Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi

Afyon Kocatepe University Journal of Science and Engineering

AKÜ FEMÜBİD **23** (2023) 065101 (1445-1457) AKU J. Sci. Eng. **23** (2023) 065101 (1445-1457)

DOI: 10.35414/akufemubid.1325843

Araştırma Makalesi / Research Article

An Empirical Study on the Performance of the Distance Metrics

Fatih AYDIN[1*]

[1] *Department of Computer Engineering, Faculty of Engineering, Balıkesir University, 10145 Balıkesir, Türkiye.*

**Corresponding author e-mail: fatih.aydin@balikesir.edu.tr ORCID ID: https://orcid.org/0000-0001-9679-0403*

Geliş Tarihi: 11 Temmuz 2023 ; Kabul Tarihi: 7 Aralık 2023

Keywords

Machine learning; Metric learning; Information gain; No free lunch theorems; K-nearest neighbors.

Abstract

Metrics are used to measure the distance, similarity, or dissimilarity between two points in a metric space. Metric learning algorithms perform the finding task of data points that are closest or furthest to a query point in m-dimensional metric space. Some metrics take into account the assumption that the whole dimensions are of equal importance, and vice versa. However, this assumption does not incorporate a number of real-world problems that classification algorithms tackle. In this research, the existing information gain, the information gain ratio, and some well-known conventional metrics have been compared by each other. The 1-Nearest Neighbor algorithm taking these metrics as its meta-parameter has been applied to forty-nine benchmark datasets. Only the accuracy rate criterion has been employed in order to quantify the performance of the metrics. The experimental results show that each metric is successful on datasets corresponding to its own domain. In other words, each metric is favorable on datasets overlapping its own assumption. In addition, there also exists incompleteness in classification tasks for metrics just like there is for learning algorithms.

Uzaklık Metriklerinin Performansı Üzerine Ampirik Bir Çalışma

Anahtar kelimeler

Makine öğrenmesi; Metrik öğrenme; Bilgi kazancı; No free lunch teoremleri; K-en yakın komşular

Öz

Metrik, bir metrik uzayda iki nokta arasındaki mesafeyi, benzerliği veya farklılığı ölçmek için kullanılır. Metrik öğrenme algoritmaları, m boyutlu metrik uzayda bir sorgulama noktasına en yakın veya en uzak olan veri noktalarını bulma görevini gerçekleştirir. Bazı metrikler, tüm boyutların eşit öneme sahip olduğu varsayımını dikkate alır ve bunun tersi de geçerlidir. Ancak bu varsayım, sınıflandırma algoritmalarının üstesinden geldiği bazı gerçek dünya problemleriyle örtüşmez. Bu araştırmada; mevcut bilgi kazanımı, bilgi kazanım oranı ve bazı iyi bilinen konvansiyonel metrikler birbirleri ile karşılaştırılmıştır. Bu metrikleri meta parametresi olarak alan 1-En Yakın Komşular algoritması 49 veri kümesine uygulanmıştır. Metriklerin performansını ölçmek için sadece doğruluk oranı ölçütü kullanılmıştır. Deneysel sonuçlar, her metriğin kendi domainine karşılık gelen veri setlerinde başarılı olduğunu göstermektedir. Başka bir deyişle; her metrik, kendi varsayımıyla örtüşen veri kümelerinin lehinedir. Ayrıca öğrenme algoritmalarında olduğu gibi metrikler için de sınıflandırma görevlerinde eksiklikler mevcuttur.

1. Introduction

Metric learning is a topic of research dealing with distance, similarity, or other criterion (e.g., any optimal metric regardless of the location) between data points. It is the key to the success of many machine learning algorithms, e.g., the k-nearest neighbor algorithm (k-NN) in classification, the k-means algorithm in clustering, and the principal component analysis technique in dimensionality reduction.

The metric learning algorithms can have high performance on some real-world troubles, and every algorithm has some intrinsic characteristics:, the form of metric, learning paradigm, optimality of the solution, scalability, dimensionality reduction and so (Bellet et al. 2013). In terms of learning paradigms, there are three paradigms: fully supervised, unsupervised, and semi-supervised. Fully supervised is a learning paradigm in which learning algorithms have labeled examples. Unsupervised is a learning paradigm in

which learning algorithms do not have the labels of individual training examples. Semi-supervised is a learning paradigm in which only a subset of training examples is given with labels (Parmar et al. 2021).

The type of learned metric is a crucial choice. One can define two leading metric families: local or global linear metrics and local or global nonlinear metrics (Bellet et al. 2015). The expressive power of linear metrics (e.g., Mahalanobis distance) is restricted, but they are simpler to optimize and less prone to overfitting. That is, they mostly cause convex formulations and thereby, non-local optimality of the solution. Nonlinear metrics (e.g., the χ^2 distance) generally lead to non-convex formulations (subject to locality) and overfitting, but they are able to also acquire non-linearity in the data. Local metrics are a type of metric where linear and nonlinear local metrics are learned together to overcome complex problems, i.e., heterogeneous data. They are more inclined to overfit as opposed to non-local methods because the number of parameters they learn can be very large. The optimality of the solution denotes the generalization capability of the algorithm to discover the parameters of the metric that adequately fulfill the desirable criterion. The solution is guaranteed to be the global optimum for convex problems. Otherwise, the solution may exclusively be stuck in a local optimum for non-convex problems. Scalability with dimensionality refers to that metric learning algorithms should also deal appropriately with the dimensionality of the data (Bellet et al. 2013, 2015, Peng et al. 2018).

In the machine learning area, there are many learning approaches applied to classification and clustering problems, and lazy learning is one of them. Lazy learning makes up local models, as opposed to other learning approaches. The forming of local models is a severe deficiency of lazy learning. Despite that deficiency, however, lazy learning is relatively successful over real-world problems. The k-NN algorithm is a well-known lazy learning algorithm. The k-NN algorithm relies on the finding of *k*-data points that are closest to a query point in *m*-dimensional metric space (Beyer et al. 1999) and has low bias and high variance (Manning and Raghavan 2009). First of all, a distance metric is necessary to measure "closeness" between two points in a metric space (Han and Kamber 2006). In this respect, there are many distance metrics: Euclidean distance, the City Block distance, Chebyshev distance, Minkowski distance, and so on. As well as finding the optimum value of *k*, selecting a proper metric is vital to the classification with high accuracy (Hechenbichler and Schliep 2004).

The main contributions and findings of this study are as follows:

- There is at least one dataset where each metric delivers the highest result.
- Considering all the possible datasets, the average performance of the metrics approaches each other.
- As for the real-world datasets, the average performance of the metrics changes depending on the strength of their assumptions.

This paper is structured as follows: Section 2 thoroughly explains how metric learning algorithms advance state-of-the-art. In Section 3, the preliminaries required for the essence of the paper are presented. The subsections such as the metric axioms and no free lunch theorems have been subsumed by Section 4. In Section 5, the experimental procedure is expressed in detail. The comparative results with some distance metrics and empirical results are presented in Section 6. Finally, Section 7 terminates with the conclusions.

