition performance evaluation under a single task. But in a realistic scenario, the classifier is faced with a set of recognition tasks in a complex recognition environment as constructed in Figure 9, and we indeed want to evaluate the classifier’s generalization ability in this group of tasks. Specifically, as shown in Figure 10(b), for the dataset Iris, the classification accuracy of SVM (RBF) is consistently higher than other classifiers at SNR=20dB, but the lowest at SNR=5dB. Since the 5dB task is more difficult than the 20dB one, we can’t conclude whether the classifier performance is good or not. At this point, how could the classifier’s performance be measured?

The simple idea is to assign a certain weight \( W \in \mathbb{R}^n \) to the recognition accuracy \( P_{\text{acc}} \in \mathbb{R}^n \) of the classifier on that group of tasks according to the difficulty of the recognition task, and \( n \) is the number of tasks. The classification ability \( \theta \) on these tasks is defined as

\[
\theta = W^T P_{\text{acc}}. \tag{27}
\]

\( W \) is determined by the difficulty of the recognition task. The more difficult the task, the higher the weight value. According to the prior experiment, the task difficulty depends to some extent on the data separability. Thus, \( W \) as a mapping matrix is parameterized by the separability \( R_S \). To quantify this mapping relationship, we seek a functional form \( f(\cdot) \) of the mapping matrix \( W \).

\[
\theta = f(P_{\text{acc}}; R_S) \tag{28}
\]

\( f(\cdot) \) needs to be obtained by fitting a given \( P_{\text{acc}} \) and \( \theta \). \( P_{\text{acc}} \) can be derived directly from the classification results, whereas \( \theta \) is uncertain. Therefore, it is first necessary to construct the known \( \theta \) based on the following assumptions.

1) For different difficulty tasks, homogeneous classifiers with fixed parameters exhibit different recognition accuracies.

2) For the same dataset, homogeneous classifiers with fixed parameters exhibit consistent recognition ability values.

3) For homogeneous classifiers with different parameter settings, their relative ability value can be inferred from the recognition accuracy.

Based on the assumptions stated above, we choose the SVM (Linear) model on the Iris dataset to perform the anti-noise experiment depicted in Figure 9. \( k \) SVMs with relative ability values \( \theta \) were obtained by adjusting the parameters \( C \) (Table 3). \( \theta \in \mathbb{R}^k \) take \( k \) values evenly from 0 to 1. Each SVM tests on \( n \) noisy tasks, and get \( P_{\text{acc}}^j \in \mathbb{R}^n (j = 1, 2, \ldots, k) \). \( k \)-group \( P_{\text{acc}}^j \) is sorted from small to large according to its largest element, and we have \( P_{\text{acc}} = [P_{\text{acc}}^1, P_{\text{acc}}^2, \ldots, P_{\text{acc}}^k] \in \mathbb{R}^{n \times k} \). Figure 11 shows the mapping of \( P_{\text{acc}} \) and \( \theta \).
A classification performance evaluation measure considering data separability

Fig. 11 The map of $P_{\text{acc}}$ and $\theta$. Here, $k = 30$, $n = 15$. Each column of $P_{\text{acc}}$ records the classification accuracy of an SVM model on 16 tasks associated with the same color point column in Figure (a). Each row of $P_{\text{acc}}$ records the classification accuracy of 30 SVM models on a single task. The row values are sequentially concatenated to obtain the task curve shown in Figure (b).

Observe that the shape of the curve in Figure 11(b) is more consistent with that of the Sigmoid function, but the upper and lower bounds of the task curve are variable; thus, equation (29) is adopted as the fitting function.

$$P_{\text{acc}} = \frac{u - l}{1 + \exp(-a \ast (\theta - b))} + u \quad (29)$$

Plotting the curve of equation (29) in Figure 12, we explore the properties of this function.

Fig. 12 Fitting function plot.
The parameters $u$, $l$, $a$, and $b$ reflect the function properties as follows.

