Research Article

Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets

Najat Ali[1] · Daniel Neagu[1] · Paul Trundle[1]

Received: 13 June 2019 / Accepted: 24 September 2019 / Published online: 6 November 2019

Abstract
Distance-based algorithms are widely used for data classification problems. The k-nearest neighbour classification (k-NN) is one of the most popular distance-based algorithms. This classification is based on measuring the distances between the test sample and the training samples to determine the final classification output. The traditional k-NN classifier works naturally with numerical data. The main objective of this paper is to investigate the performance of k-NN on heterogeneous datasets, where data can be described as a mixture of numerical and categorical features. For the sake of simplicity, this work considers only one type of categorical data, which is binary data. In this paper, several similarity measures have been defined based on a combination between well-known distances for both numerical and binary data, and to investigate k-NN performances for classifying such heterogeneous data sets. The experiments used six heterogeneous datasets from different domains and two categories of measures. Experimental results showed that the proposed measures performed better for heterogeneous data than Euclidean distance, and that the challenges raised by the nature of heterogeneous data need personalised similarity measures adapted to the data characteristics.

Keywords k-nearest neighbour · Heterogeneous data set · Combination similarity measures

1 Introduction

Classification is a supervised machine learning process that maps input data into predefined groups or classes [1]. The main condition for applying a classification technique is that all data objects should be assigned to classes, and that each of the data objects should be assigned to only one class [2].

Distance-based classification algorithms are techniques used for classifying data objects by computing the distance between the test sample and all training samples using a distance function. Distance-based algorithms though were originally proposed to deal with one type of data using distance-based measurements to determine the similarity between data objects. These algorithms were subsequently developed to enable handling of heterogeneous data as real-world data sets are often diverse in types, format, content and quality, particularly when they are gathered from different sources.

In general, when classifying heterogeneous data using distance-based algorithms, there are two categories of methods. The first category converts values from one data type to another (e.g. binning data, interpolating or projecting data) and then, distance-based algorithms can be used with an appropriate measurement to classify the data.

However, this method is not effective as the similarity measure of the transformed data does not necessarily represent consistently the similarity of the original heterogeneous data, especially when the transformation is not fully reversible. Moreover, the data conversion could also fundamentally alter values to make them more equidistant, meaning there are no guarantees that data will be

✉ Najat Ali, Nali50@bradford.ac.uk; Daniel Neagu, D.Neagu@Bradford.ac.uk; Paul Trundle, P.R.Trundle@Bradford.ac.uk | [1]Faculty of Engineering and Informatics, University of Bradford, Bradford BD7 1DP, UK.

interpreted correctly, which introduces the risk of losing or altering vital information in the process of decision the classification task is designed to support.

The second category extends distance-based algorithms to match the heterogeneous data. This can be done using a distance measures that can handle heterogeneous data.

One common classification technique based on the use of distance measures is k-nearest neighbours (k-NN) [3]. The traditional k-NN classification algorithm finds the k-nearest neighbour(s) and classifies numerical data records by calculating the distance between the test sample and all training samples using the Euclidian distance [4].

The primary focus of the k-NN classifier has been on data sets with pure numerical features [5]. However, k-NN can also be applied to other type of data includes categorical data [6]. Several investigations have been done to find a proper categorical measures for such data, such as the works presented in [7–12].

Moreover, it also can be applied to classify data described by numerical and categorical features such as studies reported in [7, 13].

This paper aims to investigate the performance of k-NN classification on heterogeneous data sets using two types of measures:the well-known (Euclidean and Manhattan) distances and the combination of similarity measures that are formed by fusing existing numerical distances with binary data distances. It also aims to provide a first attempt of guidance as to the best combination of similarity function that can be used with k-NN for heterogeneous data classification (of numerical and binary features). The rest of this paper is organised as follows. Section 2 provides the concepts, background and literature review relevant for the research topic. Section 3 briefly describes the six well-known distance functions that are used in this study and explains the proposed technique for classifying heterogeneous data. Section 4 presents the experimental work and results. Finally, Sect. 5 presents the conclusion and future work.

2 Background

2.1 Distance and similarity measures

The concept of similarity between data objects is widely used across many domains to solve a variety of pattern recognition problems such as categorisation, classification, clustering and forecast [14]. Various measures have been proposed in the literature for comparing data objects [15]. In this section the concepts of distance measure, similarity measure are introduced, followed by a review of the k-NN algorithm and its performance evaluation.

Definition 1 A distance measure $d : X \times X \rightarrow R$ is a function called metric if it satisfies the following requirements [16] $\forall x, y, z \in X$:

1. $0 \leq d(x, y)$ (*Non-negative*);
2. $d(x, y) = 1$,if and only if $x = y$ (Identity);
3. $d(x, y) = d(y, x)$ (Symmetry);
4. $d(x, z) \leq d(x, y) + d(y, z)$ (Triangle inequality).

However, similarity measurement shows more debates, as it provides some flexibility in the identification of how close two data objects could be. Similarity measure is generally perceived as complementary to a distance measure.

Definition 2 similarity measure $S : X \times X \rightarrow R$ is a function that satisfies the following requirements $\forall x, y \in X$:

1. $0 \leq S(x, y)$ (*Non-negative*);
2. $S(x, y) = 1$,if and only if $x = y$ (Identity);
3. $S(x, y) = S(y, x)$ (Symmetry).

2.2 K-nearest neighbour classifier (k-NN)

In this section, we look at the classification that uses the concept of distance for classifying data objects. The k-NN classifier is one of the simplest and most widely used in such classification algorithms. k-NN was proposed in 1951 by Fix and Hodges [17] and modified by Cover and Hart [3]. The technique can be used for both classification and regression [18].

The main concept for k-NN depends on calculating the distances between the tested, and the training data samples in order to identify its nearest neighbours. The tested sample is then simply assigned to the class of its nearest neighbour [19].

In k-NN, the k value represents the number of nearest neighbours. This value is the core deciding factor for this classifier due to the k-value deciding how many neighbours influence the classification. When $k = 1$then the new data object is simply assigned to the class of its nearest neighbour. The neighbours are taken from a set of training data objects for where the correct classification is already known. k-NN works naturally with numerical data. Various numerical measures have been used such as Euclidean, Manhattan, Minkowsky, City-block, and Chebyshev distances. Amongst these, the Euclidean is the most widely used distance function with k-NN [20]. The main steps of k-NN algorithm in Fig. 1 are:

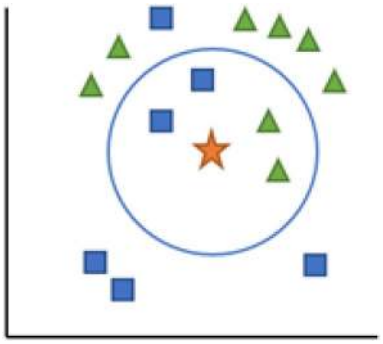

Fig. 1 k-Nearest neighbour classification (k = 4)

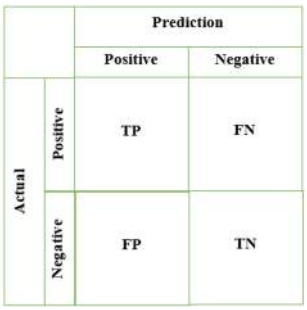

Fig. 2 A confusion matrix for binary classification

1. Determine the number of nearest neighbours (K values).
2. Compute the distance between test sample and all the training samples.
3. Sort the distance and determine nearest neighbours based on the K-th minimum distance.
4. Assemble the categories of the nearest neighbours.
5. Utilise simple majority of the category of nearest neighbours as the prediction value of the new data object.

