Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 46 (2015) 457 – 462

Procedia Computer Science

International Conference on Information and Communication Technologies (ICICT 2014)

ARS*k*NN-A *k*-NN Classifier Using Mass Based Similarity Measure

Ashish Kumar[a,]*, Roheet Bhatnagar[a], Sumit Srivastava[a]

[a]*Manipal University Jaipur, Dehmi Kalan, Jaipur, 303007, India*

Abstract

k-Nearest Neighbor (*k*-NN) classification technique is one of the most elementary and straightforward classification methods. Although distance learning is in the core of *k*-NN classification, similarity can be preferred upon distance in several practical applications. This paper proposes a novel algorithm for learning a class of an instance based on a similarity measure which does not calculate distance, for *k*-Nearest Neighbor (*k*-NN) classification.

Peer-review under responsibility of organizing committee of the International Conference on Information and Communication Technologies (ICICT 2014)

Keywords: Classification Algorithm; Data Mining; Nearest Neighbor; Similarity Measures;

1. Introduction

Classification techniques is a domain of much enthusiasm toward established researchers, due to their wide range of applications, viz; handwriting recognition, biometric identification, computer-aided diagnosis and many more. Among so many classification techniques, the conventional *k*-nearest Neighbor (*k*-NN) classification technique is one of the simplest and oldest. In this technique, *k* nearest neighbors to the queried instance are found and then the class of the queried instance is determined by a majority rule i.e. the class which is most present in the set of nearest neighbor[1]. Till now the *k*-NN technique is studied and used by many researchers from many communities in their research works. The accuracy of *k*-NN classifiers is significantly dependent on the metric used to calculate the distances / similarities between the queried instance and the instances with labelled class in the dataset[2]. Most of the work on learning for *k*-NN classification is based on distance metrics[2,3,4,5]. However, in several practical situations e.g. text classification, similarities (e.g. cosine similarity) could be favoured over distances (e.g. Euclidean distance).

* Corresponding author. Tel.: +91-823-324-4583.
E-mail address: aishshub@gmail.com

Peer-review under responsibility of organizing committee of the International Conference on Information and Communication Technologies (ICICT 2014)
doi:10.1016/j.procs.2015.02.044

Furthermore, several research works have shown that the use of similarity measure can be preferred over the distance measure in non textual datasets (like Balance, Wine and Iris) also[6]. The selection of similarity/distance measure depends upon the measurement type or representation of objects to be classified. Till recent, before the introduction of the concept of mass, most of the similarity measures directly or indirectly are depended on distance measures. Currently introduced mass based similarity measure, named as Massim, is an alternative to distance-based similarity measure and has shown very good results in information retrieval[7]. It is focused around the cardinality of the smallest neighborhood area covering the two instances for which it is measured. This paper proposes to use this mass based similarity measure, Massim, in *k*-NN classification technique as metric learning.

The paper is organized into 4 sections. Section 1 is the introduction and gives an overview of classification techniques. Section 2 discusses the related works on *k*-Nearest Neighbor and mass based similarity measure, 'Massim'. This section also explains the working principle of Massim. Section 3 introduces ARS*k*NN classifier which uses Massim as its core similarity learning among two instances. The paper concludes in section 4 and also lists future work to be carried out.

2. Related Works

This section discusses about the *k*-NN classifiers, similarity measures and then about Massim which is newly developed mass based similarity measure. Working principle of Massim is also discussed in this section.

2.1. k-Nearest Neighbor (k-NN) Classification

The conventional *k*-nearest neighbor[1] is extended from Nearest Neighbor (NN) classifier and used the *k*-NN rule proposed by Fix and Hodges in 1951. It is non parametric and lazy learning classification technique. It is also known by different names like memory based classification, instance based classification, case based classification and many more. There are three key elements in *k*-NN classifier:

- A set of labeled instances (e.g., a set of stored records)
- A distance or similarity metric to compute distance between instances.
- The value of *k*, the number of nearest neighbors.

