Time-Series Classification with Constrained DTW Distance and Inverse-Square Weighted *k*-NN

Zoltan Geler
Department of Media Studies, Faculty of Philosophy, University of Novi Sad
Novi Sad, Serbia
zoltang@ff.uns.ac.rs

Vladimir Kurbalija
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad
Novi Sad, Serbia
kurba@dmi.uns.ac.rs

Mirjana Ivanović
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad
Novi Sad, Serbia
mira@dmi.uns.ac.rs

Miloš Radovanović
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad
Novi Sad, Serbia
radacha@dmi.uns.ac.rs

***Abstract*—The problem of time-series classification witnessed the application of many techniques for data mining and machine learning, including neural networks, support vector machines, and Bayesian approaches. Somewhat surprisingly, the simple 1-nearest neighbor (1NN) classifier, in combination with the Dynamic Time Warping (DTW) distance measure, is still competitive and not rarely superior to more advanced classification methods, which includes the majority-voting k-nearest neighbor (kNN) classifier. In this paper we focus on the kNN classifier combined with the inverse-squared weighting scheme, and its interaction with constrained DTW distance. By performing experiments on the entire UCR Time Series Classification Archive we show that with proper selection of the constraint parameter *r* and neighborhood size *k*, inverse-square weighted kNN consistently outperforms 1NN.**

Keywords—time series, classification, nearest neighbor, inverse squared weighting, dynamic time warping

I. Introduction

Time series represent the most common form of temporal data [1]. They are made up of a sequence of real numbers, where each number signifies a measurement of some observed phenomenon in a particular moment of time. The measurements are usually made in equal time intervals [2], but that may not always be the case. Time-series analysis represents a prospective area of computer science which utilizes different methods in order to understand the underlying phenomenon or to make forecasts.

Time series are used for data analysis in almost all aspects of human activity, from science, medicine or ecology to technology, economy and telecommunications [2], [3]. Various machine-learning and data-mining techniques are used for discovering new and useful information from time series. These techniques include common task types such as: indexing, classification, clustering, prediction, segmentation, anomaly detection and others [4].

Classification and clustering represent two most fundamental tasks in time-series mining since they are usually sub-tasks of other algorithms including: anomaly detection, segmentation, rule induction, visualization, etc. [5]. For the purpose of classification, which is the main topic of this paper, several well-known machine-learning approaches were used, including: neural networks, support vector machines, Bayesian classifiers, decision trees and others. However, a number of influential research works highlight the fact that the simple 1NN classifier is highly competitive and that it can even generate better results than aforementioned more complicated approaches [4]–[6].

The majority-voting *k*NN algorithm is a natural extension of the 1NN algorithm. The choice of parameter *k* is of particular importance since it directly influences the quality of the classifier. Although the majority-voting *k*NN classifier cannot outperform the simple 1NN classifier with time-series data, our recent studies [7], [8] show that the *k*NN can be superior to 1NN when weighting schemes are introduced. These weighting schemes assign weights to the neighbors proportionately to their distance to the query object, and in such a way favor closer neighbors.

All mentioned NN classification techniques strongly rely on the choice of the distance measure. This measure must be carefully chosen in time series domain, since it should measure (usually subjective) notion of dissimilarity which is based mainly on shapes appearing in time series objects. The simplest Euclidean measure [9] is intuitive and easy for computation, but it is impractical in most applications since it is sensible to distortions and shiftings along the time axis [10]. To cope with this problem several elastic measures were developed like: Dynamic Time Warping (DTW) [11], Longest Common Subsequence (LCS) [12], Edit Distance with Real Penalty (ERP) [13], and Edit Distance on Real sequence (EDR) [14]. All these measures have better classification accuracies than Euclidean measure, but they are more computationally demanding. The introduction of global constraint parameter *r* into the calculation of elastic distance measures can significantly reduce the computation costs and in some cases even improve the classification accuracies [15], [16].

The purpose of this paper is to provide insight into the performance of the weighted kNN classifier with the most commonly known weighting scheme (the inverse squared distance—IS [17]) and the most promising underlying distance measure (DTW). The paper also inspects the relationship between the two most important parameters for weighted kNN classification with elastic distance measure: *k* and *r*. All the presented experiments were performed using the Framework for Analysis and Prediction (FAP) library [18]. The world's largest publicly available collection of labeled time series datasets [19] is used for experimental part of the research.

The rest of the paper is organized as follows. Required background knowledge together with the overview of the

978-1-7281-6799-2/20/$31.00 ©2020 IEEE

related work is given in the next section. Section 3 contains the descriptions of the experiments, while in Section 4 the results are presented and discussed. The last section concludes the paper and gives directions for future research.

II. Background and Related Work

The choice of the appropriate distance (or similarity) measure is the most important aspect of many distance-based data analysis tasks like classification, clustering and indexing. In the domain of time series, Euclidian distance is considered as the simplest and most intuitive distance measure [20]. It is calculated as the sum of the distances between corresponding data-points. However, due to the linear aligning of the points of the time series it is sensitive to distortions and shifting along the time axis [10].

To overcome this shortcoming several elastic distance measures are proposed, such as: DTW, EDR, ERP and LCS. Among them, the most commonly used measure is definitely DTW [4]–[7], [10], [11]. This measure is similar to Euclidean distance, but it allows one-to-many alignment of points. Such alignment allows a meaningful comparison of two time series even if they are shifted along the x-axis.