2. Works on Metric Learning

The first work regarding metric learning begins with Fix and Hodges' paper entitled "Discriminatory Analysis - Nonparametric Discrimination: Consistency Properties" (Fix and Hodges 1951). They tried to discover a rule hinged on the nearest neighbors' idea. Nilsson proposed the use of the nearest neighbor rule in pattern recognition problems (Nilsson 1965). Cover and Hart used a metric to detect the nearest neighbors and showed that the convergence of the nearest neighbor to any point is independent of the metric in Euclidean n-space (Cover and Hart 1967). Fukunaga and Hostetler showed that the performance of a k-NN

classifier is dependent on the choice of metric (Fukunaga and Hostetler 1973). Brown and Koplowitz unveiled that a proper choice of metric is significant. Furthermore, they showed that the performance of a classifier could be increased through the weighting of distance measurements (Brown and Koplowitz 1979). Short and Fukunaga deal with the trouble regarding the selecting of the best distance measurement by decreasing the difference between the nearest neighbor classification and the asymptotic nearest neighbor errors (Short and Fukunaga 1981).

Jia et al. suggested a novel distance metric for nominal data hinged on the properties of nominal values. Their method relies on the fact that the distance between two values belonging to a feature is decided by both the frequency probabilities of these two values and those of other features (Jia et al. 2016). Gu et al. proposed a novel distance metric that consists of the standard Euclidean distance and a directional divergence derived from the cosine similarity, in order to cope with high-dimensional problems (Gu et al. 2017). Siyu et al. developed a new algorithm named Multi-Instance Transfer Metric Learning, which tries to build a bridge between the distributions of diverse domains by employing the bag weighting strategy. Thus, they tried to overcome the inconsistency between a source domain and a target domain drawn from dissimilar distributions (Jiang et al. 2018). Utkin and Ryabinin presented a metric learning algorithm hinged on the Deep Forest offered by Zhou and Feng (Zhou and Feng 2019). The major idea of the algorithm is to appoint the weights to decision trees in the Random Forest so as to decrease openness between examples from the similar class and to augment them between examples from distinct classes (Utkin and Ryabinin 2019). Zabihzadeh et al. proposed a new strategy for metrics in latent space. Their algorithm tries to find out an optimal pairing from the feature space to a latent space that decreases the gap between the same examples and also augments the distance between distinct ones (Zabihzadeh et al. 2019). Zhang et al. proposed a novel parameterization technique for comprising the squared Mahalanobis distance into the Gaussian RBF kernel so as to form a new measure for common learning distance metric and kernel classifier (Zhang et al. 2019).

Additional to the abovementioned works, there exist some studies done by weighting dimensions (or features) according to their importance levels, for instance, the information gain (Hall 1999, Taneja et al. 2014, Duneja and Puyalnithi 2017), the relevance of features in similarity computations (Aha 1998), Pearson correlation coefficient ranking (Guyon and Elisseeff 2003, Grabczewski and Jankowski 2006), Fisher coefficient (Grabczewski and Jankowski 2005), Chi-squared coefficient (Vivencio et al. 2007) and numerous studies based on decision trees and probability distribution distance (Jankowski and Usowicz 2011).

The abovementioned metrics take into account the assumption that the whole dimensions are of the same importance. However, this circumstance does not approximate the intrinsic structure of a set of real-world troubles that algorithms like the k-NN algorithm tackle (Taneja et al. 2014, Duneja and Puyalnithi 2017). Hence, every dimension should be handled with unequal importance by a specific function (Aydın 2022).

3. Preliminaries

In this section, we recapture some essentials in preparation for something fuller.

3.1 Metric axioms

A metric space is a set defined by a global space notion between its elements. A metric space has to fulfill the essential properties as follows (Rudin 1976, Munkres 2017): (P1) $d(u,v) \geq 0$, (P2) $d(u,v) = 0 \iff u = v$, (P3) $d(u,v) = d(u,v)$, and (P4) $d(u,v) \leq d(u,t) + d(v,t)$, where the metric d on a set M is a function $d: M \times M \to \mathbb{R}$ such that for $\forall\, u,v,t \in M$. The principles in (P1), (P2), (P3), and (P4) are so-called non-negativity, identity of indiscernibles, symmetry of distances, and the triangle inequality, respectively for a metric space. For instance, the City Block metric $d: \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$

on $\mathbb{R}^n$ is defined by $d(p,q) = \sum_{i=1}^{n} |p_i - q_i|$.

3.2 No free lunch theorems

No Free Lunch (NFL) theorems state that there is no universal method that has the best performance overall the possible problems (Wolpert 1996, Wolpert and Macready 1997). Hence, there are datasets on which every algorithm is both successful and unsuccessful since there is no best algorithm. In other words, NFL theorems specifies that: the average performances of each method over all the real-world troubles are the same (Wolpert and Macready 2005). Wolpert and Macready present two main NFL theorems: the first one, which is about invariant objective functions while the search is in progress, and the latter is about objective functions that may alter. More formally, let an optimization problem f be represented as a mapping in the set $F = \{f: X \mapsto Y\}$, where the search space X is finite, and the space of possible cost values Y is finite. Besides, let the performances of any two algorithms a_1 and a_2 iterated *m* times on a cost function be conditional probabilities measured with $P(d_m^y | f, m, a_1)$ and $P(d_m^y | f, m, a_2)$, respectively. Then, $P(d_m^y | f, m, a)$ is independent of *a* when averaged over all cost functions, as shown in Equation (1). A primary corollary of this result is that the precise way to match the sample to a performance measure is insignificant, and this is Wolpert and Macready's first theorem.

$$\sum_f P(d_m^y | f, m, a_1) = \sum_f P(d_m^y | f, m, a_2) \qquad (1)$$

where d_m^y denotes the ordered set of size *m* of cost values $y \in Y$ corresponding to $x \in X$. This theorem clearly shows that what a method boosts in performance for a problem class and is unavoidably balanced through its performance on the rest issues; this is the only way for all algorithms to have the same average performance (Wolpert and Macready 1997). We do not address Wolpert and Macready's second theorem because the second theorem is connected to time-varying objective functions. NFL theorems have serious implications for learning algorithms. In particular, Wolpert and Macready's first theorem within these theorems is direct regarding this paper. In the context of distance metrics and metric learning, there is no strategy that outperforms others in all problems. To put it more explicitly, universal distance metrics or metric learning algorithms are impossible. Then, we can remark on each distance metric is only successful over its own domain set. Moreover, all discussions to do are to revolve around this remarkable conclusion.

4. Experimental Process

We have used 49 datasets to measure empirically the competitiveness and performance of the distance metrics. The descriptive information regarding the datasets is shown in Table 1. We tested all the metrics on 49 datasets from The UCI Machine Learning Repository (Int. Ref. 1), mlbench (Int. Ref. 2), and MATLAB Sample Data Sets (Int. Ref. 3).

The experiments done have been delimited by the classification problems. The k-Nearest Neighbors (k-NN) algorithm has been employed as a classifier. The distance metrics used in the experiments are: the *baseline* metric, the *IG* (d_{IG}) (Aydın 2022), the *IGR* (d_{IGR}), the *City Block* (the Manhattan distance), the *Chebyshev*, the *correlation* (Székely et al. 2007), the *cosine* (Korenius et al. 2007), the *Euclidean*, the *Hamming* (Norouzi et al. 2012), the *Jaccard* (Hancock 2004), the *Mahalanobis* (De Maesschalck et al. 2000), and the *Spearman* (Monjardet 1998). A 10-fold cross-validation technique has been run to assess the k-NN algorithm on the datasets. In the experiments, the parameter *k* of the k-NN algorithm has been chosen as 1. Thus, to compare the performances of the metrics independently from the other parameters of the k-NN algorithm; we set the values of the remaining parameters likewise. The k-NN algorithm has been operated on each dataset five times. Thereby, more accurate results have been tried to obtain by testing the distance metrics on five different cross-validated training sets derived from each dataset. The average rank values of the metrics are quantified by Spearman's rank correlation coefficient (Spearman 1904).