1) $u$ and $l$ can represent the classifier’s upper and lower bounds of recognition accuracy on a set of recognition tasks, respectively.

2) $(u - l) \times a$ reflects the slope of the function. The flatter the function, the harder the task and the lower the recognition accuracy.

3) $b$ affects the right shift rate of the function. The larger the right shift magnitude, the more difficult the task is, and the less accurate the classifier is.

Then we research the relationship between $R_S$, $u$, $l$, $a$, and $b$, where $u$ and $l$ are determined by the upper and lower limits of classifier performance; while $a$ and $b$ depend on the difficulty of the task. The relationship verified on the Iris dataset is given by Figure 13.

**Fig. 13** the relationship between $R_S$, $u$, $l$, $a$, and $b$ on the Iris dataset. The lowest $R_S$ value corresponds to the lowest $b$ and the steepest slope $(u - l) \times a$, indicating that the task is simple. The highest $R_S$ value corresponds to the highest $b$ and the flattest slope $(u - l) \times a$, indicating that the task is the most challenging.

Then we employ the polynomial fitting approach, with $a$ and $b$ represented by $R_S$

\[ f_\alpha(R_S) = h_0 + h_1 R_S + h_2 R^2_S \]
A classification performance evaluation measure considering data separability

\[ f_b(R_S) = p_0 + p_1 R_S + p_2 R_S^2. \]  \hspace{1cm} (30)

The mapping function \( f^{-1}(\cdot) \) from classification accuracy to classifier ability with separability \( R_S \) as a parameter is now obtained.

\[ P_{acc} = f^{-1}(\theta; R_S) = \frac{u - l}{1 + \exp(-f_a(R_S) * (\theta - f_b(R_S)))} + u \]  \hspace{1cm} (31)

To examine the validity of this mapping function, we need to substitute the recognition accuracy of another classifier into the equation (31) to ensure the uniqueness of its recognition ability value, demonstrating that the recognition ability value exists as an inherent property of the classifier.

Figure 14 shows the results of fitting the task curve with the SVM model as a reference and an evaluation of the LR model’s recognition ability on this curve.

![Fig. 14 The fitting task curves and the mapping points.](image)

This evaluation method has a high assessment accuracy in the middle of the task curve. The evaluation of the LR model ability values for \( \alpha = 0.01 \) and \( \alpha = 0.1 \) are distributed over a small interval, and a set of recognition accuracies essentially map to a unique recognition ability value. Whereas at the two ends of the curve, a slight change in recognition accuracy may bring about a significant deviation in recognition ability due to the presence of the saturation zone.
A classification performance evaluation measure considering data separability

4.3 CNN layers’ performance evaluated by data separability

As an extension of machine learning classifiers, deep learning classifiers have greatly improved recognition performance but are not satisfactory in model interpretability. The abstraction lies in the fact that we cannot fully grasp the substantive role played by each module in the deep network. The convolution module, for example, is widely believed to play the role of feature extraction, enabling the final output data to be more separable. Nonetheless, is it better to have more convolutional modules? And is the contribution of each convolutional module to the extraction of potential features consistent? How do we measure this contribution? To answer the above questions, in this section, we design modular classifier recognition performance evaluation methods to evaluate the performance of each convolutional component of the CNN.

Effective separability indices are invaluable for the performance evaluation of radar signal classification algorithms [28]. Compared to the classification of optical images, radar target recognition requires more of a metric to clearly understand and grasp the substantial effect of each module when using a deep neural network. This is because the radar images are not very understandable for the non-expert. And it is more difficult to identify the classes to which they belong after their semantic features are extracted by the convolutional layer, which makes it more difficult to judge the meaning of the existence of each module. The evaluation method proposed is to insert a feature separability analysis module after each convolutional block to monitor its performance, and the separability measure used is RS.