The algorithm for DTW computation tries to find an optimal path in the matrix which consists of distances between all points of time series $Q = (q_1, q_2, \dots, q_n)$ and $S = (s_1, s_2, \dots, s_m)$ based on the following recursive formula:

$$D_{i,j} = \begin{cases} 0 & i = j = 0 \\ \infty & i = 0, j > 0 \text{ or } i > 0, j = 0 \\ d(q_i, s_j) + min \begin{cases} D_{i-1,j-1} \\ D_{i-1,j} \\ D_{i,j-1} \end{cases} & i, j \geq 1 \end{cases}$$

for $0 \leq i \leq n$ and $0 \leq j \leq m$, where $d(q_i, s_j)$ is the distance between q_i and s_j, and $D_{n,m}$ gives the distance between Q and S.

This algorithm is implemented using dynamic programming which is of quadratic complexity. So, although DTW is more robust than Euclidean distance concerning sensitivity to noise and time axis shifts, it is far more computationally demanding. Furthermore, the unrestricted alignments of points can even lead to pathological aligning of the points where a small part of one time series maps onto a large section of the other time series. The introduction of the global constraints in the dynamic programming algorithm can improve both of these problems. Global constraints, such as Sakoe-Chiba band [21], narrow the warping window around the diagonal in the search matrix. The width of the constraint is controlled with the parameter *r*, which is usually represented as the percentage of the time-series length.

Let Q and S be time series of the same length n. By applying the Sakoe-Chiba constraint to the DTW algorithm, the distance $D_{n,n}$ between them is calculated based on the following formula:

$$D_{i,j} = \begin{cases} 0 & i = j = 0 \\ \infty & i = 0 \text{ } or \text{ } j = 0 \\ \infty & i, j \geq 1, |i - j| = w \\ d(q_i, s_j) + min \begin{cases} D_{i-1,j-1} \\ D_{i-1,j} \\ D_{i,j-1} \end{cases} & i, j \geq 1, |i - j| < w \end{cases}$$

for $0 \leq i, j \leq n$, where the value of w is obtained by multiplying the length of the time series (n) by the value of the parameter r.

In addition to the Sakoe-Chiba band, there are other forms of constraing the warping window. For example, in [23] we have shown that, when it comes to the 1NN classifier, the Itakura parallelogram produced lower error-rates than the Sakoe-Chiba band in the case of 1/3 of the analyzed datasets (28 out of 85).

The introduction of global constraints improves the computational effectiveness and prevents unnatural aligning between points. Besides that, the constrained measures represent qualitatively different measures than their unconstrained counterparts (especially for r< 15% of the length of the time series) [15], mostly with higher classification accuracies [16].

The simple 1NN classifier is usually considered as a baseline classifier in the field of time series [4]–[6]. A general standpoint in the community is that its accuracy is hard to beat. The probable reason for this is the fact that the first neighbor has a particularly important role in the field of time series. The natural extension of 1NN, the majority voting *k*NN classifier cannot outperform the accuracy of 1NN for an arbitrary value of *k>1* [4], [6]. However, the approach which weights the votes of the neighbors in the process of majority-voting, in accordance with their similarity with the query object, gives advantage to *k*NN. The weighting approach still favors the first neighbor, but gives a chance to the other close neighbors to vote (with reduced weights). In our previous researches [7], [8], among many weighting schemes the inverse squared distance—IS [17] is selected as one of the most prominent ones. However, those previous studies were based mainly on unconstrained versions of DTW. In this paper, we will investigate the behavior of IS weighting scheme in kNN classifier with constrained versions of DTW measure. Furthermore, previous studies were based on the evaluation on the old version of time-series repository (46 datasets), while this study is based on the new version (128 datasets).

III. Experimental Setup

In this paper, we examine whether *k*NN and its weighted version (denoted as IS) can compete with 1NN when it comes to DTW constrained using the Sakoe-Chiba band. Weights were calculated as the reciprocal values of the squared distances between the time series.

Classification accuracies were obtained by applying 10 runs of stratified tenfold cross-validation (SCV10x10). In each run, the datasets were divided into 10 approximately equal sized stratified disjoint subsets, in a random way. The optimal size of the Sakoe-Chiba band (and, in case of the kNN and IS classifiers, the optimal number of the neighbors) was chosen by stratified nine-fold cross-validation (SCV1x9) on the training set (composed of 9 out of the 10 subsets). The accuracy was then computed on the test set (the remaining subset) using the optimal values of the parameters *r* (the width of the warping window) and *k* (the number of neighbors). Finally, for each dataset, the average value of the 100 thus obtained classification accuracies and optimal parameter values were calculated.

Drawing on the results of some of our previous research [7], [8], [15], [16], [22], [23], the following widths of the Sakoe-Chiba band were considered: 100% (unconstrained distance), 90%, 80%, 70%, 60%, 50%, 45%, 40%, 35%, 30%, and all values from 25% to 0% in steps of 1% of the length of time series. The optimal value of the number of nearest neighbors was sought in the range between 1 and 30.