Table 1. The benchmark datasets used in the experiments (The imbalance ratio specifies the ratio of the number of samples in the majority and minority classes).

#	Dataset	Sample size	#Feature	#Class	Imbalance ratio
1	Arrhythmia	452	280	13	122.50
2	Auditrisk	776	27	2	1.54
3	Avila	20867	11	12	857.20
4	BanknoteAuthentication	1372	5	2	1.27
5	BloodTransfusion	748	5	2	3.20
6	BostonHousing2	506	19	92	30.00
7	BreastCancer	699	10	2	1.90
8	BreastTissue	106	10	6	1.57
9	Cardiotocography3	2126	22	3	9.40
10	Cardiotocography10	2126	22	10	10.92
11	ClimateModel	540	19	2	10.73
12	ConnectionistBench	208	61	2	1.14
13	DiabeticRetinopathy	1151	20	2	1.13
14	DNA	3186	181	3	2.16
15	Ecoli	336	8	8	71.50
16	FisherIris	150	5	3	1.00
17	FrogsMFCCs_Families	7195	23	4	65.00
18	FrogsMFCCs_Genus	7195	23	8	61.03
19	FrogsMFCCs_RecordID	7195	23	60	458.00
20	FrogsMFCCs_Species	7195	23	10	51.15
21	Glass	214	10	6	8.44
22	Haberman	306	4	2	2.78
23	HTRU2	17898	9	2	9.92
24	Ionosphere	351	35	2	1.78
25	Leaf	340	15	30	2.00
26	LetterRecognition	20000	17	26	1.10
27	LibrasMovement	360	91	15	1.00
28	LSVTvoiceRehabilitation	126	311	2	2.00
29	Madelon	2000	501	2	1.00
30	MAGICGammaTelescope	19020	11	2	1.84
31	MEU_MobileKSD	2856	72	56	1.00
32	OpticalRecognition	3823	65	10	1.03
33	Ovariancancer	216	4001	2	1.27
34	PageBlocks	5473	11	5	175.46
35	ParkinsonSpeech	1040	27	2	1.00
36	QSARBiodegradation	1055	42	2	1.96
37	Satellite	6435	37	6	2.44
38	Seeds	210	8	3	1.00
39	Sonar	208	61	2	1.14
40	Vehicle	846	19	4	1.10
41	VertebralColumn	310	7	2	2.10
42	Vowel	990	11	11	1.00
43	WallFollowingRobotNavigation2	5456	3	4	6.72
44	WallFollowingRobotNavigation4	5456	5	4	6.72
45	WallFollowingRobotNavigation24	5456	25	4	6.72
46	Winequalityred	1599	12	6	68.10
47	Winequalitywhite	4898	12	7	439.60
48	Yeast	1484	9	10	92.60
49	Zoo	101	17	7	10.25

5. Results and Discussion

In the machine learning field, any reasonable classifier is supposed to be a higher performance than a random predictor on any dataset. Thus, it can be said that the predictions of the classifiers are acceptable. In that case, a random predictor can be used as a baseline to measure the performance of classifiers. In this respect, we can measure the performances of the metrics likewise. First, we need to define a random distance metric. The random distance metric measures the distance between two points as follows: $d(p,q) = \sum_{i=1}^{n} |p_i - q_i|^{3X_i}$ where $X_i \sim$ U([0, 1]). Namely, a vector X consists of random variables uniformly distributed on [0, 1]. Now, let us analyze the

average performance of the metrics used in the experiments using the random distance metric. We have compared 12 distance metrics on 49 benchmark datasets. The task was posed as a classification problem. Besides, all the detailed experimental results are shown in Table 2.

The average classification accuracy rates of the *Baseline*, the *IG*, the *City Block*, *cosine*, *Euclidean*, and the *Mahalanobis* metrics have been found to be 79.6705%, 81.9371%, 82.3231%, 80.1956%, 81.3542%, and 79.9944%, respectively. The *City Block*, the *IG*, and the *Euclidean* distance metrics have the first three highest average classification accuracy rates on the benchmark datasets in turn.

Table 2. The average classification accuracy rates with their standard deviations of the k-NN algorithm in terms of the metrics.