We use the typical SAR image MSTAR\(^3\) as the experimental data, where the SOC dataset is shown in Table 5.

| Class   | Serial No. | Depression | No. Images | Depression | No. Images |
|---------|------------|------------|------------|------------|------------|
| BMP-2   | 9563       | 17°        | 233        | 15°        | 196        |
| BTR-70  | C71        | 17°        | 233        | 15°        | 196        |
| T-72    | 132        | 17°        | 232        | 15°        | 196        |
| BTR-60  | k10yt7532  | 17°        | 256        | 15°        | 195        |
| 2S1     | b01        | 17°        | 299        | 15°        | 274        |
| BRDM-2  | E-71       | 17°        | 298        | 15°        | 274        |
| D7      | 92v13015   | 17°        | 299        | 15°        | 274        |
| T-62    | A51        | 17°        | 299        | 15°        | 273        |
| ZIL-131 | E12        | 17°        | 299        | 15°        | 274        |
| ZSU-234 | d08        | 17°        | 299        | 15°        | 274        |

\(^3\)The URL for downloading the dataset: https://www.sdms.afrl.af.mil/datasets/mstar/
A classification performance evaluation measure considering data separability

Fig. 15 Network architecture and feature separability analysis module.

Fig. 16 Classification accuracy and feature separability analysis results.
On the dataset presented in Table 5, we evaluate the performance of some convolutional blocks of CNN provided by Chen et al. [29]. And to reduce the computational effort of RS, we apply a 2*2 average pooling to the feature map. The network structure and the feature separability analysis module are shown in Figure 15.

After 100 epochs of training, the convergence of recognition accuracy on the test set and the variation of feature separability extracted by each convolutional block are set out in Figure 16.

The most striking result from Figure 16 is that the separability of features extracted by each convolutional block keeps step with the final classification accuracy. When classification accuracy improves dramatically, the RS value falls precipitously. And when the classification accuracy converges, the RS value becomes steady. Furthermore, we also find that the convolutional block in a deeper network has a more significant function in extracting a more separable feature with a lower RS value ($RS_1 > RS_2 > RS_3 > RS_4$). And the deeper feature map exhibits a wider dynamic range of RS value ($\Delta RS_1 < \Delta RS_2 < \Delta RS_3 < \Delta RS_4$). The evidence above claims that all four convolutional blocks perform well in this network, and the deeper block does better work.

5 Conclusion

We propose a new separability measure based on the data encoding rate. The measure can capture the relative relationship between the expansion rate of the sample distribution in the whole space and the subspace’s compression rate. We validate the effectiveness of the proposed measure on a typical synthetic two-class dataset and confirm its positive correlation with the classification accuracy in a series of noisy tasks constructed from real datasets. We designed machine learning and deep learning classifier model evaluation methods based on the above two basic argumentation experiments. We build a mapping model for machine learning classifiers from classification accuracy to classifier ability. In the model, the task difficulty is characterized by the measure, and the classification accuracy assesses the value of the classifier capability as its inherent properties with a separability measure as a parameter. The method applies to a subset of classifiers under certain assumptions. For deep learning classifiers, we use a modular evaluation approach. The ability of each convolutional block at different depths of the network to extract separable features is discussed using the proposed measure.

Data separability quantification provides some basis for analyzing, understanding, and enhancing model performance. This paper focuses on its application in the recognition field. In fact, it can also be applied to evaluate clustering results [30], understand the demerit of each feature [31, 32], provide a theory for building multi-classifier decision [33], reduce data complexity as a loss function [34], and other fields. Furthermore, a more comprehensive metric should be served as a general standard. The values are still comparable when comparing datasets with different sample sizes and feature dimensions. The
existing separability measures still lack the above generalization properties, and this is a point that deserves further study.

**Declarations**

- **Conflict of interests** The authors declare that they have no conflict of interest.
- **Funding** This work was supported in part by National Natural Science Foundation of China under Grants 62022091 (Corresponding author: Xinyu Zhang.)
- **Data Availability Statements** The data in 4.1 and 4.2 that support the findings of this study are openly available in UCI Machine Learning Repository at https://archive.ics.uci.edu, reference number [28]. And the network we analyzed in 4.3 is derived at https://github.com/fudanxu/MSTAR-AConvNet, reference number [29].