TABLE I. AVERAGE CLASSIFICATION ACCURACIES (IN PERCENTAGES)

ID	Dataset	1NN	kNN	IS	ID	Dataset	1NN	kNN	IS
1	ACSF1	**58.95**	58.70	57.20	65	ItalyPowerDemand	96.71	96.88	**97.14**
2	Adiac	**67.50**	67.13	67.16	66	LargeKitchenAppliances	**81.57**	81.05	80.91
3	AllGestureWiimoteX	**73.46**	**73.46**	72.78	67	Lightning2	**89.24**	89.00	88.51
4	AllGestureWiimoteY	76.84	76.40	**77.67**	68	Lightning7	77.58	77.46	**77.61**
5	AllGestureWiimoteZ	71.21	70.88	**71.87**	69	Mallat	98.72	98.70	**98.87**
6	ArrowHead	89.77	89.63	**90.01**	70	Meat	**99.83**	99.58	**99.83**
7	BME	**99.89**	**99.89**	**99.89**	71	MedicalImages	80.35	80.04	**82.49**
8	Beef	**58.50**	55.33	58.33	72	MelbournePedestrian	86.22	**86.74**	**86.74**
9	BeetleFly	**87.50**	85.25	86.75	73	MiddlePhalanxOutlineAgeGroup	69.62	**74.50**	74.35
10	BirdChicken	**95.75**	94.00	**95.75**	74	MiddlePhalanxOutlineCorrect	79.33	79.88	**80.02**
11	CBF	**99.91**	**99.91**	**99.91**	75	MiddlePhalanxTW	55.17	62.11	**62.51**
12	Car	**80.25**	79.58	78.33	76	MixedShapesRegularTrain	**94.17**	94.14	94.10
13	Chinatown	97.07	**97.75**	97.56	77	MixedShapesSmallTrain	**93.94**	93.90	93.90
14	ChlorineConcentration	**99.77**	**99.77**	99.76	78	MoteStrain	**95.04**	94.98	94.62
15	CinCECGTorso	**99.92**	**99.92**	**99.92**	79	NonInvasiveFetalECGThorax1	84.30	84.12	**84.84**
16	Coffee	**99.10**	98.70	**99.10**	80	NonInvasiveFetalECGThorax2	90.54	90.75	**91.33**
17	Computers	70.90	72.00	**72.72**	81	OSULeaf	72.34	71.75	**73.91**
18	CricketX	80.19	79.74	**81.03**	82	OliveOil	**88.83**	88.50	88.17
19	CricketY	**79.49**	78.74	79.40	83	PLAID	81.47	81.11	**82.45**
20	CricketZ	80.49	80.24	**80.51**	84	PhalangesOutlinesCorrect	77.80	79.29	**79.53**
21	Crop	77.07	77.07	**77.28**	85	Phoneme	34.39	38.01	**38.94**
22	DiatomSizeReduction	**99.94**	99.91	**99.94**	86	PickupGestureWiimoteZ	71.50	69.40	**73.10**
23	DistalPhalanxOutlineAgeGroup	76.58	81.19	**82.12**	87	PigAirwayPressure	16.70	**16.70**	**16.70**
24	DistalPhalanxOutlineCorrect	77.13	**78.91**	78.86	88	PigArtPressure	28.63	28.05	**29.27**
25	DistalPhalanxTW	71.26	75.25	**75.75**	89	PigCVP	**23.84**	**23.84**	23.75
26	DodgerLoopDay	51.33	55.69	**57.75**	90	Plane	**99.62**	**99.62**	**99.62**
27	DodgerLoopGame	93.99	94.57	**96.08**	91	PowerCons	**94.69**	**94.69**	94.64
28	DodgerLoopWeekend	**99.75**	**99.75**	**99.75**	92	ProximalPhalanxOutlineAgeGroup	75.26	**84.40**	83.95
29	ECG200	88.95	88.75	**90.90**	93	ProximalPhalanxOutlineCorrect	82.05	83.75	**84.24**
30	ECG5000	93.36	**94.92**	94.90	94	ProximalPhalanxTW	76.40	80.46	**80.50**
31	ECGFiveDays	99.77	99.77	**99.85**	95	RefrigerationDevices	61.25	59.20	**62.89**
32	EOGHorizontalSignal	68.24	68.20	**69.61**	96	Rock	86.57	86.43	**87.00**
33	EOGVerticalSignal	66.19	65.95	**67.98**	97	ScreenType	**50.75**	48.87	50.17
34	Earthquakes	70.95	79.25	**79.31**	98	SemgHandGenderCh2	89.40	90.08	**90.38**
35	ElectricDevices	80.65	81.12	**81.49**	99	SemgHandMovementCh2	67.26	68.54	**68.89**
36	EthanolLevel	26.96	**29.67**	28.40	100	SemgHandSubjectCh2	84.12	83.68	**84.24**
37	FaceAll	**98.41**	**98.41**	98.39	101	ShakeGestureWiimoteZ	**90.90**	90.70	**90.90**
38	FaceFour	94.74	93.40	**94.95**	102	ShapeletSim	**91.00**	90.80	**90.65**
39	FacesUCR	**98.43**	98.42	98.41	103	ShapesAll	85.89	85.89	**86.06**
40	FiftyWords	80.69	80.38	**81.35**	104	SmallKitchenAppliances	69.68	71.20	**71.27**
41	Fish	**86.29**	85.14	85.29	105	SmoothSubspace	93.20	95.13	**95.33**
42	FordA	68.45	76.31	**76.68**	106	SonyAIBORobotSurface1	**98.29**	98.16	98.29
43	FordB	67.47	77.76	**77.89**	107	SonyAIBORobotSurface2	**98.59**	98.56	98.51
44	FreezerRegularTrain	99.40	99.40	**99.41**	108	StarLightCurves	93.61	94.05	**94.08**
45	FreezerSmallTrain	99.38	99.38	**99.43**	109	Strawberry	**96.33**	**96.33**	96.24
46	Fungi	**100.00**	**100.00**	**100.00**	110	SwedishLeaf	**88.41**	88.24	88.21
47	GestureMidAirD1	**60.99**	60.40	60.19	111	Symbols	98.26	98.24	**98.29**
48	GestureMidAirD2	58.44	**60.40**	59.50	112	SyntheticControl	99.23	99.40	**99.45**
49	GestureMidAirD3	28.14	26.13	**29.65**	113	ToeSegmentation1	**85.39**	85.17	85.04
50	GesturePebbleZ1	93.15	92.70	**94.37**	114	ToeSegmentation2	88.90	88.55	**89.50**
51	GesturePebbleZ2	93.29	92.34	**94.37**	115	Trace	**99.90**	**99.90**	**99.90**
52	GunPoint	98.50	98.55	**98.85**	116	TwoLeadECG	**99.91**	99.88	**99.91**
53	GunPointAgeSpan	96.52	96.27	**97.27**	117	TwoPatterns	**100.00**	**100.00**	**100.00**
54	GunPointMaleVersusFemale	99.42	99.56	**99.58**	118	UMD	**99.28**	**99.28**	**99.28**
55	GunPointOldVersusYoung	**97.98**	97.94	97.76	119	UWaveGestureLibraryAll	**98.08**	98.08	98.04
56	Ham	**80.18**	78.25	79.76	120	UWaveGestureLibraryX	80.67	**81.71**	81.65
57	HandOutlines	86.72	87.10	**87.45**	121	UWaveGestureLibraryY	74.36	76.72	**76.73**
58	Haptics	44.97	49.10	**49.86**	122	UWaveGestureLibraryZ	74.54	74.98	**75.28**
59	Herring	**58.92**	55.21	57.93	123	Wafer	**99.86**	99.85	99.85
60	HouseTwenty	93.70	94.15	**94.58**	124	Wine	**99.91**	**99.91**	**99.91**
61	InlineSkate	56.34	54.18	**58.43**	125	WordSynonyms	82.13	81.71	**82.69**
62	InsectEPGRegularTrain	**94.56**	94.43	94.43	126	Worms	59.51	59.51	**64.01**
63	InsectEPGSmallTrain	**95.31**	95.12	94.97	127	WormsTwoClass	68.31	66.56	**72.60**
64	InsectWingbeatSound	60.69	**67.32**	67.04	128	Yoga	95.04	95.04	**95.36**