According to [21], the k-NN classifier can be used to classify new data objects using only their distance to labelled samples. However, some works consider any metric or non-metric measures used with this classifier: several studies have been conducted to evaluate the k-NN classifier using different metric and non-metric measures such as the studies presented in [7, 10, 22–26].

2.3 Performance metrics for classification

The most widely used technique for summarizing the performance of a classification algorithm is the Confusion Matrix. Figure 2 shows the confusion matrix for the case of binary classification with the following elements:

Table 1 Evaluation measures for binary class data set

Measure	Formula
Accuracy	$= \frac{TP+TN}{TP+FP+FN+TN}$
Precision	$= \frac{TP}{TP+FP}$
Recall	$= \frac{TP}{TP+FN}$
F-score	$= \frac{2(Recall \times Precision)}{(Recall+Precision)}$

Table 2 Evaluation measures for multi class data set

Measure	Formula
Accuracy	$= \frac{\sum_{i=1}^{n} \frac{TP_i+TN_i}{TP_i+FP_i+FN_i+TN_i}}{n}$
Precision	$= \frac{\sum_{i=1}^{n} \frac{TP_i}{TP_i+FP_i}}{n}$
Recall	$= \frac{\sum_{i=1}^{n} \frac{TP_i}{TP_i+FN_i}}{n}$
F-score	$= 2\frac{(Recall \times Precision)}{(Recall+Precision)}$

1. *True Positives (TP)* is defined by the total number of accurate outputs when the actual class of the data object was True and the prediction was also the True value.
2. *True Negatives (TN)* is defined by the total number of accurate outputs when the actual class of the data object was False and the predicted is also the False value.
3. *False Positives (FP)* when the actual class of the data object was False and the output value was the True value
4. *False Negatives (FN)* when the actual class of the data object was True and the output value was the False value.

2.3.1 Metrics computed from a confusion matrix

A confusion matrix gives a useful information about how well the model does. However, its elements can be used to calculate many performance metrics to get even more information. Among the most popular are (see also Tables 1, 2):

1 *Accuracy* is the most intuitive performance measure, and defined as the ratio of the number of correctly classified objects to the total number of objects evaluated.

2 *Precision* it is simply a ratio of correctly predicted positive data objects to the total predicted positive data objects.

3 *Recall* it is defined by the number of correct positive results divided by the total number of relevant samples (all samples that should have been identified as positive).
4 *F-score* it can be defined as a weighted average of the precision and recall. An F-score is considered perfect when reaches its best value at 1, while the model is a total failure when it reaches the 0 value.

Tables 1 and 2 show the evaluation measures for binary and multi-class dataset respectively.

2.4 Related work

As we mentioned earlier, plenty of studies investigated, analysed, and evaluated the performance of k-NN on pure numerical and pure categorical data sets. Regarding applying k-NN to heterogeneous data described by numerical and categorical features, the most widely used method is to treat the data before feeding to the classifier. This can be done by converting non-numerical features into numerical features using different techniques, and then the traditional k-NN can be applied with any numerical distance.

A study presented by Hu et al. [7] evaluated the performance of k-NN on three types of medical data sets, pure numerical, pure categorical, and mixed data using different numeric measures. They treat non-numerical features by encoding them as binary. Similar technique also has been applied in some studies such as [8, 13, 27].

On the other hand, studies have used the combination approach for classifying heterogeneous data using k-NN. Such study presented by Pereira et al. [28] have proposed a new measure for computing the distance between heterogeneous data objects and used this measure with k-NN. This distance is called Heterogeneous Centered Distance Measure (HCDM). It is based on a combination of two techniques: Nearest Neighbour Classifier (CNND) distance for numerical features and Value Difference Metric (VDM) with k-NN for classifying heterogeneous data sets, described by two different features type; numerical and categorical. The combination measures include:

Heterogeneous Euclidean-Overlap Metric (HEOM), which uses the overlap metric for categorical features and the normalized Euclidean distance for numerical features; Heterogeneous Manhattan-Overlap Metric (HMOM), which uses the overlap metric for categorical features and Manhattan distance for numerical features; Heterogeneous Distance Function (HVDM) which uses the Value Difference Metric (VDM) for categorical features and the normalized Euclidean distance for numerical features

In [29], Deekshatulu et al. have proposed a new classification algorithm which combines k-NN and genetic algorithm, to predict heart disease of a patient for Andhra Pradesh population. The authors also have applied the model to medical data and non-medical data sets such as Hypothyroid, liver disorder, primary tumour, and Weather data sets. In this model the features are ranked based on their value. The least ranked features are removed, and the classification algorithm is built based on evaluated features. Generally, the most commonly used approaches for classifying heterogeneous data by k-NN classifier can be described as a mixture of numerical and categorical features which include:

1. *Conversion approach* a method of converting the data set into a single data type, and then applying appropriate distance measures to the transformed data.
2. *Unified approach* a method to integrate two or more different measures to infer the overall value.

3 Measures for comparing data objects

As we mentioned in the previous section, a combination approach is one of the most widely used methods for comparing data objects described by a mixture of data types. The simple idea of applying this technique for calculating the similarity between two data objects described by a mixture of features is to split these features into subsets based on their data type and then to identify the similarity between the subsets of same type. The next step is to combine these measures to obtain a single value representing the similarity between two data objects. In this study, we have used the combination approach to generate a number of similarity measures based on the existing measures to handle heterogeneous data when the representation of the data includes a mixture of numerical and binary features. The data is first divided into pure numerical and pure binary features, specific distances are then applied to the numerical and binary features, and the result of the two distances is assembled into one single distance using a weighted average to form the combined distance value.

3.1 Measures for numerical data

In [30] Cha categorized the numerical distances into eight distance families. The study presented by Prasath et al. [23], classified the distance measures following a similar classification done by Cha. Their study also evaluated the performance (measured by accuracy, precision and recall) of the k-NN with the classified distance families for classifying numerical data.