A number of metrics have been proposed to improve the *k*-NN classifiers such as Euclidian distance, adaptive distance[8], Mahalanobis distance metric[2] and local metric[9] and the conventional *k*-NN classifier also has seen so many extensions and improvements till now. For example, nearest feature line (NFL) classifier[10], nearest feature plane (NFP) and nearest feature space (NFS) classifiers[11], center-based nearest neighbor classifier (CBNNC)[12], Nearest neighbor line classifier (NNLC)[13], meaningful nearest neighbor (MNN)[14], distance lower bound based nearest neighbor[5], MaxNearestDist based nearest neighbor[15], and probably correct *k*-nearest neighbor (PCKN)[16] have also been proposed. All these are either direct or indirect classifiers based on dissimilarity (distance) or similarity metrics.

Till now none of the *k*-NN classifier is based on mass based similarity metrics and this paper proposes a novel algorithm for the same.

2.2. Similarity Measure

Similarity is a concept which is not only used in data mining but also in psychology and ecology. A similarity measure is a real-valued function that quantifies similarity between two instances (objects). The value is higher when instances are more similar. Similarity measures are also known as similarity coefficients. Various similarity and distance measures have been proposed in different fields over a hundred years, after Jaccard proposed a similarity measure to classify biological species in 1901. A survey of 76 binary similarity and distance measures along with their correlations through hierarchical clustering technique is done by Seung-Seok Choi et. al[17]. Literature shows that [18] and [19] have done a comprehensive work on similarity measures. All the existing distance based similarity measures are binary function and they fulfill all four distance axioms (reflexivity, non-negativity, symmetry, triangle inequality). However, some researchers challenged the triangle inequality axiom by developing

non metric similarity measures[20]. The main drawback with distance based similarity measures is that they are computationally costly in huge datasets because of their high time complexities.

2.3. Massim - A Mass Based Similarity Measure

Massim is very much fundamentally different from all the similarity measures because it is based on mass[21], which is cardinality of the region, rather than distance. The concept of mass and mass estimation is introduced by Kai Ming Ting et. al.[22] in 2010. It is used as an alternative of density estimation to solve various data mining problems. Mass itself is a unary function but Massim based on mass is a binary function to measure similarity between two instances. Table 1 show the comparison between mass based and distance based similarity measures. Unlike distance-based similarity measure, Massim not satisfy all four distance axioms, as given in Table 2. It principally based on data distribution in local region and does not calculate distance, which is core calculation of distance based similarity measures. Massim gives us guarantee that two similar instances must be in same local neighborhood.

Table 1. Mass-based Similarity Measure versus Distance-based Similarity Measure[7]

	Mass-based similarity measure	**Distance-based similarity measure**
Computation	Mass(x, y) is primarily based on data distribution in the local region of the feature space.	dist(x, y) is solely based on the positions of x and y in the feature space.
Definition	Mass base function M(x, y) measures the cardinality of the smallest local region covering both x and y.	dist(x, y) measures the length of the shortest path from x to y.
Inequality	similarity(x, y) > similarity(x, z) ≡ Mass(x, y) < Mass(x, z)	similarity(x, y) > similarity(x, z) ≡ dist(x, y) < dist(x, z)
Metric	The measure does not satisfy some distance axiom.	All distance axioms usually hold.

Table 2. Axioms used for Mass-based Similarity and Distance-based Similarity[7]

	Mass-based Similarity	**Distance-based Similarity**	
Axiom 1	Mass(x, y) ≥ 1	dist(x, y) ≥ 0	(non-negativity)
Axiom 2	i. ∀x, y Mass(x, x) ≤ Mass(x, y) ii. ∃x ≠ y Mass(x, x) ≠ Mass(y, y)	dist(x, y) = 0 ⇔ x = y	(identity of indiscernibles)
Axiom 3	Mass(x, y) = Mass(y, x)	dist(x, y) = dist(y, x)	(symmetry)
Axiom 4	Mass(x, z) < Mass(x, y) + Mass(y, z)	dist(x, z) ≤ dist(x, y) + dist(y, z)	(triangle inequality)