**References**

[1] Cano, J.R.: Analysis of data complexity measures for classification. Expert Systems With Applications 40(12), 4820–4831 (2013). https://doi.org/10.1016/j.eswa.2013.02.025

[2] Wicaksonoa, P., Aryagunab, P.A.: Analyses of inter-class spectral separability and classification accuracy of benthic habitat mapping using multispectral image. Remote Sensing Applications: Society and Environment 19, 100335 (2020). https://doi.org/10.1016/j.rsase.2020.100335

[3] Li, S., Hao, Q., Gao, G., Kang, X.: The effect of ground truth on performance evaluation of hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing 56(12), 7195–7206 (2018). https://doi.org/10.1109/TGRS.2018.2849225

[4] Oprea, M.: A general framework and guidelines for benchmarking computational intelligence algorithms applied to forecasting problems derived from an application domain-oriented survey. Applied Soft Computing 89, 106103 (2020). https://doi.org/10.1016/j.asoc.2020.106103

[5] Lee, Y., Lee, J., Hwang, S.J., Yang, E., Choi, S.: Neural complexity measures. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 9713–9724 (2020)

[6] Zhang, C., Samy, B., Moritz, H., Benjamin, R., Oriol, V.: Understanding deep learning (still) requires rethinking generalization. Communications of the ACM 64(3), 107–115 (2021). https://doi.org/10.1145/3446776

[7] Hossin, M., Sulaiman, M.N.: A review on evaluation metrics for data classification evaluations. International Journal of Data Mining and
A classification performance evaluation measure considering data separability

Knowledge Management Process 5(2), 1–11 (2015). https://doi.org/10.5121/ijdkp.2015.5201

[8] Yu, S., Li, X., Feng, Y., Zhang, X., Chen, S.: An instance-oriented performance measure for classification. Information Sciences 580, 598–619 (2021). https://doi.org/10.1016/j.ins.2021.08.094

[9] Fernández, A., García, M.S., and Galar, Prati, R.C., Krawczyk, B., Herrera, F.: Data intrinsic characteristics. In: Learning from Imbalanced Data Sets, pp. 253–277. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98074-4_10

[10] Bello, M., Nápoles, G., Vanhoof, K., Bello, R.: Data quality measures based on granular computing for multi-label classification. Information Sciences 560, 51–67 (2021). https://doi.org/10.1016/j.ins.2021.01.027

[11] Li, C., Wang, B.: Fisher Linear Discriminant Analysis. CCIS Northeastern University (2014)

[12] Hossin, M., Sulaiman, M.N.: A framework for dynamic classifier selection oriented by the classification problem difficulty. Pattern Recognition 76(1), 175–190 (2018). https://doi.org/10.1016/j.patcog.2017.10.038

[13] Ho, T.K., Basu, M.: Complexity measures of supervised classification problems. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(3), 289–300 (2002). https://doi.org/10.1109/34.990132

[14] Lorena, A.C., García, L.P.F., Lehmann, J., Souto, M.C.P., Ho, T.K.: How complex is your classification problem?: A survey on measuring classification complexity. ACM Computing Surveys 52(5), 1–34 (2019). https://doi.org/10.1145/3347711

[15] Ferraro, M.B., Giordani, P.: A review and proposal of (fuzzy) clustering for nonlinearly separable data. International Journal of Approximate Reasoning 115(C), 13–31 (2019). https://doi.org/10.1016/j.ijar.2019.09.004

[16] Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the Surprising Behavior of Distance Metrics in High Dimensional Space. In: Van den Bussche, J., Vianu, V. (eds) Database Theory — ICDT 2001. ICDT 2001. Lecture Notes in Computer Science, vol 1973. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-44503-X_27 (2001)