In this study, we will investigate the performance of k-NN for classifying heterogeneous data by using

measures from three different families. We have chosen the most representative measures from these families, as they have been applied with k-NN in different studies for classifying the data and represent good references for critical comparisons of results reported hereby. The five chosen measures belong to the following families:

1. L_p *Minkowski family* it is also known as the p-norm distance. The chosen measures from this family include:

   (i) Manhattan distance is defined by:

   $$d(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{1}$$

   (ii) Euclidean distance is defined by:

   $$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2}$$

2. *Inner product family* distance measures belonging to this family are calculated by some products of pair wise values from both vectors. Two measures have been selected from this family:

   (i) Cosine similarity measure is defined by:

   $$S(x,y) = \frac{x.y}{||x||y||} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x^2{}_i}\sqrt{\sum_{i=1}^{n} y^2{}_i}} \tag{3}$$

   (ii) Jaccard distance is defined by:

   $$d(x,y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{(x_i)^2 + (y_i)^2 - [(x_i)(y_i)]} \tag{4}$$

3. L_1 *distance family* the distances in this family are calculated based on finding the absolute difference. Only one measure have been chosen from this family:

   (i) Canberra distance is defined by:

   $$d(x,y) = \sum \frac{|x_i - y_i|}{|x_i| + |y_i|} \tag{5}$$

As we mentioned in this section, the chosen measures have been widely applied with k-NN for classifying the datasets in the selected case studies presented in [7, 22, 26, 31–33]. Most the equations are confirmed metrics: Euclidean, Manhattan, Canberra according to [34, 35], and Jaccard according to [36], satisfy the conditions in Definition 1. Cosine measure is not metric. It does not satisfy condition 4 in Definition 1.

3.2 Measures for categorical data

Generally, categorical data is classified as a type of qualitative data [37]. Such data corresponds to a possible representation for nominal, binary, ordinal, and interval instances. For the sake of simplicity, in this work, we will focus on only one type of categorical data which is binary data.

The set of measures developed for dealing with binary data is known as matching coefficients [38]. They calculate the distance between two data objects x and y defined as $x = \{x_1, x_2, \ldots, x_p\}$, and $y = \{y_1, y_2, \ldots, y_p\}$, where p represents the number of binary features in each data object.

The strategy behind these methods is that the two data objects are viewed as similar to the degree that they share a common pattern of feature values among the binary variables. The matching coefficient values range between 0 for not similar at all and 1 for completely similar [39]. Figure 3 shows the main four quantities of binary features.

Any binary feature has only one of two cases: 0 means that the feature is absent and 1 means that the feature is present, this is called symmetric binary features [39]. Those are listed below :

(i) a represents the total number of features in both x and y have a value of 1.
(ii) b represents the total number of features where the feature of x is 0 and y is 1.
(iii) c represents the total number of features where x is 1 and y is 0.
(iv) d represents the total number of features in both x and y have a value of 0.

Each feature in data objects must belong to one of these four categories a, b, c, and d, and $a + b + c + d = p$, where p is the total number of binary features. There are various similarity measures for binary data proposed in the literature.

In [40], Choi et al. has compared 76 binary similarity measures and classified them hierarchically to observe close relationships among them.

	Presence of x_i	Absence of x_i	Sum
Presence of y_i	a	c	a+c
Absence of y_i	b	d	b+d
Sum	a+b	c+d	P=a+b+c+d

Fig. 3 The main four quantities of binary features to compare two m-dimensional objects classification

The overlap similarity measure is widely used in data mining tasks such as clustering, classification, and regression for handling binary data. It is also known as a simple matching similarity measure. The overlap similarity measure determines by the number of corresponding features that have identical values. The measure is defined by:

$$s(x,y) = \frac{a+d}{p} \tag{6}$$

Researchers in different studies have also applied the overlap measure with k-NN for both classification and regression tasks. They used overlap measure for comparing categorical (nominal/ binary) data such as studies presented in [32, 41, 42].

However, the main limitation of this measure is that this measure only determines whether the features are match to one another (a and d), and does not make full use of the rest of the classification information. Therefore, in this study, Jaccard coefficient similarity measure is adopted to deal with binary data and is defined as:

$$s(x,y) = \frac{a}{a+b+c} \tag{7}$$

It should be noted that Jaccard coefficient similarity measure excludes d from consideration which represents joint absences for both features. According to [43], the d value in Fig. 3 does not necessarily represent resemblance between data objects, since a large proportion of the binary dimensions in two data objects are more likely to have negative matches.

On the other hand, the study presented by Faith et al. [44] considered d value in the calculation of comparing binary data. However, their studies showed that positive matches as more considerable, therefore they give the former less weight comparing to the negative matches.

3.3 Similarity measures for objects described by heterogeneous features

Many aggregation operators were used to aggregate the values obtained through multiple similarity measures for data mining applications such as clustering and classification. Plenty studies have introduced such aggregated similarity measures [45–47]. This includes measures for different types of data such as classical data (numerical and categorical), fuzzy data, and intuitionistic Fuzzy data or even the combination between them. Some of these studies include study presented by Bashan et al. [48] have introduced a classical similarity measure called weighed average similarity measure. It is based on the combination between numerical and categorical similarities. They also introduced weighed average similarity measure based on the combination between classical and fuzzy similarities for comparing heterogeneous data sets. Another study [46] have proposed the weighted average similarity measure between intervals of linguistic 2-tuples for solving fuzzy group decision making issue. Studies presented in [49, 50] have also proposed weighted average similarity measures for Intuitionistic Fuzzy data. The proposed measures are applied to various pattern recognition problems.

Actually, this approach already existed in other machine learning algorithms: for example in random forest [51] when trained on the subsets, the weights are calculated according to the global outputs.

In this work, we used the weighted average methods for giving the weights to numerical and binary similarities that will be used with k-NN for classifying heterogeneous features.

The weighted average of set of values $x_1, x_2, \ldots, x_n$ with corresponding weights $w_1, w_2, \ldots w_n$ is computing from the following formula:

$$\bar{x}_w = \frac{w_1x_1 + w_2x_2 + \cdots + w_nx_n}{w_1 + w_2 + \cdots + w_n} \tag{8}$$

where $w_1, w_2, \ldots w_n > 0$. It should be noted that if $w_1 + w_2 + \cdots + w_n = 1$then:

$$\bar{x}_w = w_1x1 + w_2x2 + \cdots + w_nxn \tag{9}$$

If $w_1 + w_2 + \cdots + w_n > 1$then Eq. 8 can be used.