#	*Baseline*	IG	IGR	*City Block*	*Chebyshev*	*correlation*	*cosine*	*Euclidean*	*Hamming*	*Jaccard*	*Mahalanobis*	*Spearman*
1	56.90±0.7	57.08±0.1	54.20±0.0	56.86±0.3	56.33±0.2	57.21±0.2	57.26±0.2	57.43±0.4	54.51±0.2	55.58±0.1	—	**57.52±0.3**
2	97.50±0.1	94.41±0.3	60.70±0.0	**97.47±0.2**	96.34±0.2	96.91±0.2	96.60±0.2	97.04±0.1	93.09±0.2	93.25±0.1	—	96.24±0.3
3	85.09±0.1	99.83±0.0	99.87±0.0	87.39±0.0	73.17±0.1	74.33±0.1	78.15±0.2	79.54±0.1	**99.89±0.0**	**99.89±0.0**	76.04±0.1	65.65±0.1
4	99.71±0.1	97.94±0.2	86.34±0.4	99.93±0.0	**100±0.0**	93.43±0.3	99.93±0.0	99.93±0.0	53.88±0.5	53.88±0.5	**100±0.0**	59.56±0.1
5	72.03±0.9	71.90±0.9	71.82±0.9	72.27±0.4	73.48±0.6	73.31±0.3	73.85±0.7	72.21±0.5	75.77±0.4	75.77±0.4	—	**76.20±0.0**
6	81.14±0.9	**96.08±0.1**	**96.08±0.1**	95.17±0.1	93.75±0.2	87.35±0.1	87.98±0.2	94.94±0.1	84.38±0.3	84.38±0.3	82.05±0.6	52.05±0.4
7	95.67±0.4	96.36±0.2	**96.48±0.3**	96.02±0.2	92.64±0.4	90.38±0.6	90.70±0.4	95.36±0.2	94.02±0.2	94.02±0.2	94.13±0.6	89.27±0.4
8	60.37±1.3	64.71±1.3	**69.81±1.3**	60.00±2.0	55.28±2.4	56.03±1.0	56.60±0.8	56.60±2.0	31.69±0.8	31.69±0.8	63.01±1.5	32.64±0.8
9	89.02±0.5	**91.36±0.2**	90.92±0.2	90.83±0.3	89.89±0.1	88.46±0.1	88.47±0.2	90.23±0.3	90.79±0.1	90.37±0.2	—	87.25±0.2
10	70.28±0.5	74.02±0.4	**75.16±0.3**	74.51±0.3	71.39±0.3	71.89±0.6	71.59±0.5	73.80±0.4	70.89±0.4	71.16±0.5	—	67.69±0.4
11	88.51±0.7	**91.40±0.4**	90.70±0.8	88.59±0.1	88.18±0.4	87.33±0.4	86.92±0.5	88.37±0.2	16.81±0.1	16.81±0.1	88.81±0.2	86.07±0.4
12	84.51±1.5	83.46±1.3	77.50±1.0	83.94±0.8	78.84±0.8	85.09±0.6	82.88±0.4	82.11±0.7	49.23±2.1	49.23±2.1	78.46±1.6	**85.76±0.9**
13	62.29±0.9	62.13±0.5	61.85±0.7	61.66±0.5	65.03±0.3	**66.34±0.5**	65.80±0.5	62.17±0.2	61.59±0.3	61.77±0.4	63.47±0.7	61.30±0.4
14	73.72±0.3	**73.75±0.2**	**73.75±0.2**	73.45±0.3	28.88±0.1	73.72±0.3	73.53±0.3	73.45±0.3	73.45±0.3	73.49±0.4	52.48±0.3	73.70±0.3
15	76.13±1.3	**81.78±0.9**	71.42±1.0	80.83±0.6	80.41±0.5	80.59±0.3	79.64±0.4	81.42±0.6	56.13±0.4	56.13±0.4	—	76.48±0.6
16	94.26±1.2	94.40±0.8	90.93±0.3	95.46±0.3	**96.80±0.3**	93.60±0.3	96.13±0.3	96.00±0.0	80.00±1.4	80.00±1.4	90.66±0.9	66.66±0.0
17	98.55±0.1	98.93±0.0	98.52±0.0	98.88±0.0	98.68±0.0	**99.04±0.0**	99.03±0.0	98.97±0.0	57.67±0.0	57.67±0.0	97.87±0.0	98.03±0.0
18	98.00±0.1	98.56±0.0	98.19±0.0	98.93±0.0	98.65±0.0	**99.04±0.0**	99.02±0.0	98.94±0.0	57.67±0.0	57.67±0.0	97.87±0.0	98.01±0.0
19	74.52±0.4	85.28±0.2	84.42±0.2	86.86±0.0	85.41±0.0	**88.14±0.1**	87.62±0.1	87.24±0.0	0.82±0.0	0.82±0.0	78.32±0.1	77.96±0.0
20	97.79±0.1	98.61±0.0	98.12±0.0	98.74±0.0	98.42±0.0	**98.87±0.0**	98.86±0.0	98.77±0.0	9.77±0.0	9.77±0.0	97.46±0.0	97.92±0.0
21	69.34±2.7	**74.48±0.8**	73.17±1.2	73.64±2.1	72.05±1.7	71.02±1.3	72.52±1.2	74.11±1.4	47.66±1.0	49.25±0.8	66.63±0.8	51.02±0.5
22	65.22±1.1	66.53±1.3	65.94±0.6	64.90±0.5	65.68±0.5	65.88±0.4	66.92±0.1	**67.84±0.8**	64.90±0.4	63.92±0.7	63.98±0.8	67.18±0.4
23	96.29±0.1	96.75±0.0	96.82±0.0	96.38±0.0	96.07±0.1	96.08±0.1	96.12±0.1	96.24±0.0	91.44±0.0	91.44±0.0	**97.23±0.0**	93.99±0.0
24	89.17±0.8	**92.36±0.4**	64.10±0.0	90.54±0.5	88.60±0.4	88.20±0.3	88.60±0.7	86.55±0.2	42.27±0.2	43.76±0.2	—	87.40±0.1
25	61.47±1.1	60.41±0.7	68.35±1.0	65.23±0.3	54.29±0.5	61.52±0.3	61.88±0.4	59.70±0.7	6.23±0.5	6.23±0.5	**78.94±0.4**	44.58±0.9
26	94.96±0.1	95.06±0.1	92.32±0.0	95.35±0.1	80.37±0.1	95.08±0.1	95.76±0.1	**95.87±0.1**	87.70±0.1	87.67±0.1	94.52±0.0	90.72±0.1
27	84.05±1.7	85.22±0.8	82.50±0.6	85.72±0.5	84.33±0.5	**86.61±0.8**	84.88±0.8	86.22±0.6	42.66±0.8	42.66±0.8	47.05±1.7	79.94±0.8
28	56.50±2.2	69.20±2.5	65.39±3.9	57.61±3.1	56.03±1.9	63.17±1.6	62.53±1.7	52.69±1.9	36.03±0.6	36.03±0.6	—	**70.00±2.2**
29	55.52±1.1	52.59±0.4	52.48±0.4	65.98±0.6	58.17±0.2	**66.10±0.5**	64.44±0.5	64.34±0.3	50.87±0.4	50.87±0.4	51.42±0.4	59.10±0.4
30	75.62±0.3	78.15±0.1	71.27±0.1	78.61±0.1	77.72±0.1	74.77±0.1	74.72±0.1	78.38±0.1	65.63±0.0	65.62±0.0	**81.83±0.1**	67.53±0.0
31	35.96±0.9	44.88±0.5	**66.66±0.2**	55.88±1.8	34.74±2.7	41.12±2.4	41.04±2.4	45.49±2.1	23.24±3.0	23.24±3.0	—	45.79±2.4
32	94.83±0.2	97.58±0.1	10.15±0.0	98.21±0.0	97.89±0.0	**98.62±0.0**	98.58±0.0	98.56±0.0	88.56±0.1	84.19±0.1	—	98.03±0.1
33	91.38±1.2	**93.70±0.7**	**93.70±0.4**	92.12±0.6	81.29±1.0	92.77±0.4	92.87±0.4	91.20±0.8	57.87±0.0	57.87±0.0	—	89.62±0.9
34	94.86±0.1	95.40±0.1	95.26±0.0	95.73±0.1	95.70±0.1	**96.68±0.1**	96.64±0.1	95.71±0.1	92.79±0.1	92.79±0.1	96.37±0.1	93.64±0.1
35	59.44±1.5	56.73±0.4	56.01±0.3	**65.13±0.9**	58.80±0.4	57.21±0.4	56.76±0.2	63.07±0.7	51.36±0.7	51.17±0.8	64.21±0.9	57.00±0.9
36	80.58±0.5	80.09±0.7	79.62±0.6	80.62±0.3	74.63±0.5	81.27±0.5	81.25±0.3	79.39±0.5	75.58±0.6	76.72±0.4	**84.37±0.3**	81.08±0.4
37	88.71±0.3	90.66±0.2	88.80±0.1	**90.77±0.1**	86.80±0.1	78.63±0.1	79.88±0.2	90.66±0.1	84.75±0.1	84.75±0.1	67.26±0.2	71.39±0.2
38	89.71±1.2	92.19±0.8	89.61±0.4	90.66±0.5	89.61±0.8	**93.52±0.5**	92.38±0.8	90.47±0.9	56.95±1.1	56.95±1.1	91.90±0.6	50.00±0.0
39	84.13±1.1	84.23±1.2	77.40±1.5	84.23±0.6	78.94±1.1	85.28±1.2	83.55±0.4	82.59±0.5	49.80±1.7	49.80±1.7	79.71±2.0	**86.05±0.8**
40	62.81±1.0	66.76±0.9	68.20±0.4	67.96±0.4	60.14±0.7	68.46±0.5	67.21±0.5	65.36±0.5	63.14±0.2	63.21±0.2	**77.25±0.4**	53.97±0.6
41	75.87±2.3	77.03±0.8	73.03±1.1	82.64±1.0	82.83±1.0	76.12±1.0	79.54±1.2	**83.35±1.1**	65.35±0.5	65.35±0.5	—	68.12±0.1
42	98.20±0.2	98.06±0.2	95.65±0.4	98.82±0.1	98.66±0.2	98.04±0.4	98.42±0.4	**98.84±0.2**	13.11±0.4	13.21±0.4	98.32±0.4	68.98±0.4
43	96.19±0.2	98.10±0.0	98.39±0.1	98.80±0.0	98.76±0.0	26.03±0.2	65.36±0.2	98.82±0.1	66.96±0.2	66.96±0.2	**98.97±0.0**	23.60±0.0
44	93.04±0.3	94.78±0.1	93.71±0.1	97.29±0.1	97.17±0.0	84.52±0.2	95.80±0.1	97.30±0.1	65.17±0.3	65.17±0.3	**97.51±0.0**	41.92±0.1
45	90.74±0.2	**95.00±0.1**	94.53±0.1	92.71±0.1	82.19±0.2	88.38±0.1	88.73±0.1	88.59±0.1	80.75±0.1	80.75±0.1	87.18±0.1	91.69±0.2
46	59.33±0.9	62.52±0.2	61.70±0.1	60.46±0.4	58.88±0.2	61.76±0.6	61.76±0.6	60.08±0.5	57.73±0.3	57.99±0.4	**64.05±0.4**	17.08±0.2
47	60.44±0.4	61.82±0.2	61.53±0.3	60.21±0.3	58.68±0.3	59.72±0.3	60.09±0.3	59.68±0.4	56.59±0.3	56.57±0.3	**65.13±0.4**	4.74±0.1
48	45.86±0.8	44.44±0.5	41.42±0.4	**53.59±0.4**	52.16±0.2	53.45±0.4	53.57±0.5	52.53±0.4	34.74±0.3	34.60±0.3	52.50±0.4	43.20±0.5
49	97.42±0.5	**98.01±0.0**	96.03±0.0	96.03±0.0	79.00±1.0	96.03±0.0	97.02±0.0	**98.01±0.0**	96.03±0.0	96.03±0.0	92.67±1.3	95.04±0.0
Avg.	79.67±16	81.93±16	77.97±18	**82.32±15**	77.99±18	78.92±17	80.19±15	81.35±16	59.75±25	59.76±25	79.99±16	69.35±22