The statistical significance of the differences between the average accuracies of the examined classifiers was checked using the Friedman test [24] and the Bonferroni-adjusted [25] Wilcoxon signed-rank test [26]. In addition, the statistically significant wins and losses counts were determined relying on a significance level of 0.001 when conducting the corrected resampled t-test [27].

The experiments were performed on the 128 real-world datasets of the UCR Time Series Classification Archive [19] prepared by utilizing Paparizzos's script [28]. This collection contains the majority of all publicly available, labelled time-series datasets in the world and it covers a wide range of different topics, including: robotics, physics, medicine, sport, biology, astronomy, seismology, food science, and others. While the smallest dataset consists of 40 time series, the largest one contains as many as 24000. The length of the time series is between 15 and 2844, and the number of classes varies between 2 and 60.

Classification accuracies were calculated by employing the free Framework for Analysis and Prediction (FAP) Java library [18] which implements various time-series data mining algorithms, including time-series representations, preprocessing, distance measures, classification, and methods for evaluating the accuracy of classifiers. This platform has already been employed to carry out experiments in a variety of research domains [29] and it is actively used in teaching [30], too.

IV. Experimental Results

The obtained SCV10x10 classification accuracies of the three considered classifier (1NN, unweighted *k*NN, and weighted kNN) are shown in Table I in which the best results are in bold. According to this category, of a total of 128 datasets, the 1-nearest neighbor classifier generated the best results in 50 cases, the majority-voting k-nearest neighbor algorithm in 27 cases, and the inverse-squared variation of the weighted kNN method in 85 cases.

The labels in the first column of Table I (denoted as ID) are used in Table II that displays the detected average optimal values of parameters *k* and *r* for each dataset.

A general picture of the examined classifiers' performance can be read from Table III and IV, which summarize the detailed results from Table I and II.