Definition 3 The similarity between two data records R_1 and R_2 described as a mixture of numerical and binary features is a mapping $S : D_1 \times D_2 \longrightarrow [0, 1]$, where: D_1 demonstrates the numerical features $D_2 = R, R, R, \ldots R_z$. D_2 demonstrates the binary features $D_2 = \{0, 1\}, \{0, 1\}, \{0, 1\}, \ldots \{0, 1\}_k$ defined as:

$$S_{Het}(R_1, R_2) = \frac{w_1(S_{Num}(R_1, R_2)) + w_2(S_{Bin}(R_1, R_2))}{(w_1 + w_2)} \tag{10}$$

where S_{Num} is numerical similarity value, S_{Bin} is categorical similarity value, and w_1 and w_2 are non-negative values which can be used for giving weights for numerical and

Table 3 The combination of similarity measures based on a weighted average

The measure	The formula
M_{ej}	$S_1(R_1, R_2) = \frac{w_1(S_{Euclidean}(R_1,R_2))+w_2(S_{Jaccard_{Bin}}(R_1,R_2))}{(w_1+w_2)}$
M_{coj}	$S_2(R_1, R_2) = \frac{w_1(S_{Cosine}(R_1,R_2))+w_2(S_{Jaccard_{Bin}}(R_1,R_2))}{(w_1+w_2)}$
M_{jj}	$S_3(R_1, R_2) = \frac{w_1(S_{Jaccard_{Num}}(R_1,R_2))+w_2(S_{Jaccard_{Bin}}(R_1,R_2))}{(w_1+w_2)}$
M_{caj}	$S_4(R_1, R_2) = \frac{w_1(S_{Canberra}(R_1,R_2))+w_2(S_{Jaccard_{Bin}}(R_1,R_2))}{(w_1+w_2)}$

binary features respectively. We have introduced a list of similarity measures based on Definition 3. Table 3 shows the combination of similarity measures that have been generated based on Eq. 8 from well-known distances. These measures will be used in the next section for the experimental work.

4 Experimental analysis

This section evaluates the effectiveness of both traditional k-NN, and k-NN with the combination of similarity measurements over six heterogeneous data sets from different domains. The data sets are described by mixtures of numerical and binary features only. The characteristics of the data sets are shown in Table 4. Two data sets named Hypothyroid and Hepatitis are taken from the UCI Machine Learning Repository [52], and four data sets named Treatment, Labour training evaluation, Catsup, Azpro data sets are taken from the R packages. More description of the data sets is available in [53]. The UCI data sets have been considered after some in depth review of existing UCI benchmark data sets to satisfy the following conditions:

1. Data set should contain numerical and binary features only.
2. The data should not contain more than 3% of missing values.
3. The number of features for each type of data should be enough for calculating the similarity (not less than 2).
4. The number of classes should be small.

Both (benchmark and real) data sets types have been chosen to cover small to medium size data sets.

4.1 Data pre-processing

Before running the experiments, all datasets were pre-processed by removing irrelevant features (ID), and data objects with missing values. Numerical features were normalised to fall between 0 and 1. Each data set was split randomly into 80% for training and 20% for the testing sets.

Five k values were evaluated: 1, 3, 5, 7 and 9 neighbours. We investigated the implementation of k-NN with two different categories of measures; the first category includes Euclidean and Manhattan measures while the second category includes the four combination of similarity measures, which are described in Table 3.

It should be noted that we applied normalised Euclidean and normalised Manhattan distances to numerical datasets. Therefore, all the obtained results fall between 0 and 1. Because the similarity is complement of the distance, in this study the similarity is computed based on:

$$S(x, y) = 1 - d(x, y). \tag{11}$$

All the measures are used with the k-NN classifier individually with three different weights, and these measures are applied with k-NN to the same training and test samples each time. For evaluating the performance of k-NN we have used both accuracy (A) and F-score (F) metric. It should be noted that:

1. The values of w_1 and w_2 are set by default as following:
   (i) When the numerical features are most impotent than the binary features, we set $w_1 = 0.8$ and $w_2 = 0.6$.
   (ii) When the binary features are most impotent than the numerical features, we set $w_1 = 0.6$ and $w_2 = 0.8$.
   (iii) When the numerical and binary features have the same degree of importance, we set $w_1 = 0.5$ and $w_2 = 0.5$.
2. The values $w_1 = 0$ and $w_2 = 1$ or $w_1 = 1$ and $w_2 = 0$ are not suggested for heterogeneous data because this leads to using a single measure, negating the advantages of a combined measures.

The implementation of classifying heterogeneous data can be summarised in the following steps:

1. For each data, set the value of k, w_1 and w_2.
2. Split the data randomly into 80% for training and 20% for the test sample.
3. Apply k-NN with the measures Euclidean, Manhattan, M_{ej}, M_{coj}, M_{jj}, and M_{caj} independently to the data set.
4. Repeat steps 2 and 3 for a number of times (3 times).
5. Calculate the average of both accuracy and F-score values.

Table 4 Summary of data sets properties

Data set	#. Instances	#. Numerical features	#. Binary features	#. Classes
Hypothyroid	2643	6	14	3
Hepatitis	155	6	13	2
Treatment	2675	5	4	3
Labour training evaluation	15,992	5	4	2
Catsup	2798	4	8	4
Azpro	3589	2	3	2

4.2 Experimental results

The experimental works have been done in three stages. For each stage, the implementation steps are applied with different weight values as mentioned above. In the first stage of the experimental work, we assume that the numerical features are more important than the binary features. Tables 5, 6, 7, 8 and 9 show the results obtained by applying k-NN to six heterogeneous data sets with k = 1, 3, 5, 7, and 9 $w_1 = 0.8$ and $w_2 = 0.6$.

As it can be seen from the experiments, for traditional k-NN, the results showed that k-NN with Manhattan distance produces better results compared to the classifier with Euclidean distance for all data sets and all k values.

The experiments showed that k-NN with the combination of similarity measures performs well for classifying the six heterogeneous data sets, and outperforms k-NN with Euclidean distance. The four combination of similarity measures are efficient in handling both numerical and binary features together. However, among of them, M_{caj} performed the lowest in most cases.

Table 5 The results obtained by k-NN with all measures and K = 1, $w_1 = 0.8$ and $w_2 = 0.6$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	100	100	100	100	100	100	100	100	100	100	100	100
Hepatitis	92.31	88.71	100	100	100	100	100	100	100	100	100	100
Treatment	91.34	86.94	98.41	97.57	100	100	100	100	100	100	98.54	94.76
Labour training evaluation	88.56	81.38	92.56	88.22	94.82	89.84	94.81	88.35	95.45	89.58	90.75	86.27
Catsup	66.91	56.56	72.76	62.34	71.65	60.11	72.35	61.29	71.68	6022	69.36	59.34
Azpro	56.83	46.67	59.43	49.81	59.51	52.83	60.11	52.77	60.46	53.07	59.75	49.71

Table 6 The results obtained by k-NN with all measures and K = 3, $w_1 = 0.8$ and $w_2 = 0.6$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	92.81	88.25	95.67	91.65	95.67	92.13	96.59	91.49	96.99	92.82	94.64	88.37
Hepatitis	89.97	84.24	95.16	90.48	96.97	93.22	96.42	92.95	95.78	91.99	92.72	89.84
Treatment	87.89	82.93	91.85	85.75	92.83	86.86	90.61	85.48	90.48	86.61	89.06	83.95
Labour training evaluation	80.65	71.88	82.67	79.14	83.32	79.58	85.86	80.94	85.51	81.42	80.82	74.71
Catsup	68.23	59.57	73.68	68.41	74.63	67.48	76.13	64.57	76.87	68.65	71.89	65.75
Azpro	58.65	52.36	63.54	54.27	65.23	53.87	64.81	55.44	63.92	53.74	62.34	53.39

Table 7 The results obtained by k-NN with all measures and K = 5, $w_1 = 0.8$ and $w_2 = 0.6$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	91.78	86.58	95.81	91.55	95.36	90.54	95.14	90.09	95.39	91.86	93.41	87.24
Hepatitis	88.97	82.24	90.16	84.48	91.77	84.95	90.42	83.95	91.78	83.99	89.72	79.84
Treatment	83.45	69.81	85.12	73.89	86.83	75.64	85.56	72.47	86.90	73.66	85.69	69.58
Labour training evaluation	78.75	71.75	80.39	71.53	81.35	73.82	80.47	74.67	80.22	70.79	80.79	73.68
Catsup	67.24	58.48	70.67	66.73	70.81	67.74	71.84	63.25	72.13	64.24	71.62	61.13
Azpro	56.57	44.69	60.36	46.32	59.91	44.30	61.69	47.69	60.30	44.51	60.60	43.55

Moreover, Manhattan distance and the combination of similarity measures produce very close results.