The average rank values of the metrics are shown in Figure 1(b). In light of the result, the *IG*, the *IGR*, the *City Block*, the *correlation*, the *cosine*, the *Euclidean*, and the *Mahalanobis* metrics have obtained higher average rank values than the *Baseline* distance metric on the benchmark datasets. The average rank value of the *Baseline* distance metric has been found to be 6.2857. The average rank values of the abovementioned metrics have been found to be 8.3878, 6.6531, 8.7347, 7.6837, 7.8878, 8.1531, and 7.0946, respectively. The *City Block*, the *IG*, and the *Euclidean* distance metrics have the first three

highest rank values on the benchmark datasets in turn.

Concerning the number of datasets on which the metrics have the highest classification accuracy rates, the experimental results are shown in Figure 1(c). As seen clearly from those results; the first three metrics are the *IG*, the *correlation*, and the *Mahalanobis*.

Regarding the number of datasets on which the metrics have higher classification accuracy rates than the *Baseline*, the experimental results are shown in Figure 1(d). In light of the results, the *City Block*, the *IG*, and the *Euclidean* distance metrics have the greatest number of datasets in turn.

As a consequence, we can state that the *IG* distance metric is a measurement system that yields acceptable and successful results on a considerable number of datasets. Moreover, the performance of the *IG* is better compared to other metrics used in the experiments, on average.

In the machine learning area, one of the important theorems is the "No Free Lunch" theorem, as well.

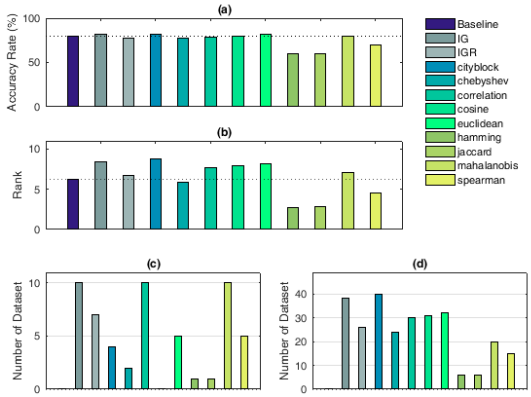

Figure 1. (a) the average classification accuracy rates of the metrics (b) the average rank values of the metrics (c) the number of the datasets on which the metrics have the highest classification accuracy rates (d) the number of the datasets on which the metrics have higher classification accuracy rates than the Baseline metric.

The NFL theorems deal with classification algorithms, search algorithms, and optimization algorithms. According to the consequences of these theorems, there is no universal approach that has the best performance on all possible datasets since the domains are partially different, in which each algorithm is successful. Furthermore, the average performances of the algorithms converge with each other as the number of benchmark datasets increases. In some works done concerning metric learning, the process of defining the different metrics is regarded as an optimization problem.

Therefore, the NFL theorem defined for optimization problems involves metric learning problems. Depending on the increment in the number of datasets, the change in the average classification accuracy rates of the metrics is shown in Figure 2. According to those results, the final average classification accuracy rates of the *Hamming* and the *Jaccard* metrics are quite different from the others, excluding the *Spearman* metric. The final average classification accuracy rates of the *Hamming* and the *Jaccard* metrics have been found to be 59.7573% and 59.7610%, respectively. The final average classification accuracy rate of the *Spearman* metric has been found to be 69.3592%. The reason for so much deviation of the average classification accuracy rates of the *Hamming* and the *Jaccard* metrics from the others is that the domains of most of the benchmark datasets are very different from those of the *Hamming* and the *Jaccard* metrics. Accordingly, as the number of datasets corresponding to the domains of the *Hamming* and the *Jaccard* metrics increases, the average classification accuracy rates of all the metrics approach each other.

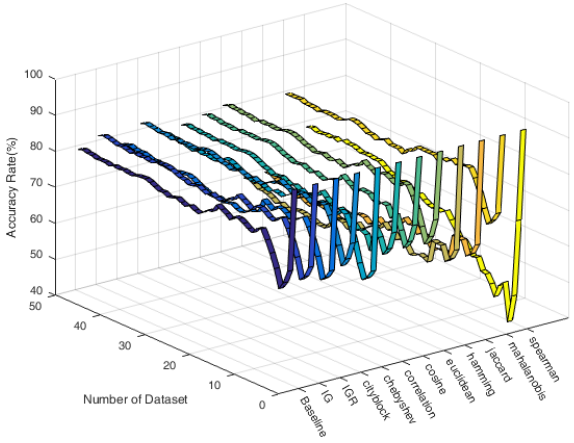

Figure 2. The change of the average classification accuracy rates of the metrics as the number of the datasets increases.

We used the Kruskal-Wallis test to measure the significance level between the average classification accuracy rates of the metrics. Thus, we can show that all the metrics exclusive of the *Hamming* and the *Jaccard* metrics come from the same population.

The Kruskal-Wallis test is a nonparametric test, i.e., a distribution-free test, and contrasts the medians of the groups of data to detect whether the samples come from the same population (or distribution). The Kruskal-Wallis test uses the ranks of the data to calculate the test statistics. Additionally, the Kruskal-Wallis test uses a chi-square statistic, and the *p*-value measures the significance level of the chi-square statistic.

Regarding the medians of the groups of the data and the ranks of the data, the statistics are shown in Figure 3. According to the experimental results, the value of *p* was measured as 2.46558e-10, and the *p*-value points out that the Kruskal-Wallis test refuses the null hypothesis that the whole data come from the similar distribution at the 1% significance level.