TABLE II. Optimal Values of Parameters *K* and *R*

	1NN	kNN		IS			1NN	kNN		IS			1NN	kNN		IS			1NN	kNN		IS	
ID	**r**	**k**	**r**	**k**	**r**	**ID**	**r**	**k**	**r**	**k**	**r**	**ID**	**r**	**k**	**r**	**k**	**r**	**ID**	**r**	**k**	**r**	**k**	**r**
1	18.9	1.1	19.2	4.6	20.2	33	2.7	1.5	2.9	17.1	3.6	65	0.0	8.4	0.0	8.8	0.1	97	16.7	2.4	18.9	16.8	16.4
2	0.8	1.4	0.8	3.3	0.6	34	7.4	13.9	10.7	13.8	10.3	66	65.1	1.6	65.7	4.9	47.9	98	1.4	5.1	1.5	6.3	1.4
3	16.5	1.0	16.5	2.7	16.1	35	10.9	3.9	11.1	9.3	12.5	67	8.9	1.3	9.7	2.2	9.7	99	1.2	5.1	1.3	6.1	1.2
4	12.8	1.6	13.4	7.5	13.1	36	1.4	19.0	1.3	17.8	0.9	68	8.4	3.9	12.0	6.0	10.6	100	1.0	4.3	1.2	5.7	1.1
5	11.1	2.1	13.3	4.4	13.8	37	3.6	1.0	3.6	1.6	3.5	69	4.4	2.0	5.2	4.3	6.3	101	30.1	1.1	29.9	1.3	30.5
6	0.2	4.2	0.2	7.3	0.1	38	2.7	4.1	3.0	5.8	2.4	70	0.0	1.4	0.0	1.4	0.0	102	2.3	1.3	2.4	2.6	2.6
7	3.0	1.0	3.0	1.0	3.0	39	3.5	1.0	3.5	1.7	3.6	71	11.4	2.8	12.1	13.2	12.9	103	4.8	1.0	4.8	3.1	5.4
8	0.6	2.8	0.9	4.5	0.3	40	6.0	1.8	6.3	6.0	7.2	72	0.0	3.8	0.0	5.3	0.0	104	13.8	4.9	21.7	8.8	19.3
9	8.3	1.3	8.7	1.8	8.6	41	1.1	1.8	1.1	4.2	1.2	73	0.7	24.6	3.7	21.7	2.7	105	6.9	10.6	7.0	11.6	7.0
10	6.7	1.1	6.9	3.1	7.3	42	0.8	26.4	1.0	28.0	1.0	74	0.0	5.1	0.0	6.1	0.0	106	0.8	1.5	0.8	3.0	0.9
11	1.9	1.0	1.9	1.0	1.9	43	1.1	27.7	2.0	28.7	1.9	75	4.3	21.9	5.0	23.3	5.8	107	1.9	1.3	1.9	2.2	1.8
12	1.4	1.6	1.2	3.1	1.5	44	1.1	1.0	1.1	2.3	1.1	76	2.7	1.1	2.7	4.3	2.3	108	10.0	3.4	12.9	4.3	12.7
13	0.2	9.3	0.9	11.2	2.5	45	1.6	1.0	1.6	2.3	1.5	77	1.6	1.2	1.5	4.1	1.6	109	4.2	1.0	4.2	1.4	4.0
14	0.0	1.0	0.0	1.0	0.0	46	0.4	1.0	0.4	1.0	0.4	78	26.9	1.1	26.3	8.1	16.1	110	2.4	1.3	2.4	2.6	2.4
15	2.0	1.0	2.0	1.0	2.0	47	9.5	1.3	10.0	2.4	10.9	79	0.0	4.6	0.5	7.3	0.3	111	6.7	1.1	6.7	2.5	6.4
16	0.6	2.0	0.8	2.1	0.9	48	30.1	4.6	37.0	6.2	41.0	80	0.0	4.4	0.0	6.5	0.1	112	7.9	3.0	10.6	3.2	10.4
17	51.5	5.8	34.5	11.4	36.0	49	4.7	7.5	8.5	13.7	3.1	81	5.8	1.8	5.6	5.3	5.6	113	18.9	1.1	19.0	3.3	16.8
18	9.8	2.0	9.6	4.8	7.7	50	2.6	9.8	2.5	19.0	2.4	82	0.1	2.0	0.1	2.6	0.1	114	7.5	1.7	6.9	2.4	6.9
19	10.2	1.5	11.5	4.3	10.5	51	2.6	10.7	2.3	18.2	2.4	83	22.3	1.7	20.7	5.3	21.9	115	3.8	1.0	3.8	1.0	3.8
20	7.3	2.1	8.6	4.1	8.6	52	2.8	3.6	3.3	4.6	3.8	84	0.3	4.6	0.3	5.4	0.3	116	3.4	1.8	4.0	1.8	3.9
21	0.0	1.0	0.0	3.8	0.0	53	6.3	2.2	5.8	5.0	5.1	85	10.7	11.0	20.2	15.0	22.5	117	4.2	1.0	4.2	1.0	4.2
22	0.5	1.0	0.5	1.0	0.5	54	4.4	1.9	5.3	2.0	5.5	86	14.6	3.0	13.4	10.9	21.5	118	2.6	1.6	2.9	2.8	4.1
23	4.3	13.7	6.4	13.1	6.0	55	7.1	1.0	7.1	3.1	7.3	87	7.5	1.0	7.5	1.1	7.5	119	3.0	1.0	3.0	1.8	3.1
24	3.5	6.9	4.5	7.8	4.4	56	0.2	3.3	0.4	8.4	0.6	88	5.0	1.8	5.1	6.1	5.7	120	6.0	4.1	6.7	5.5	6.2
25	1.0	20.2	4.6	21.6	4.3	57	1.4	4.1	3.2	6.6	5.2	89	15.8	1.0	15.8	1.1	15.8	121	4.0	5.3	6.5	9.8	5.0
26	1.2	5.0	1.5	6.2	1.5	58	4.0	10.8	4.7	16.9	4.4	90	5.0	1.0	5.0	1.0	5.0	122	3.7	3.7	5.5	5.3	4.3
27	1.7	3.4	2.7	3.9	2.5	59	5.0	17.5	3.1	10.1	3.6	91	5.5	1.0	5.5	9.3	6.0	123	1.2	1.1	1.2	1.6	1.2
28	0.1	1.0	0.1	1.0	0.1	60	35.9	4.3	34.1	4.6	36.1	92	1.8	25.7	1.2	26.0	1.3	124	0.1	1.0	0.1	1.0	0.1
29	1.3	2.4	1.0	4.0	0.8	61	3.9	3.4	3.9	11.7	4.9	93	1.6	5.2	0.9	7.6	0.7	125	6.4	1.6	6.4	5.0	7.3
30	3.1	5.1	4.5	9.8	2.6	62	18.5	1.0	18.6	1.2	18.3	94	2.6	11.9	3.2	12.7	3.5	126	21.5	1.0	21.5	22.2	8.4
31	1.0	1.7	1.0	2.4	1.0	63	19.5	1.4	19.6	2.5	18.5	95	6.5	5.1	11.2	19.0	12.5	127	12.1	8.7	22.9	17.6	24.4
32	3.2	1.7	3.0	8.8	3.0	64	1.0	11.4	2.3	24.8	2.2	96	0.3	2.6	0.8	4.6	0.4	128	2.0	1.0	2.0	4.2	2.1

Table III presents the number of datasets for which the given classifier produced larger (>), smaller (<), and equal (=) accuracy compared to the other methods. These results indicate that IS is more successful than both *k*NN and 1NN and that the number of datasets for which *k*NN underperformed 1NN is greater than that for which it outperformed 1NN (64 vs. 41). It should be also noted that in the case of 28 datasets, the 1NN classifier generated better results than both *k*NN and IS.

