m_p-dissimilarity: A data dependent dissimilarity measure

Sunil Aryal*, Kai Ming Ting†, Gholamreza Haffari* and Takashi Washio‡
*Clayton School of Information Technology, Monash University, Australia
Email: {sunil.aryal, gholamreza.haffari}@monash.edu
†School of Engineering and Information Technology, Federation University, Australia
Email: kaiming.ting@federation.edu.au
‡The Institute of Scientific and Industrial Research, Osaka University, Japan
Email: washio@ar.sanken.osaka-u.ac.jp

***Abstract*—Nearest neighbour search is a core process in many data mining algorithms. Finding reliable closest matches of a query in a high dimensional space is still a challenging task. This is because the effectiveness of many dissimilarity measures, that are based on a geometric model, such as ℓ_p-norm, decreases as the number of dimensions increases.**

In this paper, we examine how the data distribution can be exploited to measure dissimilarity between two instances and propose a new data dependent dissimilarity measure called 'm_p-dissimilarity'. Rather than relying on geometric distance, it measures the dissimilarity between two instances in each dimension as a probability mass in a region that encloses the two instances. It deems two instances in a sparse region to be more similar than two instances in a dense region, though these two pairs of instances may have the same geometric distance.

Our empirical results show that the proposed dissimilarity measure indeed provides a reliable nearest neighbour search in high dimensional spaces, particularly in sparse data. m_p-dissimilarity produced better task specific performance than ℓ_p-norm and cosine distance in classification and information retrieval tasks.

***Index Terms*—distance measure, ℓ_p-norm, m_p-dissimilarity**

I. Introduction

In order to make a prediction for a given query, many data mining algorithms search for the k closest matches or nearest neighbours (NNs) of the query in a database, and make a prediction based on those kNNs. They use a similarity or dissimilarity measure to find kNNs. Minkwoski distance (aka ℓ_p-norm) [1] is a widely used dissimilarity measure. Even though it performs well in many applications, its effectiveness degrades as the number of dimensions increases. In high dimensional space, data distribution becomes sparse which makes the concept of distance meaningless - "*curse of dimensionality*". All pairs of points are almost equidistant for a wide range of data distributions and distance measures [2, 3] resulting in unreliable closest match that leads to erroneous predictions.

The performance of distance measure depends on the data distribution and task at hand. A distance measure that performs well in one distribution or task may perform poorly in others. A huge variation in performance can be observed when a distance measure is used in different data distributions and tasks. We hypothesize that this variation is because the distance measure computes the dissimilarity between two instances solely based on the geometric positions. The data distribution (i.e., the relative position of the two instances with respect to the rest of the data) is not taken into consideration.

Many psychologists have expressed their concerns on the geometric model of dissimilarity measure [4, 5]. They have argued that the judged dissimilarity between two instances is influenced by the context of measurements and other instances in proximity. Krumhansl [5] has suggested a distance-density model of dissimilarity measure, arguing that two instances in a relatively dense region would be less similar than two instances of equal distance but located in a less dense region. For example, two white persons will be judged as more similar when compared in Africa (where there are less white and more black people) than in America (where there are many white people.)

In this paper, we propose a new dissimilarity measure called 'm_p-dissimilarity' in which data distribution is the primary factor in measuring dissimilarity between instances. Rather than using a spatial distance in each dimension, m_p-dissimilarity evaluates the dissimilarity between two instances in terms of probability mass in a region covering the two instances in each dimension. The final dissimilarity between the two instances is estimated as a power mean of dissimilarities in each dimension as in ℓ_p-norm. The intuition behind the proposed dissimilarity measure is that two instances are likely to be more dissimilar if there are more instances in between and around them in many dimensions. Under the proposed data dependent dissimilarity measure, two instances in a dense region of the distribution are more dissimilar than two instances having the same geometric distance in a sparse region, as prescribed by psychologists.

This paper makes the following contributions:

1) Propose a new data dependent dissimilarity measure called m_p-dissimilarity.
2) Provide its theoretical basis and interpretation.
3) Compare the performance of m_p-dissimilarity against ℓ_p-norm and cosine distance in moderate to high dimensional data sets from text and music domains in classification and information retrieval tasks.

The rest of the paper is organised as follows. Two widely used geometric distance measures ℓ_p-norm and cosine distance are discussed in Section II. The proposed data dependent dissimilarity measure, m_p-dissimilarity, is discussed in Section III. Empirical results are provided in Section IV followed by conclusions and future work in the last section. From now on we refer m_p-dissimilarity and ℓ_p-norm by m_p and ℓ_p, respectively.

II. Measures based on geometric models

A wide range of geometric (proximity based) dissimilarity measures are used in the literature which are discussed in [1] . In this section, we discuss the two most widely used measures: ℓ_p-norm and cosine distance.