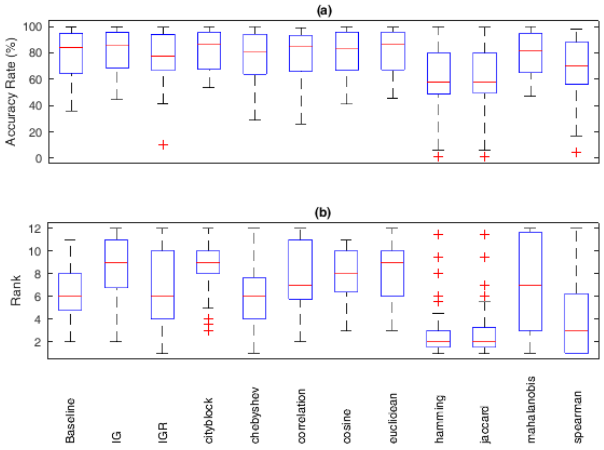

Figure 3. (a) the medians and other statistics of the groups of the data regarding the classification accuracy rates on the benchmark datasets, according to the metrics (b) according to the classification accuracy rates on the benchmark datasets of the metrics, and the statistics of their ranks. The central mark points to the median and the bottom and top edges of the box point to the first quartile and third quartile, respectively. The whiskers lengthen to the farthest data points regardless of outliers, and the outliers are marked separately by the '+' symbol.

Additionally, we conducted a follow-up test known as the Multiple Comparison Procedures to detect which data comes from a different distribution. In other words, we carried out a multiple comparison test to identify information about which data averages are crucially different or not. The Multiple Comparison Procedures are devised to ensure an upper limit on the possibility that any comparison will be mistakenly found significant.

The related test results are shown in Figure 4. In light of these results, the *Hamming* and the *Jaccard* metrics have mean ranks significantly different from all the metrics excluding the *Spearman* metric. Besides, no metrics have mean ranks significantly different from the *Spearman* metrics. As a result, we can state that the average performances of all the metrics used in the experiments are the same. Notice that most of the benchmark datasets are quite different from types of datasets on which the *Hamming* and the *Jaccard* metrics outperform.

Finally, we would like to underline that there exist datasets on which each metric is successful as well as unsuccessful.

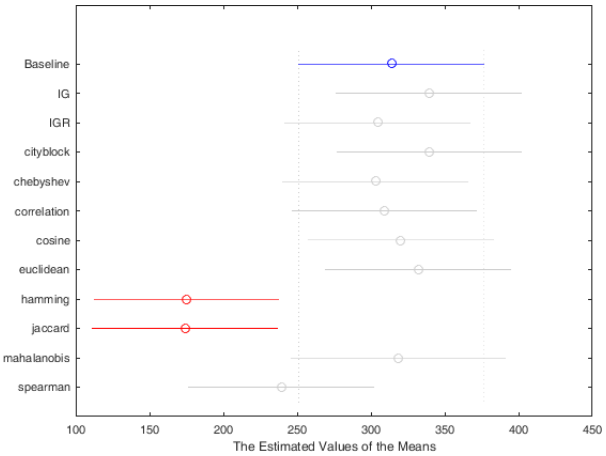

Figure 4. The estimated values of the means and comparison intervals, according to a multiple comparison test. The mean of the Baseline distance metric is highlighted, and the comparison interval is in blue. Because the comparison intervals for those of the Hamming and the Jaccard metrics do not intersect with the intervals for that of the Baseline distance metric, they are highlighted in red.

Now, let us compare the *IG* distance metric with the second-best performance metric on some datasets. According to the first two features with the highest-information gain of the *ClimateModel* dataset, the results of the two nearest neighbor

searches of the *IG* and the Mahalanobis metrics at ten random query points are shown in Figure 5.

The values of the first two features with the highest-information gain of the *ClimateModel* dataset have been found to be 0.0867 and 0.0859, respectively. Additionally, the value of the feature with the lowest information gain was found to be 0.0008. The average classification accuracy rates of the *IG* and the *Mahalanobis* metrics on the *ClimateModel* dataset have been found to be 91.4074% and 88.8148%, respectively. In other words, the *IG* distance metric has the highest average classification accuracy rate in comparison to the other metrics on the *ClimateModel* dataset. The *Mahalanobis* metric, except the other metrics based on the information gain (ratio), has the second-highest average classification accuracy rate on the *ClimateModel* dataset.

According to the results in Figure 5, at least half of the two nearest neighbor searches of the *IG* and the *Mahalanobis* metrics at ten random query points are the same. For the rest of them, the nearest neighbor searches of the *IG* metric to the query points are more distant in comparison to those of the *Mahalanobis* metric.

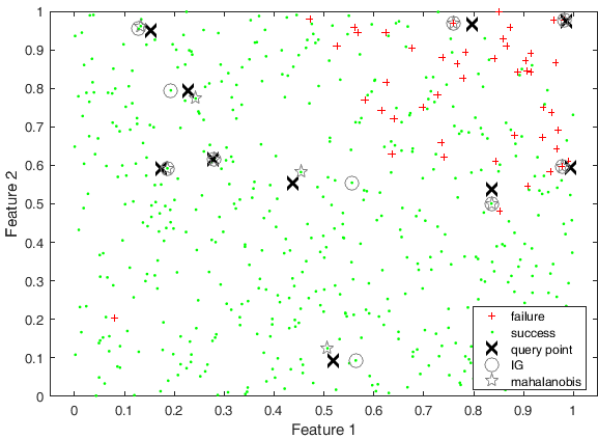

Figure 5. The results of the two nearest neighbor searches of the IG and the Mahalanobis metrics at 10 query points, according to the first two features with the highest-information gain of the ClimateModel dataset

According to the first two features with the highest information gain of the *ClimateModel* dataset, the decision boundaries of two k-NN classifiers applying to the *IG* and the *Mahalanobis* metrics are shown in Figure 6. Decision boundaries are defined as lines in which a data point is equally likely to exist in any class. Smoothing decision boundaries is the key to avoiding overfitting. According to the results in Figure 6, we can remark that the *IG* and the *Mahalanobis* metrics form smooth decision boundaries. However, the decision boundary of a k-NN classifier with the *IG* distance metric resembles the decision boundary of a decision tree. The reason is that the decision boundary of such a model is decided by the overlap of the orthogonal half-planes, representing the results of each decision. Moreover, the *IG* distance metric builds a more scattered and straight decision border compared to that of the *Mahalanobis* metric on the *ClimateModel* dataset. The decision border of a k-NN classifier with the *Mahalanobis* distance metric resembles the decision boundary of the Support Vector Machine classifier with the kernel. The reason is that its decision border is smoother compared to one with the *IG* distance metric. To put it short, the decision boundary of the k-NN classifier with the *Mahalanobis* metric is smoother in comparison with the *IG* distance metric. However, the generalization performance of the k-NN classifier with the *IG* distance metric is better in comparison to one with the *Mahalanobis* metric, for just the *ClimateModel* dataset. We want to indicate that it is not possible to comprehend the idea of how the *IG* and the *Mahalanobis* metrics work on the *ClimateModel* dataset through just two features. However, we can generally remark that the *Mahalanobis* metric measures the distance between a distribution and a point and focuses on how many standard deviations a point deviates from the mean of a distribution. As a point approaches the mean of the distribution, the *Mahalanobis* distance converges to zero; otherwise, it goes away from zero. The *IG* distance metric measures the distance between two points according to the sum of the powers of the absolute differences between the axes of two points, and the information gain method is used in the calculation of the powers. Accordingly, we can say that a dimension with high-information changes the *IG* distance more in comparison to ones with less information. Furthermore, the locations of the points around a query point are critical, as well. The reason is that the other dimensions can become crucial if the values of two points on a

dimension with the highest information are equal to each other. Moreover, a k-NN classifier with the *IG* distance metric carries out searches along all the axes of a query point, but not around it.

According to the first two features (i.e., Feature 69 and Feature 16) with the highest information gain ratio of the *MEU_MobileKSD* dataset, the decision boundaries of two k-NN classifiers applying to the *IGR* and the *City Block* metrics are shown in Figure 7. Accordingly, we can remark that the *IGR* and the *City Block* metrics form smooth decision boundaries. The values of the first two features with the highest-information gain ratio of the *MEU_MobileKSD* dataset have been found to be 0.5345 and 0.5193, respectively.