TABLE III. The Number of Datasets for Which the Given Method Produced Larger (>), Smaller (<), and Equal (=) Accuracy Compared to the Other Methods

	>	<	=	
kNN	41	64	23	**1NN**
IS	78	34	16	**1NN**
IS	82	33	13	**kNN**

According to Table IV, the highest average accuracy (82.66%) and the lowest standard deviation (17.7) were achieved by the IS classifier. The lowest average accuracy

(81.59%) and the highest standard deviation (18.37) were produced by the 1NN method. As for the average width of the warping window, it is between 6.6 (IS) and 7.1 (*k*NN). The deviation of the parameter *r* is the highest in the case of 1NN (9.5) and the lowest in the case of IS (8.6). In terms of the average number of the nearest neighbors, the weighted NN classifier generated higher value than the unweighted version (7 vs. 4.4).

TABLE IV. AVERAGE ACCURACIES (IN PERCENTAGES) AND AVERAGE VALUES OF THE PARAMETERS *K* AND *R* (*AVG*) ALONG WITH THE CORRESPONDING STANDARD DEVIATIONS (*STDEV*)

	1NN		**kNN**			**IS**		
	acc.	**r**	**acc.**	**k**	**r**	**acc.**	**k**	**r**
avg	81.59	6.6	82.10	4.4	7.1	82.66	7.0	6.9
stdev	18.37	9.5	18.04	5.6	9.3	17.70	6.4	8.6

Table IV shows that the differences between the average classification accuracies are not particularly big. However, the result of the Friedman test ($\chi^2(2)$=31.664, p<0.001) indicates that the differences between the performance of the examined classifiers are statistically significant. Furthermore, the p-values of the pairwise Wilcoxon sign-rank test adjusted by applying the Bonferroni scheme reveals that the statistically significant differences are between the weighted kNN and the other two classifiers (Table V). The advantage of the IS approach has been further confirmed by the number of statistically significant wins and losses computed using the corrected resampled t-test (Table VI).

TABLE V. BONFERRONI-ADJUSTED P VALUES FOR THE PAIRWISE WILCOXON SIGN-RANK TEST OF THE DIFFERENCES IN AVERAGE ACCURACIES ACROSS THE DATASETS

ACCURACY		82.66	82.10	81.59
		IS	**kNN**	**1NN**
82.66	**IS**		0.00000	0.00000
82.10	**kNN**			1.00000
81.59	**1NN**			

TABLE VI. STATISTICALLY SIGNIFICANT WINS AND LOSSES COUNTS ACROSS ALL DATASETS

	IS	**kNN**	**1NN**
W	69	24	4
L	0	31	66
W-L	69	-7	-62

Based on Fig. 1 which depicts the frequencies of different values of the parameter *r* and the percentages of folds for which the best results were obtained with $r \leq 10$ and $r > 10$, it can be concluded that all three classifiers prefer narrow warping windows: for more than 80% of the folds (observed over all datasets) the optimum value of parameter *r* is not greater than 10.

When it comes to the optimal number of neighbors, Fig. 2 clearly shows the difference between the unweighted and the weighted kNN. While the percentage of the folds for which the majority-voting kNN was reduced to 1NN is about 53%, in the case of the IS classifier it is only about 29%. It seems that when weighting is applied, better results are more likely to be obtained if more than one (two) neighbors are considered.