2.4. Working principle of Massim

There are two versions of Massim: One dimensional and Multi-dimensional. For one dimensional, the idea is to generate a model E, which partitions the feature space into the smallest convex local region R(x, y | E), which covers x, y∈ℝ, as a result of binary splits, which divides the real line defined by D={x1, < x2, < x3, …, <xn} into two non-overlapping local regions r1 (where x ≤s_i) and r2 (where x ≥s_i); and r3, which covers entire real line. Now Massim Mass(x, y) is defined as follows:

$$Mass(x, y) = \left(\sum_{i=1}^{n-1} M_i(x, y)^e p(s_i)\right)^{\frac{1}{e}} \quad (1)$$

where, e is mean either harmonic or arithmetic, p(s_i) is the probability of selection of the binary split (s_i) and

$$M_i(x,y) = \begin{cases} i & if\ R(x,y \mid E) = \mathrm{r}_1 \\ n-1 & if\ R(x,y \mid E) = \mathrm{r}_2 \\ n & if\ R(x,y \mid E) = \mathrm{r}_3 \end{cases}$$

Another smoother one dimensional Massim, which is known as Level-h Massim, is calculated recursively as follows:

$$\mathrm{Mass}^{\mathrm{h}}(\mathrm{x},\mathrm{y}) = (\textstyle\sum_{\mathrm{i}=1}^{\mathrm{n}-1} \mathrm{Mass}_{\mathrm{i}}^{\mathrm{h}-1}(\mathrm{x},\mathrm{y})^{\mathrm{e}}\mathrm{p}(\mathrm{s}_{\mathrm{i}}))^{\frac{1}{\mathrm{e}}} \quad (2)$$

with the condition,

$$\mathrm{Mass}^{\mathrm{h}}(\mathrm{x},\mathrm{y}) = \begin{cases} \sum_{\mathrm{i}=1}^{\mathrm{n}-1} \mathrm{m}_{\mathrm{i}}(\mathrm{x})\mathrm{p}(\mathrm{s}_{\mathrm{i}}), & \mathrm{h}=1 \\ \sum_{\mathrm{i}=1}^{\mathrm{n}-1} \mathrm{mass}_{\mathrm{i}}^{\mathrm{h}-1}(\mathrm{x})\mathrm{p}(\mathrm{s}_{\mathrm{i}}), & \mathrm{h}>1 \end{cases}$$

where,

$$\mathrm{m}_{\mathrm{i}}(\mathrm{x}) = \begin{cases} \mathrm{i} & \text{if } \mathrm{x} \le \mathrm{x}_{\mathrm{i}} \\ \mathrm{n}-1 & \text{if } \mathrm{x} \ge \mathrm{x}_{\mathrm{i}} \end{cases}$$

Level-h Massim considers the way of data distribution more effectively and is used for multi-modal data distribution[7].

In multi-dimensional Massim, the similarity of two instances (x,y) is calculated by normal averaging the mass of all smallest local regions which wrap both instances (x,y) and these random local regions those assure the concept of mass inequality, are generated using half-space splits, as proposed by Kai Min Ting et al.[21]. In a multi-dimensional feature space, every half-space split is performed on a randomly chosen attribute. For an h-level split, a tree structure is formed in which each path has h half-space splits, and the tree has a total of 2^h non-overlapping local regions[7].

The multi-dimensional mass similarity is calculated as follows:

$$Mass^h(\mathbf{x},\mathbf{y}) \approx (\tfrac{1}{t}\textstyle\sum_{i=1}^{t} \mathfrak{M}_{h,i}(\mathbf{x},\mathbf{y})^e)^{\frac{1}{e}} \quad (3)$$

where, $t > 1$ is the number of random chosen regions to be used to describe the mass similarity of x and y; $\mathfrak{M}_{\mathrm{h,i}}(\mathrm{x},\mathrm{y})$ is the mass in $\mathrm{R}(\mathrm{x},\mathrm{y}|\mathrm{E}_{\mathrm{i}}(\mathrm{h}))$, which is the smallest region of various mutli-dimensional models $\mathrm{E}_{\mathrm{i}}(\mathrm{h})$ covering both x and y. For the implementation part of mass-based similarity estimation please do refer to[7].