The results also showed that the optimal number of k is 1 for Hypothyroid and Hepatitis, Treatment, and Labour training evaluation data sets. K = 3 is the optimal number for Catsup and Azpro data sets. Our results showed that some of measures outperform the others.

Table 10 shows the best measures are used with k-NN for each given k value when $w_1 = 0.8$ and $w_2 = 0.6$

Based on Table 10, it is clear that k-NN with combination of similarity measures outperform traditional k-NN.

In the second stage of the experimental work, we assume that the binary features are more important than the numerical features. Tables 11, 12, 13, 14 and 15 show the results obtained by applying k-NN to six heterogeneous data sets with k = 1, 3, 5, 7, and 9 and $w_1 = 0.6$ and $w_2 = 0.8$.

According to the results k-NN with Manhattan distance outperforms k-NN with Euclidean distance.

The obtained results showed that the optimal number is k = 1 for Hypothyroid and Hepatitis, Treatment, and Labour training evaluation data sets. K = 3 is the optimal number for Catsup and Azpro data set.

Table 8 The results obtained by k-NN with all measures and K = 7, $w_1 = 0.8$ and $w_2 = 0.6$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	88.71	81.62	90.23	83.46	89.61	83.61	92.21	85.61	90. 34	84.67	90.17	83.17
Hepatitis	86.78	77.45	88.67	82.83	88.35	82.99	87.29	80.29	87.65	80.87	84.71	79.85
Treatment	73.76	61.87	75.76	62.87	75.14	62.50	76.88	63.87	74.65	61.87	72.66	60.57
Labour training evaluation	65.09	50.45	68.23	54.83	70.85	58.91	71.62	62.56	72.22	65.23	70.79	63.49
Catsup	64.33	53.17	66.74	51.69	65. 82	50. 73	68.59	59.44	68.25	57.96	67.62	55.87
Azpro	66.57	46.60	67.36	50.70	69.91	52.56	68.69	51.69	68.30	50.78	68.60	52.89

Table 9 The results obtained by k-NN with all measures and K = 9, $w_1 = 0.8$ and $w_2 = 0.6$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	85.71	74.61	86.00	77.54	88. 45	78.56	88.62	76.83	87.64	75.78	87.65	73.80
Hepatitis	84.78	74.79	85.67	75.89	86.35	75.32	85.29	75.29	86.65	76.14	84.71	74.73
Treatment	65.45	57.25	71.82	65.56	72.03	62.47	75.18	66.62	73.64	64.21	70.45	62.36
Labour training evaluation	64.09	52.56	66.23	53.40	68.85	53.89	68.62	54.27	69.22	55.84	69.79	54.07
Catsup	57.61	45.83	60.72	49.89	60 .69	49.06	60.85	48.16	59.76	46.93	59.89	44.72
Azpro	64.39	44.76	66.48	47.78	67.59	46.80	67.00	48.32	66.73	46.59	65.39	45.42

Table 10 The best measures are used with k-NN for each given k value when $w_1 = 0.8$ and $w_2 = 0.6$

Data set	K = 1	K = 3	K = 5	K = 7	K = 9
Hypothyroid	Euclidean, Manhattan, M_{ej}, M_{coj}, M_{jj}, M_{caj}	M_{jj}	Manhattan, M_{jj}	M_{coj}	M_{ej}
Hepatitis	Manhattan, M_{ej}, M_{coj}, M_{jj}, M_{caj}	M_{ej}, M_{coj}	M_{ej}	Manhattan, M_{ej}	M_{ej}
Treatment	M_{ej}, M_{coj}, M_{jj}	M_{ej}	M_{ej}	M_{coj}	M_{coj}
Labour training evaluation	M_{jj}	M_{coj},M_{jj}	M_{ej},M_{coj}	M_{jj}	M_{jj}
Catsup	Manhattan	M_{jj}	M_{ej}, M_{jj}	M_{coj}	M_{ej}
Azpro	M_{jj}	M_{coj}	M_{coj}	M_{ej}	M_{coj}

Table 11 The results obtained by k-NN with all measures and K = 1, $w_1 = 0.6$ and $w_2 = 0.8$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	100	100	100	100	100	100	100	100	100	100	100	100
Hepatitis	95.56	89.20	100	100	100	100	100	100	100	100	100	100
Treatment	88.78	83.87	98.11	96.97	100	100	100	100	100	100	100	100
Labour training evaluation	88.83	83.98	96.87	87.54	95.40	86.35	95.27	87.31	94.78	85.22	92.17	85.67
Catsup	64.97	55.12	68.34	55.90	69.95	58.81	71.47	59.78	71.44	62.53	69.43	57.26
Azpro	54.24	45.31	56.84	49.03	56.90	48.77	58.21	53.63	58.46	50.07	54.79	48.42

Table 12 The results obtained by k-NN with all measures and K = 3, $w_1 = 0.6$ and $w_2 = 0.8$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	89.32	85.66	93.89	89.38	93.17	90.40	94.69	90.47	95.63	91.53	92.72	88.89
Hepatitis	89.67	85.94	95.74	91.36	95.42	90.85	93.72	89.15	93.12	89.79	91.56	88.90
Treatment	79.69	66.59	83.71	75.65	81.53	74.36	82.48	75.43	84.11	78.82	82.61	72.91
Labour training evaluation	81.45	70.65	83.67	75.32	83.43	76.72	80.83	74.67	80.45	73.80	82.64	73.51
Catsup	68.93	58.23	72.88	65.91	73.33	64.89	76.08	64.57	75.56	66.73	72.74	62.69
Azpro	57.54	51.82	66.23	55.87	62.45	50.62	61.83	52.76	66.37	54.68	62.45	52.92

Table 13 The results obtained by k-NN with all measures and K = 5, $w_1 = 0.6$ and $w_2 = 0.8$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	90.67	84.67	92.45	89.41	94.71	89.66	94.62	90.34	92.55	89.34	89.84	84.77
Hepatitis	89.60	80.29	90.46	82.67	90.73	82.78	92.22	81.83	91.43	82.59	93.52	82.37
Treatment	71.94	68.90	76.85	65.56	75.72	69.44	76.34	67.87	73.83	67.72	75.69	64.58
Labour training evaluation	76.60	68.34	81.54	73.78	82.69	73.01	81.52	72.61	81.89	70.78	81.89	71.30
Catsup	66.34	56.78	69.73	63.73	71.65	65.67	71.56	62.69	70.34	63.14	71.67	60.73
Azpro	57.57	46.72	61.01	45.79	58.83	45.72	59.62	45.69	60.64	46.79	61.44	49.89

Table 16 shows the best measures are used with k-NN for each given k value when $w_1 = 0.6$ and $w_2 = 0.8$.