A. ℓ_p-norm distance

The distance between two d-dimensional vectors $\mathbf{x}$ and $\mathbf{y}$ based on ℓ_p-norm is defined as follows [1]:

$$\ell_p(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_p = \left(\sum_{i=1}^{d} abs(x_i - y_i)^p\right)^{\frac{1}{p}} \quad (1)$$

where $p > 0$, $\|\cdot\|_p$ is the p order norm of a vector, a_i is the i^{th} component of a vector $\mathbf{a}$ and $abs(\cdot)$ is the absolute value. The limit condition is defined as follows:

$$\ell_\infty(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_\infty = \max_i \ abs(x_i - y_i) \quad (2)$$

Manhattan distance (ℓ_1), Euclidean distance (ℓ_2) and Chebysev distance (ℓ_∞) are widely used ℓ_p-norm based distance functions. Euclidean distance is a popular choice of distance function as it intuitively corresponds to the distance defined in the real three-dimensional world.

B. Cosine distance

In many high dimensional problems, data have the same value (0 or any other constant) in many dimensions, creating 'sparseness'. For example, only a few terms in a dictionary appear in each document in a corpus. Many entries of a vector representing a document are zero. ℓ_p-norm is not a good choice of distance measure in such problems. The direction of vectors is more important than their lengths. The angular distance measure (aka cosine distance) [1] is more sensible choice to measure dissimilarity between two documents.

The cosine distance between two vectors $\mathbf{x}$ and $\mathbf{y}$ is defined as follows [1]:

$$\begin{aligned} d_{cos}(\mathbf{x},\mathbf{y}) &= 1 - \frac{\mathbf{x}\cdot\mathbf{y}}{\|\mathbf{x}\|_2 \times \|\mathbf{y}\|_2} \\ &= 1 - \frac{\sum_{i=1}^{d} x_i \times y_i}{\sqrt{\sum_{i=1}^{d} x_i^2} \times \sqrt{\sum_{i=1}^{d} y_i^2}} \end{aligned} \quad (3)$$

Cosine distance has been shown to perform well in many text mining problems such as text categorization, text clustering and text retrieval tasks.

III. Data dependent measure

In order to measure dissimilarity between $\mathbf{x}$ and $\mathbf{y}$, instead of using $(x_i - y_i)$ in Eqn. 1, we propose to consider the relative positions of $\mathbf{x}$ and $\mathbf{y}$ with respect to the rest of the data distribution in each dimension. The dissimilarity between $\mathbf{x}$ and $\mathbf{y}$ in dimension i can be estimated as the probability data mass in a region $R_i(\mathbf{x},\mathbf{y})$ that encloses $\mathbf{x}$ and $\mathbf{y}$. If there are many instances in $R_i(\mathbf{x},\mathbf{y})$, $\mathbf{x}$ and $\mathbf{y}$ are likely to be more dissimilar in dimension i. Using the same power mean formulation as in ℓ_p-norm, the data dependent dissimilarity measure based on probability mass can be defined as:

$$m_p(\mathbf{x},\mathbf{y}) = \left(\sum_{i=1}^{d} \left(\frac{|R_i(\mathbf{x},\mathbf{y})|}{n}\right)^p\right)^{\frac{1}{p}} \quad (4)$$

where $|R_i(\mathbf{x},\mathbf{y})|$ is the data mass in region $R_i(\mathbf{x},\mathbf{y})$, $R_i(\mathbf{x},\mathbf{y}) = [\min(x_i,y_i) - \delta, \max(x_i,y_i) + \delta]$, $\delta \geq 0$ and n is the number of data instances.

An example of $R_i(\mathbf{x},\mathbf{y})$ is shown in Figure 1. We use $\delta = \frac{\sigma_i}{2}$ (σ_i is the standard deviation of data in dimension i) in this paper.

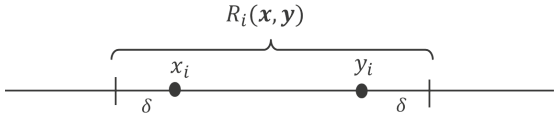


Fig. 1: $R_i(\mathbf{x},\mathbf{y})$

We call the proposed dissimilarity measure $m_p(\mathbf{x},\mathbf{y})$ 'm_p-dissimilarity'. This measure captures the essence of the distance-density model proposed by Krumhausl [5] which prescribes that two instances in a sparse region are more similar than two instances in a dense region. Although m_p employs the same power mean formulation as ℓ_p, the core calculation is based on mass rather than distance. It signifies the degree of dissimilarity: the higher the measure, the more dissimilar the two instances are; just like ℓ_p.