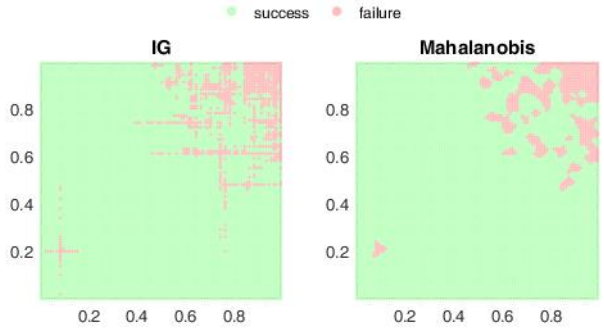

Figure 6. The decision boundaries of two different k-NN classifiers applied to the IG and the Mahalanobis metrics, according to the first two features with the highest-information gain of the ClimateModel dataset.

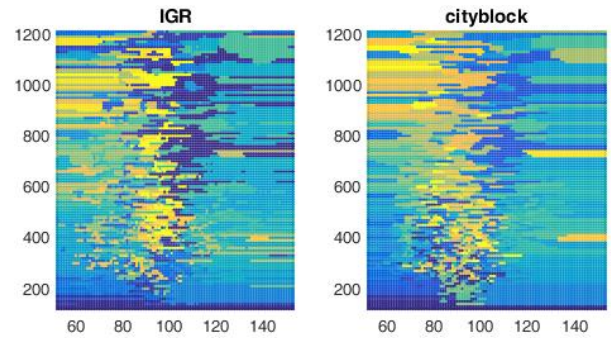

Figure 7. The decision boundaries of two k-NN classifiers applying to the IGR and the City Block metrics, according to the first two features with the highest-information gain ratio of the MEU_MobileKSD dataset (the values of the feature 16 are restricted to values between 118 and 1210).

Additionally, the value of the feature with the lowest-information gain ratio has been found to be 0.1525. The average classification accuracy rates of the *IGR* and the *City Block* metrics on the *MEU_MobileKSD* dataset have been found to be 66.6667% and 55.8894%, respectively. In other words, the *IGR* distance metric has the highest average classification accuracy rate on the *MEU_MobileKSD* dataset. The *City Block* distance metric except the other metrics based on the information gain (ratio) has the second-highest average classification accuracy rate on the *MEU_MobileKSD* dataset.

The pairwise distances between 50 instances selected randomly on the *BostonHousing2* dataset are shown in Figure 8. Four distance metrics such as the *IG*, the *IGR*, the *City Block*, and the *Euclidean* were used in the comparison. According to the results, the other three distance metrics except for the *IG* distance metric resemble each other. These four-distance metrics are the metrics giving the highest classification accuracy rates, respectively. The classification accuracy rates of these four-distance metrics are 96.0870%, 960870%, 95.1779%, and 94.9407%, respectively.

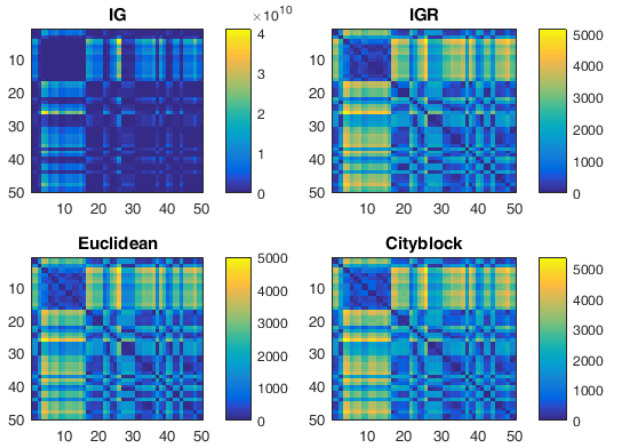

Figure 8. The pairwise distances between 50 instances that are selected randomly on the BostonHousing2 dataset, according to the IG, the IGR, the Euclidean, and the City Block distance metrics.

Putting the *IG* and the *IGR* distance metrics on both mathematical and intuitional bases, let us explain how they work. First, the information gain value of a feature tells us how far the classes diverge from each other. If so, we can say that as the information gain values of the features are large, the classes are separated so much from each other. Given the dimension with the highest information gain for a query point, the *IG* and *IGR* distance metrics move a query point much closer to the

class which is the least of the difference between the query point and the other points. However, as the number of features rises, it is required to consider the state of each feature together. For instance, in computing the distance between two points, the effect of a dimension having the lowest information gain can be much greater. The reason is that the difference between the related values of two points in a dimension having the highest information gain is less than the difference in a dimension having the lowest information gain. This situation can be a disadvantage for some datasets.

Briefly, on all the benchmark datasets, the *IG* distance metric ranks second in terms of the average classification accuracy rate. The *IGR* distance metric ranks eighth in terms of the average classification accuracy rate. Additionally, the *IG* and the *IGR* distance metrics rank second and seventh, respectively, in terms of the average rank values. Furthermore, the *IG* distance metric is a distance function to have the highest classification accuracy rate on ten datasets. The *IGR* distance metric has the highest classification accuracy rate on seven datasets. Thus, the *IG* and the *IGR* distance metrics rank first and second, respectively, in terms of the number of datasets where the metrics have the highest classification accuracy rate. Finally, the *IG* and the *IGR* distance metrics rank second and sixth, respectively, in terms of the number of datasets on which the metrics have higher classification accuracy rates than the *Baseline* metric.

6. Conclusions

In this empirical study, we have compared the performance of the distance metrics on 49 real-world datasets for the task of classification.

Each metric makes an assumption while quantifying the distance between any two points, and the measurement without the assumption is a measurement at random. Besides, there is a drawback coming along with every assumption. Therefore, we can remark that each metric is successful on datasets corresponding to its own domain. In other words, each metric is advantageous on datasets overlapping its own assumption. The experimental results verify this situation, as well. Accordingly, we can indicate that there exists incompleteness in classification tasks for metrics, too, just like there is for learning algorithms.

The information gain value of a feature gives information about how far the classes are separated from each other, and the classes split in so much that from each other as the information gain values of the features are large. Given a feature with the highest information gain for a query point, the *IG* and *IGR* distance metrics move a query point closer to the class which is the least of the difference between the query point and other points. However, as the number of features rises, it is necessary to consider each feature altogether. This approach to the aforementioned distance metrics can induce poor performance on some datasets. This naturally shows the incompleteness of these two metrics, as well.

7. References

Aha, D.W., 1998. Feature Weighting for Lazy Learning Algorithms. *In*: H. Liu and H. Motoda, eds. *Feature Extraction, Construction and Selection*. Springer, Boston, MA, 13–32.

Aydın, F., 2022. A class-driven approach to dimension embedding. *Expert Systems with Applications*, 195, 116650.

Bellet, A., Habrard, A., and Sebban, M., 2013. *A Survey on Metric Learning for Feature Vectors and Structured Data*.

Bellet, A., Habrard, A., and Sebban, M., 2015. Nonlinear and Local Metric Learning. *In*: *Metric Learning*. Springer, Cham., 33–42.

Beyer, K.S., Goldstein, J., Ramakrishnan, R., and Shaft, U., 1999. When Is '"Nearest Neighbor"' Meaningful? *In*: *ICDT '99 Proceedings of the 7th International Conference on Database Theory*. London, UK: Springer-Verlag, 217–235.