In the third stage of the experimental, our presumption is that both types of features are important. Therefore, we will assign the same weight value for both of them $w_1 = w_2 = 0.5$.

Tables 17, 18, 19, 20 and 21 show the results obtained by applying k-NN to six heterogeneous data sets with k = 1, 3, 5, 7, and 9 and $w_1 = w_2 = 0.5$.

Again, still k-NN with Manhattan distance outperforms k-NN with Euclidean distance, and the combination of similarity measures perform well with k-NN classifier.

K = 1 is the optimal number for Hypothyroid and Hepatitis, Treatment, and Labour training evaluation data sets. K = 5 is the optimal number for Catsup and Azpro data sets. Table 22 shows the best measures are used with k-NN for each given k value when $w_1 = 0.5$ and $w_2 = 0.5$.

Table 14 The results obtained by k-NN with all measures and $K = 7, w_1 = 0.6$ and $w_2 = 0.8$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	88.17	82.55	89.93	82.33	89.78	83.83	91.35	85.01	91. 63	84.97	89.34	83.63
Hepatitis	85.74	76.83	87.47	81.10	88.82	81.58	86.50	79.89	86.48	81.25	85.23	78.25
Treatment	60.76	53.51	66.49	57.82	67.46	57.43	65.43	52.47	66.76	58.54	62.39	53.72
Labour training evaluation	65.79	49.95	67.89	53.75	71.25	56.70	71.39	61.76	72.60	64.73	69.37	63.10
Catsup	65.67	53.82	65.73	52.27	66. 73	51. 63	67.66	59.23	67.36	56.21	68.34	55.94
Azpro	65.69	44.70	68.78	52.16	69.87	54.57	68.31	50.67	68.11	54.13	64.45	52.23

Table 15 The results obtained by k-NN with all measures and $K = 9, w_1 = 0.6$ and $w_2 = 0.8$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	83.22	70.36	84.51	74.11	87. 22	75.16	86.90	73.24	86.37	72.48	87.29	74.82
Hepatitis	80.90	71.30	84.15	75.52	86.35	75.51	82.45	72.29	84.33	72.83	83.71	74.73
Treatment	73.87	60.96	71.75	63.86	71.87	62.05	74.96	64.13	73.21	64.16	72.66	63.65
Labour training evaluation	64.67	51.94	65.76	53.11	67.78	53.61	67.85	55.10	69.74	55.32	68.20	53.88
Catsup	56.84	46.05	59.81	49.26	61.34	52.26	59.60	46.78	58.50	46.11	58.27	43.34
Azpro	63.77	44.41	66.77	46.88	67.89	48.35	65.65	47.80	66.14	46.51	64.98	45.23

Table 16 The best measures are used with k-NN for each given k value when $w_1 = 0.6$ and $w_2 = 0.8$

Data set	K = 1	K = 3	K = 5	K = 7	K = 9
Hypothyroid	Euclidean, Manhattan, M_{ej}, M_{coj}, M_{jj}, M_{caj}	M_{jj}	M_{ej}, M_{coj}	M_{coj}, M_{jj}	M_{ej}
Hepatitis	Manhattan, M_{ej}, M_{coj}, M_{jj}, M_{caj}	Manhattan, M_{ej}	M_{caj}	M_{ej}	M_{ej}
Treatment	M_{ej}, M_{coj}, M_{jj}, M_{caj}	M_{jj}	Manhattan, M_{coj}	M_{jj}	M_{coj}
Labour training evaluation	Manhattan	M_{ej}	M_{ej}	M_{jj}	M_{jj}
Catsup	M_{jj}	M_{coj},M_{jj}	M_{ej}	M_{coj}, M_{jj}, M_{caj}	M_{ej}
Azpro	M_{coj}	Manhattan	M_{caj}	M_{ej}	M_{ej}

Table 17 The results obtained by k-NN with all measures and $K = 1, w_1 = w_2 = 0.5$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	98.45	96.80	100	100	100	100	100	100	100	100	100	100
Hepatitis	90.24	86.67	100	100	100	100	100	100	100	100	100	100
Treatment	95.45	89.84	98.12	94.02	97.45	92.56	98.78	96.45	99.43	98.36	98.54	97.43
Labour training evaluation	86.24	76.87	92.32	85.04	90.25	83.87	91.21	86.12	90.80	85.18	88.33	81.68
Catsup	65.30	54.68	67.16	57.67	71.34	60.83	72.17	60.69	72.34	59.92	67.11	56.52
Azpro	53.61	42.57	58.33	46.87	58.13	49.33	57.64	45.77	58.11	46.49	57.55	46.12

Table 18 The results obtained by k-NN with all measures and $K = 3, w_1 = w_2 = 0.5$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	92.57	88.65	95.56	89.52	96.67	91.98	95.17	89.88	90.95	90.16	92.64	87.90
Hepatitis	90.97	85.15	94.55	89.28	95.36	91.85	94.88	89.85	95.29	90.79	90.59	87.11
Treatment	85.82	77.71	90.69	88.14	91.87	88.86	90.43	87.22	91.69	89.65	90.70	87.36
Labour training evaluation	79.40	69.38	81.48	77.89	81.73	75.60	84.21	80.16	86.14	80.78	82.47	74.83
Catsup	65.25	58.63	70.45	62.25	68.54	62.62	68.34	60.65	67.70	60.22	70.55	63.59
Azpro	59.70	51.32	63.19	50.36	65.09	53.60	65.33	52.38	63.02	51.59	58.55	48.89

Table 19 The results obtained by k-NN with all measures and $K = 5, w_1 = w_2 = 0.5$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	87.59	78.12	93.79	89.73	89.81	90.13	90.42	88.28	92.89	88.96	91.64	87.15
Hepatitis	84.65	77.54	89.76	82.06	89.75	82.15	88.86	81.37	90.67	82.15	88.68	78.38
Treatment	70.27	65.36	73.68	66.60	75.76	67.62	74.56	65.78	72.22	64.58	71.93	64.97
Labour training evaluation	76.43	65.49	78.15	68.50	79.14	71.82	78.71	72.91	78.47	70.86	78.59	70.64
Catsup	68.45	59.59	72.93	65.32	72.43	64.52	77.56	67.43	74.68	65.78	72.31	64.78
Azpro	53.44	43.62	54.87	46.45	57.76	46.56	56.21	44.98	55.37	42.87	57.76	47.83