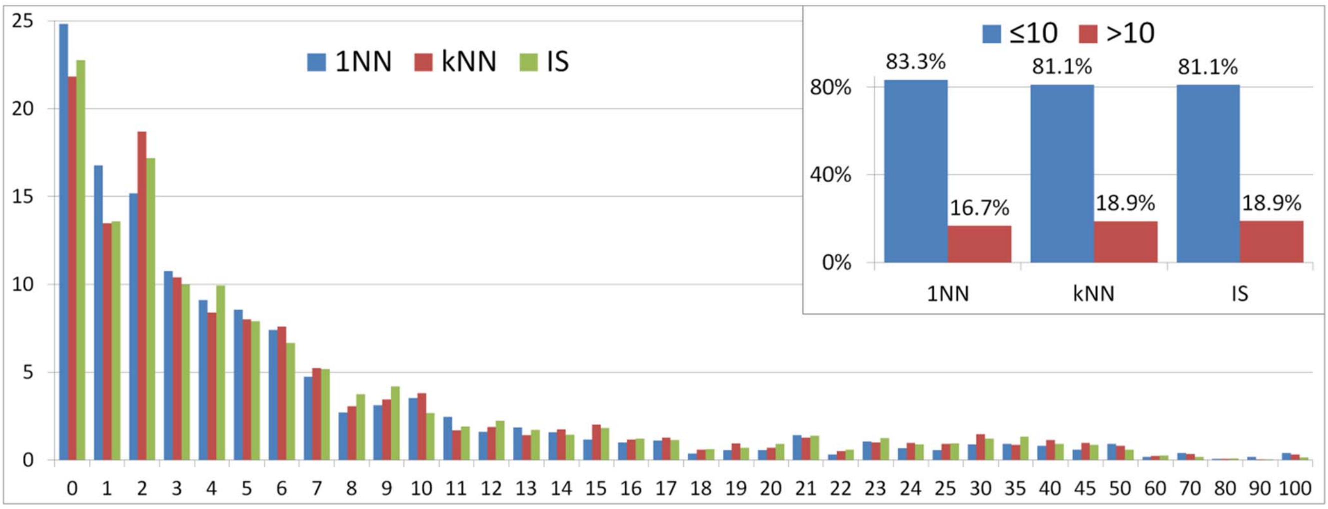

Fig. 1. Frequencies (in hundreds) of different values of the parameter *r* and the percentages of folds for which the best results were obtained with $r \leq 10$ and $r > 10$ (in the upper-right corner)

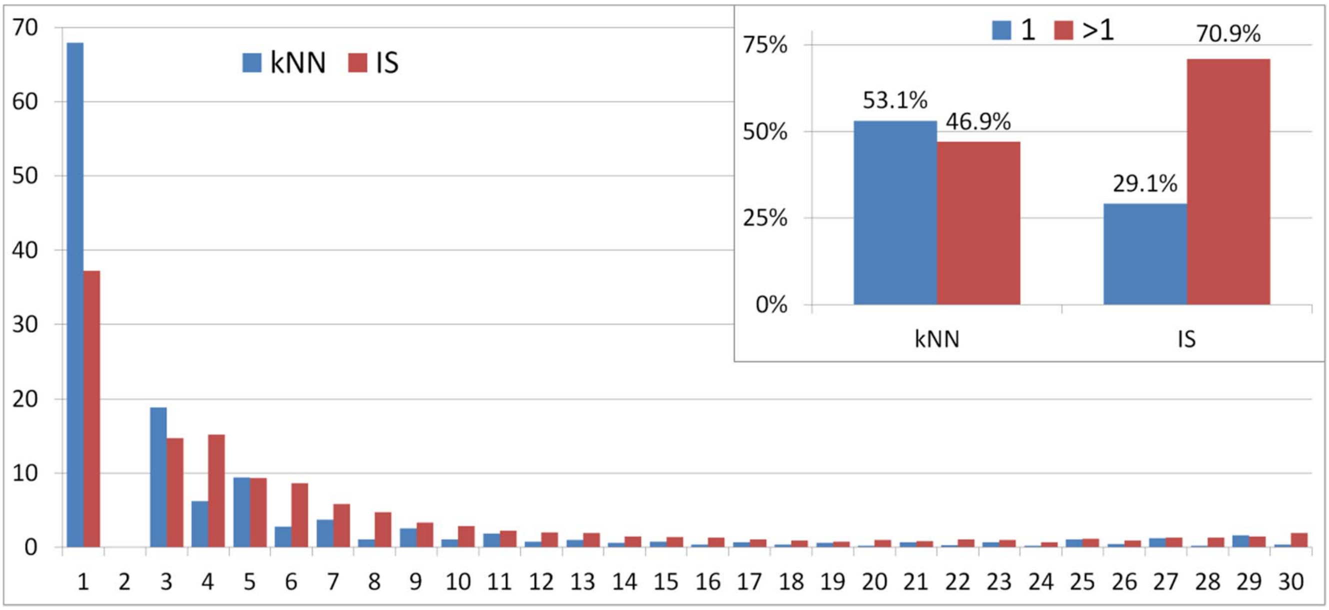

Fig. 2. Frequencies (in hundreds) of different values of the parameter *k* and the percentages of folds for which the best results were obtained with *k*=1 and *k*>1 (in the upper-right corner)

V. Conclusion

In spite of the fact that 1NN is a very competitive classifier for time series, we have shown that it can be outperformed by kNN when using constrained DTW distance, by introducing inverse-squared weighting to kNN. Our findings were supported by an extensive empirical study and multiple statistical tests on all data sets from the UCR Time-Series Archive.

In future work we plan to extend our analysis to other constrained elastic time-series distance measures and other kNN weighting schemes in order to determine which combination can generally be considered superior, in addition to providing more insight into their interactions and guidelines for their selection and application.

Acknowledgments

The authors acknowledge financial support of the Ministry of Education, Science and Technological Development of the Republic of Serbia (Grant No. 620 451-03-68/2020-14/200125). The authors would like to express special thanks to Eamonn Keogh for collecting and making available the UCR time-series datasets, and also to everyone who contributed data to the collection, without which the presented work would not have been possible. M. Ivanović is grateful for the support throughout the "CA15109—European Cooperation for Statistics of Network Data Science" project.

References

[1] S. Laxman and P. S. Sastry, "A survey of temporal data mining," *Sadhana*, vol. 31, no. 2, pp. 173–198, Apr. 2006.

[2] P. J. Brockwell and R. A. Davis, *Introduction to Time Series and Forecasting*, no. v. 1. New York, NY: Springer New York, 2002.

[3] D. J. Hand, H. Mannila, and P. Smyth, *Principles of Data Mining*. Cambridge, MA, USA: MIT Press, 2001.

[4] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: experimental comparison of representations and distance measures," in *Proceedings of the VLDB Endowment*, 2008, vol. 1, no. 2, pp. 1542–1552.

[5] H. A. Dau *et al.*, "Optimizing dynamic time warping's window width for time series data mining applications," *Data Min. Knowl. Discov.*, vol. 32, no. 4, pp. 1074–1120, Jul. 2018.