Brown, T. and Koplowitz, J., 1979. The weighted nearest neighbor rule for class dependent sample sizes (Corresp.). *IEEE Transactions on Information Theory*, **25** (5), 617–619.

Cover, T. and Hart, P., 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, **13** (1), 21–27.

Duneja, A. and Puyalnithi, T., 2017. Enhancing

Classification Accuracy of K-Nearest Neighbours Algorithm Using Gain Ratio. *International Research Journal of Engineering and Technology*, **4** (9), 1385–1388.

Fix, E. and Hodges, J.L., 1951. *Discriminatory Analysis - Nonparametric Discrimination: Consistency Properties*. Texas.

Fukunaga, K. and Hostetler, L., 1973. Optimization of k nearest neighbor density estimates. *IEEE Transactions on Information Theory*, **19** (3), 320–326.

Grabczewski, K. and Jankowski, N., 2005. Feature selection with decision tree criterion. *In*: N. Nedjah, L. de M. Mourelle, M. Vellasco, A. Abraham, and M. Köppen, eds. *Fifth International Conference on Hybrid Intelligent Systems (HIS'05)*. Rio de Janeiro, Brazil: IEEE, 212–217.

Grabczewski, K. and Jankowski, N., 2006. Mining for Complex Models Comprising Feature Selection and Classification. *In*: I. Guyon, S. Gunn, M. Nikravesh, and L.A. Zadeh, eds. *Feature Extraction*. Springer, Berlin, Heidelberg, 471–488.

Gu, X., Angelov, P.P., Kangin, D., and Principe, J.C., 2017. A new type of distance metric and its use for clustering. *Evolving Systems*, **8** (3), 167–177.

Guyon, I. and Elisseeff, A.A., 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, **3** (2003), 1157–1182.

Hall, M.A., 1999. Correlation-based feature subset selection for machine learning (Ph.D. thesis). The University of Waikato, Waikato, New Zealand.

Han, J. and Kamber, M., 2006. *Data Mining: Concepts and Techniques*. 2nd ed. San Francisco: Morgan Kaufmann.

Hancock, J.M., 2004. Jaccard Distance (Jaccard Index, Jaccard Similarity Coefficient). *In*: *Dictionary of Bioinformatics and Computational Biology*. Chichester, UK: John Wiley & Sons, Ltd.

Hechenbichler, K. and Schliep, K., 2004. Weighted k-Nearest-Neighbor Techniques and Ordinal Classification. *Collaborative Research Center 386*, **399**.

Jankowski, N. and Usowicz, K., 2011. Analysis of Feature Weighting Methods Based on Feature Ranking Methods for Classification. *In*: B.L. Lu, L. Zhang, and J. Kwok, eds. *Neural Information Processing*. Springer, Berlin, Heidelberg, 238–247.

Jia, H., Cheung, Y., and Liu, J., 2016. A New Distance Metric for Unsupervised Learning of Categorical Data. *IEEE Transactions on Neural Networks and Learning Systems*, **27** (5), 1065–1079.

Jiang, S., Xu, Y., Song, H., Wu, Q., Ng, M.K., Min, H., and Qiu, S., 2018. Multi-instance transfer metric learning by weighted distribution and consistent maximum likelihood estimation. *Neurocomputing*, **321**, 49–60.

Korenius, T., Laurikkala, J., and Juhola, M., 2007. On principal component analysis, cosine and Euclidean measures in information retrieval. *Information Sciences*, **177** (22), 4893–4905.

De Maesschalck, R., Jouan-Rimbaud, D., and Massart, D.L., 2000. The Mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, **50** (1), 1–18.

Manning, C.D. and Raghavan, P., 2009. An Introduction to Information Retrieval. *In*: *Online*. 1.

Monjardet, B., 1998. On the comparison of the Spearman and Kendall metrics between linear orders. *Discrete Mathematics*, **192** (1–3), 281–292.

Munkres, J.R., 2017. *Topology*. 2nd ed. Pearson, 119-121.

Nilsson, N.J., 1965. *Learning Machines: Foundations of trainable pattern-classifying systems*. New York: McGraw-Hill.

Norouzi, M., Fleet, D.J., and Salakhutdinov, R.R., 2012. Hamming Distance Metric Learning. *In*: Pereira F., C.J.C. Burges, L. Bottou, and K.Q. Weinberger, eds. *Advances in Neural Information Processing Systems 25 (NIPS 2012)*. Curran Associates, 1061–1069.

Parmar, J., Chouhan, S.S., Raychoudhury, V., and Rathore, S.S., 2021. Open-world Machine Learning: Applications, Challenges, and Opportunities. *ACM Computing Surveys*, **55** (10), 1–37.

Peng, Y., Hu, L., Ying, S., and Shen, C., 2018. Global Nonlinear Metric Learning by Gluing Local Linear Metrics. *In*: *Proceedings of the 2018 SIAM International Conference on Data Mining*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 423–431.

Rudin, W., 1976. *Principles of Mathematical Analysis*. 3rd ed. McGraw-Hill Education, 30-32.

Short, R. and Fukunaga, K., 1981. The optimal distance

measure for nearest neighbor classification. *IEEE Transactions on Information Theory*, **27** (5), 622–627.

Spearman, C., 1904. The Proof and Measurement of Association between Two Things. *The American Journal of Psychology*, **15** (1), 72.

Székely, G.J., Rizzo, M.L., and Bakirov, N.K., 2007. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, **35** (6), 2769–2794.

Taneja, S., Gupta, C., Goyal, K., and Gureja, D., 2014. An Enhanced K-Nearest Neighbor Algorithm Using Information Gain and Clustering. *In*: *2014 Fourth International Conference on Advanced Computing & Communication Technologies*. Rohtak, India: IEEE, 325–329.

Utkin, L. V. and Ryabinin, M.A., 2019. Discriminative Metric Learning with Deep Forest. *International Journal on Artificial Intelligence Tools*, **28** (02), 1950007.

Vivencio, D.P., R. Hruschka, E., do Carmo Nicoletti, M., dos Santos, E.B., and Galvao, S.D.C.O., 2007. Feature-weighted k-Nearest Neighbor Classifier. *In*: *2007 IEEE Symposium on Foundations of Computational Intelligence*. Honolulu, HI, USA: IEEE, 481–486.

Wolpert, D.H., 1996. The Lack of A Priori Distinctions Between Learning Algorithms. *Neural Computation*, **8** (7), 1341–1390.

Wolpert, D.H. and Macready, W.G., 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, **1** (1), 67–82.

Wolpert, D.H. and Macready, W.G., 2005. Coevolutionary Free Lunches. *IEEE Transactions on Evolutionary Computation*, **9** (6), 721–735.

Zabihzadeh, D., Monsefi, R., and Yazdi, H.S., 2019. Sparse Bayesian approach for metric learning in latent space. *Knowledge-Based Systems*, **178**, 11–24.

Zhang, W., Yan, Z., Xiao, G., Zhang, H., and Zuo, W., 2019. Learning Distance Metric for Support Vector Machine: A Multiple Kernel Learning Approach. *Neural Processing Letters*, **50**, 2899–2923.

Zhou, Z.-H. and Feng, J., 2019. Deep forest. *National Science Review*, **6** (1), 74–86.

Internet references

1- UCI Machine Learning Repository, http://archive.ics.uci.edu/ml, (28 Feb 2021).

2- Machine Learning Benchmark Problems, https://www.rdocumentation.org/packages/mlbench/versions/2.1-1, (7 Jul 2019).

3- MATLAB Sample Data Sets, https://www.mathworks.com/help/stats/sample-data-sets.html, (7 Jul 2019)