Table 20 The results obtained by k-NN with all measures and $K = 7, w_1 = w_2 = 0.5$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	84.46	80.48	88.77	82.48	88.78	82.69	89.60	84.22	87. 34	84.17	87.81	82.72
Hepatitis	83.12	76.23	86.10	78.26	86.98	81.13	85.31	79.15	85.31	79.87	84.80	78.03
Treatment	61.13	51.63	65.93	57.16	64.19	54.83	65.86	54.81	65.54	52.74	60.64	54.62
Labour training evaluation	65.10	54.65	65.54	57.22	68.39	55.18	70.43	59.22	71.48	63.59	67.89	56.10
Catsup	61.55	51.67	65.87	52.76	61. 56	50. 43	65.58	51.82	63.43	51.34	62.87	50.76
Azpro	64.85	44.38	67.26	53.23	66.13	50.64	66.42	51.35	67.64	51.45	64.32	50.22

As it can be seen from the all results obtained by the experiments, there are significant differences between the performance of k-NN with Manhattan distance and k-NN with the Euclidean distance. k-NN with Manhattan distance performs reasonably well over all heterogeneous data sets compared to k-NN with Euclidean distance.

Therefore, the results suggest that k-NN with Euclidean distance is not fit for the purpose to manage naturally heterogeneous data sets. This result supports the obtained results of previous research in [7] that was undertaken for investigating the performance of k-NN with different single measures for classifying heterogeneous data.

5 Conclusions and future work

Since the k-NN classification is based on measuring the distance between the test sample and each of the training samples, the chosen distance function plays a vital role in determining the final classification output. The major objective of this study was to investigate the

Table 21 The results obtained by k-NN with all measures and $K = 9, w_1 = w_2 = 0.5$

Dataset	Traditional k-NN				k-NN with combination similarity measures							
	Euclidean		Manhattan		M_{ej}		M_{coj}		M_{jj}		M_{caj}	
	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)	A (%)	F (%)
Hypothyroid	86.43	75.79	88. 89	77.83	89. 76	80.45	87.65	77.87	86.78	79.54	86.65	78.27
Hepatitis	72.45	64.45	78.43	71.76	76.76	65.76	75.59	67.44	76.33	66.43	74.81	63.72
Treatment	63.54	55.87	63.87	56.34	65.87	55.34	65.41	54.88	64.64	53.86	62.75	52.87
Labour training evaluation	61.60	52.83	68.83	54.40	67.73	54.38	65.12	54.88	63.65	52.52	67.26	51.82
Catsup	56.69	43.73	59.23	47.72	60 .14	50.43	62.34	49.56	59.27	45.64	57.46	41.32
Azpro	60.87	41.46	65.34	42.78	65.45	42.87	64.65	44.76	65.87	45.21	66.67	44.89

Table 22 The best measures are used with k-NN for each given k value when $w_1 = w_2 = 0.5$

Data set	K = 1	K = 3	K = 5	K = 7	K = 9
Hypothyroid	Manhattan, M_{ej}, M_{coj}, M_{jj}, M_{caj}	M_{ej}	Manhattan	M_{coj}	M_{ej}
Hepatitis	Manhattan, M_{ej}, M_{coj}, M_{jj}, M_{caj}	M_{ej}, M_{jj}	M_{jj}	M_{ej}	Manhattan
Treatment	M_{jj}	M_{jj}	M_{ej}	Manhattan	M_{ej}, M_{coj}
Labour training evaluation	Manhattan, M_{coj}	M_{jj}	M_{ej}, M_{coj}	M_{jj}	Manhattan
Catsup	M_{coj}	M_{caj}	M_{coj}	Manhattan	M_{coj}
Azpro	M_{ej}	M_{ej}	M_{caj}	Manhattan	M_{jj}, M_{caj}

performance of k-NN, using several measures includes single measures (Euclidean and Manhattan) and a number of combination of similarity measures, for computing the similarity between data objects described by numerical and binary features. Experimental results were carried out on six heterogeneous data sets from different domains.

The overall results of our experiments showed that Euclidean distance is not an appropriate measure that can be used with k-NN for classifying a heterogeneous data set of numerical and binary features.

Furthermore, our results showed that combining the results of numerical and binary similarity measures is a promising method to get better results than just using one single measure.

Moreover, we have observed that there are no significant differences among the results presented by the three cases of the given weights with k-NN, that may suggest some robustness of the algorithm to the impact of compact heterogeneous features to the classification performance.

Generally, the study has applied in global terms combination of similarity measures with k-NN. This approach does not consider data pre-processing before the analysis.

The study results suggest need for future work: some weights and measures do not necessarily perform well because of the distribution or the quality of the data. Therefore in future work we will address optimisation of the weights selection based on this characteristic of the data representing the ability and quality of the training and testing sets.

Finally, it is important to outline that this work is restricted to limited data types and number of measures, and therefore we aim to investigate the performance and applicability of k-NN for heterogeneous data sets described by more than two types of data, such as numerical, binary, nominal, ordinal, and apply a wider range of measures.

Compliance with ethical standards

Conflict of interest We hereby confirm that this article does not have any conflict of interest or personal relationship with a third party whose interests could be positively or negatively influenced by the article's content.