[6] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana, "Fast time series classification using numerosity reduction," in *Proceedings of the 23rd international conference on Machine learning - ICML '06*, 2006, pp. 1033–1040.

[7] Z. Geler, V. Kurbalija, M. Radovanović, and M. Ivanović, "Impact of the Sakoe-Chiba Band on the DTW Time Series Distance Measure for kNN Classification," in *The 7th International Conference on Knowledge Science, Engineering and Management KSEM 2014*, vol. 8793, R. Buchmann, C. V. Kifor, and J. Yu, Eds. Cham: Springer International Publishing, 2014, pp. 105–114.

[8] Z. Geler, V. Kurbalija, M. Radovanović, and M. Ivanović, "Comparison of different weighting schemes for the kNN classifier on time-series data," *Knowl. Inf. Syst.*, vol. 48, no. 2, pp. 331–378, Aug. 2016.

[9] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," *ACM SIGMOD Rec.*, vol. 23, no. 2, pp. 419–429, Jun. 1994.

[10] C. A. Ratanamahatana and E. Keogh, "Three Myths about Dynamic Time Warping Data Mining," in *Proceedings of the 2005 SIAM International Conference on Data Mining*, Philadelphia, PA: Society for Industrial and Applied Mathematics, 2005, pp. 506–510.

[11] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Knowledge Discovery in Databases: Papers from the 1994 AAAI Workshop*, 1994, vol. 10, no. 16, pp. 359–370.

[12] M. Vlachos, G. Kollios, and D. Gunopulos, "Discovering similar multidimensional trajectories," in *Proceedings 18th International Conference on Data Engineering*, 2002, pp. 673–684.

[13] L. Chen and R. Ng, "On The Marriage of Lp-norms and Edit Distance," in *Proceedings 2004 VLDB Conference*, vol. 04, M. A. Nascimento, M. T. Özsu, D. Kossmann, R. J. Miller, J. A. Blakeley, and K. B. Schiefer, Eds. Elsevier, 2004, pp. 792–803.

[14] L. Chen, M. T. Özsu, and V. Oria, "Robust and fast similarity search for moving object trajectories," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data - SIGMOD '05*, 2005, pp. 491–502.

[15] V. Kurbalija, M. Radovanović, Z. Geler, and M. Ivanović, "The Influence of Global Constraints on DTW and LCS Similarity Measures for Time-Series Databases," in *Third International Conference on Software, Services and Semantic Technologies S3T 2011 SE - 10*, 2011, vol. 101, pp. 67–74.

[16] V. Kurbalija, M. Radovanović, Z. Geler, and M. Ivanović, "The influence of global constraints on similarity measures for time-series databases," *Knowledge-Based Syst.*, vol. 56, pp. 49–67, Jan. 2014.

[17] S. A. Dudani, "The Distance-Weighted k-Nearest-Neighbor Rule," *IEEE Trans. Syst. Man. Cybern.*, vol. SMC-6, no. 4, pp. 325–327, Apr. 1976,

[18] V. Kurbalija, M. Radovanović, Z. Geler, and M. Ivanović, "A Framework for Time-Series Analysis," in *Artificial Intelligence: Methodology, Systems, and Applications SE - 5*, 2010, vol. 6304, pp. 42–51.

[19] H. Anh Dau *et al.*, "The UCR Time Series Classification Archive," 2018. https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

[20] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient similarity search in sequence databases," in *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms (FODO '93)*, vol. 730, David B. Lomet, Ed. Springer Berlin Heidelberg, 1993, pp. 69–84.

[21] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust.*, vol. 26, no. 1, pp. 43–49, Feb. 1978.

[22] Z. Geler, "Role of Similarity Measures in Time Series Analysis," PhD thesis, University of Novi Sad, Serbia, 2015.

[23] Z. Geler, V. Kurbalija, M. Ivanovic, M. Radovanovic, and W. Dai, "Dynamic Time Warping: Itakura vs Sakoe-Chiba," in *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, 2019, pp. 1–6.

[24] S. García, A. Fernández, J. Luengo, and F. Herrera, "Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power," *Inf. Sci. (Ny).*, vol. 180, no. 10, pp. 2044–2064, 2010.

[25] S. García and F. Herrera, "An Extension on 'Statistical Comparisons of Classifiers over Multiple Data Sets' for all Pairwise Comparisons," *J. Mach. Learn. Res.*, vol. 9, pp. 2677–2694, 2008.

[26] S. García, A. Fernández, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability," *Soft Comput.*, vol. 13, no. 10, pp. 959–977, Aug. 2009.

[27] R. R. Bouckaert and E. Frank, "Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms," in *Advances in Knowledge Discovery and Data Mining*, vol. 3056, H. Dai, R. Srikant, and C. Zhang, Eds. Springer Berlin Heidelberg, 2004, pp. 3–12.

[28] J. Paparrizos, "2018 UCR Time-Series Archive: Backward Compatibility, Missing Values, and Varying Lengths." 2019.

[29] V. Kurbalija, M. Ivanović, Z. Geler, and M. Radovanović, "Two Faces of the Framework for Analysis and Prediction, Part 2 - Research," *Inf. Technol. Control*, vol. 47, no. 3, pp. 489–502, Sep. 2018.

[30] V. Kurbalija, M. Ivanović, Z. Geler, and M. Radovanović, "Two Faces of the Framework for Analysis and Prediction, Part 1 - Education," *Inf. Technol. Control*, vol. 47, no. 2, pp. 249–261, Jun. 2018.