3. ARSkNN - Mass based *k*-Nearest Neighbor

In this section, we propose a novel *k*-nearest neighbor classifier which uses mass-based similarity measure rather than distance-based similarity measures.

ARS*k*NN has got two stages: 1. Modeling Stage and 2. Class Assignment Stage. In modeling (preprocessing) stage, a Similarity Forest (sForest) with t number of Similarity Trees (sTrees), is built from D dataset which has $\{(\mathrm{x}_1,\mathrm{c}_1),(\mathrm{x}_2,\mathrm{c}_2),\dots,(\mathrm{x}_{\mathrm{n}},\mathrm{c}_{\mathrm{n}})\}$ without consideration of c_{i} as described in[7]. After this in class assignment stage, ARS*k*NN is used to find the *k*-nearest neighbors (instances) in D with respect to a query instance.

For a query instance y, $\mathrm{Mass}^{\mathrm{h}}(\mathrm{x},\mathrm{y})$ is calculated for all instances x in dataset D. The nearest neighbor of y is found as follows: $\forall\ \mathrm{x}_{\mathrm{i}},\mathrm{x}_{\mathrm{j}} \in \mathrm{D}$, if $\mathrm{Mass}^{\mathrm{h}}(\mathrm{x}_{\mathrm{i}},\mathrm{y}) < \mathrm{Mass}^{\mathrm{h}}(\mathrm{x}_{\mathrm{j}},\mathrm{y})$ then similarity $(\mathrm{x}_{\mathrm{i}},\mathrm{y}) > similarity\ (\mathrm{x}_{\mathrm{j}},\mathrm{y})$. It means the instance having less mass with respect to query instance is more similar to the query instance. Now find the k nearest neighbor of the y and then decide the class of y in voted manner. For the algorithm $\mathrm{Mass}(\mathrm{x}_{\mathrm{i}},\mathrm{y},\mathrm{F},\mathrm{e})$ and similarity forest (F), we would like to refer the reader to[7].

Algorithm 1: ARSkNN (y, D, k)
Input:y – Query instance, D – Dataset which has $\{(x_1, c_1), (x_2, c_2), \dots, (x_n, c_n)\}$, k - number of nearest neighbor, Output:c_y – Class of query instance y 1: Let A ← { } 2: for i = 1 → n do 3: Mass ← Mass (x_i, y, F, e) 4: A ← A ∪ $\{x_i, c_i, \text{Mass}\}$ 5: end for 6: Sort in ascending order, the pairs in U using third components 7: c_y ← the most frequent class in [Select the first k instances from U] 9: returnc_y

The time complexity of ARS*k*NN in modeling stage is O(ntlog(d)) and in class assignment stage is O(*k*n) where, n= number of total instances in dataset; d= number of randomly selected instances from dataset; t= number of random regions to be used to define mass similarity; *k*= number of nearest neighbor.

In modeling stage, we can reduce the time complexity from O(ntlog(d)) to O((n+d)t) by using indexing technique because all the sTreees of sForest are balanced binary tree.

4. Conclusion and Future Work

This paper established the theoretical foundation of ARS*k*NN which is a novel approach of *k*-NN classifiers. The key drawback of traditional *k*-NN is that it needs to keep all the instances of dataset into memory and due to this drawback; one cannot use traditional *k*-NN classifier on big data. But in ARS*k*NN, there is no need to keep all the instances of dataset in memory because in the preprocessing stage, it is developing a sForest from sample instances of the dataset.

The simulation and testing of the algorithm on different datasets is considered for future work and research.

References

1 Cover T, Hart P, Nearest Neighbor Pattern Classification, *IEEE Trans. Information Theory*, 1967, **13**:1, p. 21-27.

2 Weinberger KQ, Lawrence KS, Distance Metric Learning For Large Margin Nearest Neighbor Classification, *The Journal of Machine Learning Research,* 2009, **10**, p. 207-244.