References

1. Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier, Amsterdam
2. Shavlik JW, Dietterich T, Dietterich TG (1990) Readings in machine learning. Morgan Kaufmann, Los Altos
3. Cover TM, Hart P (1967) Nearest neighbor pattern classification. IEEE Trans Inf Theory 13(1):21–27
4. Tan P-N (2018) Introduction to data mining. Pearson Education, Chennai
5. Wettschereck D (1994) A study of distance-based machine learning algorithms
6. Bramer M (2007) Principles of data mining, vol 180. Springer, Berlin
7. Hu L-Y, Huang M-W, Ke S-W, Tsai C-F (2016) The distance function effect on k-nearest neighbor classification for medical datasets. SpringerPlus 5(1):1304
8. Singh A, Halgamuge MN, Lakshmiganthan R (2017) Impact of different data types on classifier performance of random forest, naive Bayes, and k-nearest neighbors algorithms. Int J Adv Comput Sci Appl 8:1
9. Sentas P, Angelis L (2006) Categorical missing data imputation for software cost estimation by multinomial logistic regression. J Syst Softw 79(3):404–414
10. Todeschini R, Ballabio D, Consonni V, Grisoni F (2016) A new concept of higher-order similarity and the role of distance/similarity measures in local classification methods. Chemom Intell Lab Syst 157:50–57
11. Jiang L, Cai Z, Wang D, Jiang S (2007) Survey of improving k-nearest-neighbor for classification. In: Fourth international conference on fuzzy systems and knowledge discovery (FSKD 2007), vol 1. IEEE, pp 679–683
12. Liu C, Cao L, Philip SY (2014) Coupled fuzzy k-nearest neighbors classification of imbalanced non-IID categorical data. In: 2014 international joint conference on neural networks (IJCNN). IEEE, pp 1122–1129
13. Walters-Williams J, Li Y (2010) Comparative study of distance functions for nearest neighbors. In: Elleithy K (ed) Advanced techniques in computing sciences and software engineering. Springer, Berlin, pp 79–84
14. Deza MM, Deza E (2014) Encyclopedia of distances. Springer, Berlin ISBN 9783662443422
15. Jajuga K, Sokolowski A, Bock H-H (2012) Classification, clustering, and data analysis: recent advances and applications. Springer, Berlin
16. Deza MM, Deza E (2009) Encyclopedia of distances. Springer, Berlin, pp 1–583
17. Evelyn F, Hodges JL Jr (1951) Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, California University, Berkeley
18. Mohammed M, Khan MB, Bashier EBM (2016) Machine learning: algorithms and applications. CRC Press, Boca Raton
19. Larose DT (2015) Data mining and predictive analytics. Wiley, New York
20. Larose DT, Larose CD (2014) Discovering knowledge in data: an introduction to data mining. Wiley, New York
21. Weinshall D, Jacobs DW, Gdalyahu Y (1999) Classification in non-metric spaces. In: Advances in neural information processing systems, pp 838–846
22. Chomboon K, Chujai P, Teerarassamee P, Kerdprasop K, Kerdprasop N (2015) An empirical study of distance metrics for k-nearest neighbor algorithm. In: Proceedings of the 3rd international conference on industrial application engineering, pp 1–6
23. Prasath VB, Alfeilat HAA, Lasassmeh O, Hassanat A, Tarawneh AS (2017) Distance and similarity measures effect on the performance of k-nearest neighbor classifier—a review. arXiv preprint arXiv:1708.04321
24. Cunningham P, Delany SJ (2007) k-nearest neighbour classifiers. Mult Classif Syst 34(8):1–17
25. Todeschini R, Ballabio D, Consonni V (2006) Distances and other dissimilarity measures in chemometrics. In: Meyer RA (ed) Encyclopedia of analytical chemistry: applications, theory and instrumentation. Wiley, New York, pp 1–34
26. Lopes N, Ribeiro B (2016) On the impact of distance metrics in instance-based learning algorithms. In: Iberian conference on pattern recognition and image analysis. Springer, Berlin, pp 48–56
27. Ali N, Rado O, Sani HM, Idris A, Neagu D (2019) Performance analysis of feature selection methods for classification of healthcare datasets. In: Intelligent computing-proceedings of the computing conference. Springer, Berlin, pp 929–938
28. Pereira CL, Cavalcanti GDC, Ren TI (2010) A new heterogeneous dissimilarity measure for data classification. In: 2010 22nd IEEE international conference on tools with artificial intelligence, vol 2. IEEE, pp 373–374
29. Deekshatulu BL, Chandra P (2013) Classification of heart disease using k-nearest neighbor and genetic algorithm. Procedia Technol. 10:85–94
30. Cha S-H (2007) Comprehensive survey on distance/similarity measures between probability density functions. City 1(2):1
31. Liu H, Zhang S (2012) Noisy data elimination using mutual k-nearest neighbor for classification mining. J Syst Softw 85(5):1067–1074
32. Batista G, Silva DF et al (2009) How k-nearest neighbor parameters affect its performance. In: Argentine symposium on artificial intelligence, pp 1–12
33. Peterson MR, Doom TE, Raymer ML (2005) Ga-facilitated KNN classifier optimization with varying similarity measures. In: 2005 IEEE congress on evolutionary computation, vol 3. IEEE, pp 2514–2521
34. Akila A, Chandra E (2013) Slope finder—a distance measure for DTW based isolated word speech recognition. Int J Eng Comput Sci 2(12):3411–3417
35. Yang K, Shahabi C (2004) A PCA-based similarity measure for multivariate time series. In: Proceedings of the 2nd ACM international workshop on multimedia databases. ACM, pp 65–74
36. Cesare S, Xiang Y (2012) Software similarity and classification. Springer, Berlin
37. Silverman D (2006) Interpreting qualitative data: methods for analyzing talk, text and interaction. Sage, Beverly Hills
38. Dillon WR, Goldstein M (1984) Multivariate analysis methods and applications. Number 519.535 D5
39. Finch H (2005) Comparison of distance measures in cluster analysis with dichotomous data. J Data Sci 3(1):85–100
40. Choi S-S, Cha S-H, Tappert CC (2010) A survey of binary similarity and distance measures. J Syst Cybern Inform 8(1):43–48
41. Spencer MS, Prins SCB, Beckom MS et al (2010) Heterogeneous distance measures and nearest-neighbor classification in an ecological setting. Mo J Math Sci 22(2):108–123
42. Salvador-Meneses J, Ruiz-Chavez Z, Garcia-Rodriguez J (2019) Compressed KNN: K-nearest neighbors with data compression. Entropy 21(3):234
43. Sokal R, Sneath PHA (1963) Principles of numerical taxonomy. W.H. Freeman, San Francisco
44. Faith DP, Minchin PR, Belbin L (1987) Compositional dissimilarity as a robust measure of ecological distance. Vegetatio 69(1–3):57–68

45. Ehrig M, Haase P, Hefke M, Stojanovic N (2005) Similarity for ontologies—a comprehensive framework
46. Chen S-M, Lee L-W, Shen VRL (2011) Similarity measures between intervals of linguistic 2-tuples and the intervals of linguistic 2-tuples weighted average operator. In: 2011 international conference on machine learning and cybernetics, vol 4. IEEE, pp 1526–1531
47. Ji Q, Haase P, Qi G (2011) Combination of similarity measures in ontology matching using the OWA operator. In: Yager RR, Kacprzyk J, Beliakov G (eds) Recent developments in the ordered weighted averaging operators: theory and practice. Springer, Berlin, pp 281–295
48. Bashon Y, Neagu D, Ridley MJ (2013) A framework for comparing heterogeneous objects: on the similarity measurements for fuzzy, numerical and categorical attributes. Soft Comput 17(9):1595–1615
49. Chen S-M, Chang C-H (2015) A novel similarity measure between atanassov's intuitionistic fuzzy sets based on transformation techniques with applications to pattern recognition. Inf Sci 291:96–114
50. Chen S-M, Cheng S-H, Lan T-C (2016) A novel similarity measure between intuitionistic fuzzy sets based on the centroid points of transformed fuzzy numbers with applications to pattern recognition. Inf Sci 343:15–40
51. Ho TK (1995) Random decision forests. In: Proceedings of 3rd international conference on document analysis and recognition, vol 1. IEEE, pp 278–282
52. Dua D, Graff C (2017) UCI machine learning repository. http://archive.ics.uci.edu/ml. Accessed 21 Feb 2019
53. Dataset. https://vincentarelbundock.github.io/Rdatasets/datasets.html. Accessed 15 Feb 2019