[17] Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley Interscience, USA (2006)

[18] Madiman, M., Harrison, M., Kontoyiannis, I.: Minimum description
A classification performance evaluation measure considering data separability

length versus maximum likelihood in lossy data compression. In: International Symposium on Information Theory, 2004. IEEE, Chicago. https://doi.org/10.1109/ISIT.2004.1365499 (2004)

[19] Ma, Y., Derksen, H., Hong, W.: Segmentation of multivariate mixed data via lossy data coding and compression. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(9), 1546–1562 (2007). https://doi.org/10.1109/TPAMI.2007.1085

[20] Guan, S., Loew, M.: A novel intrinsic measure of data separability. In: Applied Intelligence, (2022). https://doi.org/10.1007/s10489-022-03395-6

[21] Shannon, C.E.: A mathematical theory of communication. The Bell System Technical Journal 27(3), 379–423 (1948). https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

[22] Macdonald, J., Wäldchen, S., Hauch, S., Kutyniok, G.: A rate-distortion framework for explaining neural network decisions. Statistics (2019). https://doi.org/10.48550/arXiv.1905.11092

[23] Wu, Z., Baek, C., You, C., Ma, Y.: Incremental learning via rate reduction. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Nashville. https://doi.org/10.1109/CVPR46437.2021.00118 (2021)

[24] Elizondo, D.A., Birkenhead, R., Gamez, M., Garcia, N., Alfaro, E.: Linear separability and classification complexity 39(9), 7796–7807 (2012). https://doi.org/10.1016/j.eswa.2012.01.090

[25] Yu, Y., Chan, K.H.R., You, C., Song, C., Ma, Y.: Learning diverse and discriminative representations via the principle of maximal coding rate reduction (2020). https://doi.org/10.48550/arXiv.2006.08558

[26] Luor, D.C.: A comparative assessment of data standardization on support vector machine for classification problems. Intelligent Data Analysis 19(3), 529–546 (2015). https://doi.org/10.3233/IDA-150730

[27] Mishra, A.K.: Separability indices and their use in radar signal based target recognition. IEICE Electronics Express 6(14), 1000–1005 (2009). https://doi.org/10.1587/elex.6.1000

[28] Lichman, M.e.a.: Uci machine learning reposit. https://archive.ics.uci.edu/ml/datasets.php (2013)

[29] Sizhe, C., Haipeng, W., Feng, X., Yaqiu, J.: Target classification using the deep convolutional networks for sar images. IEEE Transactions on
A classification performance evaluation measure considering data separability

Geoscience and Remote Sensing 54(8), 4806–4817 (2016). https://doi.org/10.1109/TGRS.2016.2551720

[30] Arbelaitz, O., Gurrutxaga, I., Muguerza, J., M.Pérez, J., Perona, I.: An extensive comparative study of cluster validity indices. Pattern Recognition 46(1), 243–256 (2013). https://doi.org/10.1016/j.patcog.2012.07.021

[31] Dong, N.T., Khosla, M.: Revisiting feature selection with data complexity. In: 2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE). IEEE, Cincinnati, OH, USA. https://doi.org/10.1109/BIBE50027.2020.00042 (2020)

[32] Zhu, Y., Sun, J., Wang, M., Yao, R., Zhang, Y.: Feature separability based on the distance matrix. In: 2017 International Conference on Orange Technologies (ICOT). IEEE, Singapore. https://doi.org/10.1109/ICOT.2017.8336087 (2017)

[33] Schilling, A., Maier, A., Gerum, R., Metzner, C., Krauss, P.: Quantifying the separability of data classes in neural networks. Neural networks : the official journal of the International Neural Network Society 139, 278–293 (2021). https://doi.org/10.1016/j.neunet.2021.03.035

[34] Charte, D., Charte, F., Herrera, F.: Reducing data complexity using autoencoders with class-informed loss functions. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021). https://doi.org/10.1109/TPAMI.2021.3127698