3 Shalev-Shwartz S, Singer Y, Ng. AY, Online And Batch Learning of Pseudo-Metrics, *Proc. twenty-first International conf. Machine learning ACM*, July 2004, p. 94.

4 Baoli L, Qin L, Shiwen Y, An Adaptive K-Nearest Neighbor Text Categorization Strategy, *ACM Trans. Asian Language Information Processing (TALIP)*, 2004, **3:**4, p. 215-226.

5 Chen YS, Hung YP, Yen TF, Fuh CS, Fast And Versatile Algorithm For Nearest Neighbor Search Based On A Lower Bound Tree, *Pattern Recognition*, 2007, **40**:2, p. 360-375.

6 Qamar AM, Gaussier E, Chevallet JP, Lim JH, Similarity Learning For Nearest Neighbor Classification, *Proc. Eighth IEEE International Conf. Data Mining (ICDM'08)*, 2008, p. 983-988.

7 Ting KM, Fernando TL, Webb GI, Mass-based Similarity Measure: An Effective Alternative to Distance-based Similarity Measures, Technical Report 2013/276, 2013, Calyton School of IT, Monash University, Australia.

8 Wang J, Neskovic P, Cooper NL, Improving Nearest Neighbor Rule with a Simple Adaptive Distance Measure, *Pattern Recognition Letters*, 2007, **28**:2, p. 207-213.

9 Noh YK, Zhang BT, Lee DD, Generative Local Metric Learning for Nearest Neighbor Classification, *In NIPS*, 2010, p. 1822-1830.

10 Li SZ, Lu J, Face Recognition Using The Nearest Feature Line Method, *IEEE Trans. Neural Networks*, 1999, **10**:2, p. 439-443.

11 Chien JT, Wu CC, Discriminant Waveletfaces and Nearest Feature Classifiers for Face Recognition, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2002, **24**:12, p. 1644-1649.

12 Gao QB, Wang ZZ, Center-Based Nearest Neighbor Classifier, *Pattern Recognition*, 2007, **40**:1, p. 346-349.

13 Zheng W, Zhao L, Zou C, Locally Nearest Neighbor Classifiers for Pattern Classification, *Pattern Recognition*, 2004, **37**:6, p. 1307-1309.

14 Omercevic D, Drbohlav O, Leonardis A, High-Dimensional Feature Matching: Employing the Concept of Meaningful Nearest Neighbors, *Proc. IEEE eleventh International Conf. Computer Vision (ICCV 2007),* Oct. 2007, p. 1-8.
15 Samet H, K-Nearest Neighbor Finding using MaxNearestDist, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2008, **30**:2, p. 243-252.
16 Toyama J, Kudo M, Imai H, Probably Correct K-Nearest Neighbor Search In High Dimensions, *Pattern Recognition*, 2010, **43**:4, p. 1361-1372.
17 Choi SS, Cha SH, Tappert CC, A Survey of Binary Similarity and Distance Measures, *Journal of Systemics, Cybernetics and Informatics*, 2010, **8**:1, p. 43-48.
18 Zezula P, Amato G, Dohnal V, Batko M, Similarity Search: The Metric Space Approach, *Springer*, 2006, **32**.
19 Cha SH, Comprehensive Survey on Distance/Similarity Measures Between Probability Density Functions, *City*, 2007, **1**:2, 1.
20 Skopal T, Bustos B, On Nonmetric Similarity Search Problems in Complex Domains, *ACM Computing Surveys (CSUR)*, 2011, **43**:4, p. 34.
21 Ting KM, Zhou GT, Liu FT, Tan SC, Mass Estimation, *Machine learning*, 2013, **90**:1, p.127-160.
22 Ting KM, Zhou GT, Liu FT, Tan SC, Mass Estimation and its Applications, *Proc. sixteenth ACM SIGKDD international conference on Knowledge discovery and data mining*, July 2010, p. 989-998.