The formulation of $m_p(\mathbf{x},\mathbf{y})$ (Eqn. 4) has a probabilistic interpretation. The simplest form of data dependent dissimilarity measure is to define a region $R(\mathbf{x},\mathbf{y}) \in \mathcal{R}^d$ that encloses $\mathbf{x}$ and $\mathbf{y}$ and estimate the probability of a randomly selected point $\mathbf{z}$ from the distribution of data, $\phi(\mathbf{x})$, falling in $R(\mathbf{x},\mathbf{y})$. Let $R(\mathbf{x},\mathbf{y})$ be centered at $\mathbf{h} = \langle h_1, h_2, \cdots, h_d\rangle$, $h_i = \frac{x_i+y_i}{2}$ and has length of $R_i(\mathbf{x},\mathbf{y})$ on dimension i. We use the shorthand R and R_i to represent $R(\mathbf{x},\mathbf{y})$ and $R_i(\mathbf{x},\mathbf{y})$, respectively. Assuming that the dimensions are independent, it can be approximated as:

$$P(\mathbf{z} \in R|\phi(\mathbf{x})) \approx \prod_{i=1}^{d} P_i(\mathbf{z} \in R|\phi(\mathbf{x})) \quad (5)$$

where $P_i(\mathbf{z} \in R|\phi(\mathbf{x}))$ is the probability of $\mathbf{z}$ falling in R in dimension i.

The approximation using Eqn. 5 is sensitive to outliers. $P(\mathbf{z} \in R|\phi(\mathbf{x}))$ becomes small (or zero) even if only one $P_i(\mathbf{z} \in R|\phi(\mathbf{x}))$ is small (or zero). An approximation which

is tolerant to outliers can be estimated by replacing the product with a summation [6].

Lemma 1. *(Minka [6]) In an outlier model having data distribution $\phi(\mathbf{x})$,*

$$\prod_{i=1}^{d} P_i\left(\mathbf{x}|\phi(\mathbf{x})\right) \propto \sum_{i=1}^{d} P_i\left(\mathbf{x}|\phi(\mathbf{x})\right)$$

Proof. Let us consider a data generation process in which in order to sample $\mathbf{x}$, a coin with probability of turning head $(1-\epsilon)$ is flipped. If the coin turns head, $\mathbf{x}$ is drawn from $\phi(\mathbf{x})$ where the probability of sampling $\mathbf{x}$ is $P\left(\mathbf{x}|\phi(\mathbf{x})\right)$ else it is drawn from a uniform distribution $1/A$ (A is the area under the domain of $\mathbf{x}$). This model considers outliers as:

$$P'_i\left(\mathbf{x}|\phi(\mathbf{x})\right) = \epsilon/A + (1-\epsilon)P_i\left(\mathbf{x}|\phi(\mathbf{x})\right) \tag{6}$$

Using Eqn. 5,

$$\begin{aligned} P'\left(\mathbf{x}|\phi(\mathbf{x})\right) &\approx \prod_{i=1}^{d} P'_i\left(\mathbf{x}|\phi(\mathbf{x})\right) \\ &\approx \prod_{i=1}^{d} \left(\epsilon/A + (1-\epsilon)P_i\left(\mathbf{x}|\phi(\mathbf{x})\right)\right) \end{aligned} \tag{7}$$

A Taylor series expansion in $(1-\epsilon)$ leads to:

$$(\epsilon/A)^d + (\epsilon/A)^{d-1}(1-\epsilon)\sum_{i=1}^{d} P_i\left(\mathbf{x}|\phi(\mathbf{x})\right) + O\left((1-\epsilon)^2\right)$$

In the extreme case where there are many outliers, i.e. ϵ is close to 1, only the first two terms matter. The first term is a constant and hence, Lemma 1 follows.

□

In addition to the above approximation given by Minka [6], we propose that the chance of a point being drawn from the outlier model can be further reduced by sampling from $\phi(\mathbf{x})^p$, yielding the probability of sampling $\mathbf{x}$ as $P(\mathbf{x}|\phi(\mathbf{x}))^p$, where $P(\cdot)^p$ is the probability of a random event occurring in p successive trials.

Lemma 2. *In the outlier model of $\phi(\mathbf{x})$, a more generalised outlier tolerant approximation can be achieved as:*

$$\prod_{i=1}^{d} P_i\left(\mathbf{x}|\phi(\mathbf{x})\right) \propto \sum_{i=1}^{d} P_i\left(\mathbf{x}|\phi(\mathbf{x})\right)^p$$

Proof. This follows from the proof of Lemma 1 by simply drawing $\mathbf{x}$ from $\phi(\mathbf{x})^p$ when head turns up in the coin toss. □

Using Lemma 2, Eqn. 5 can be expressed as follows:

$$P(\mathbf{z} \in R|\phi(\mathbf{x})) \propto \sum_{i=1}^{d} P_i(\mathbf{z} \in R|\phi(\mathbf{x}))^p \tag{8}$$

As a result of Eqn. 8 and ignoring the constant which is just a scaling factor of the dissimilarity, $m_p(\mathbf{x}, \mathbf{y})$ can be estimated as follows:

$$m_p(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{d} P_i\left(\mathbf{z} \in R|\phi(\mathbf{x})\right)^p\right)^{\frac{1}{p}} \tag{9}$$

where the outer power of $\frac{1}{p}$ is just a rescaling of $P(\mathbf{z} \in R|\phi(\mathbf{x}))$.

It is important to note the two assumptions made in the above derivation of m_p, i.e., dimension independence and outlier model. The assumption of dimension independence has been applied in data mining, e.g., Naive Bayes classifier. It has been shown that this assumption does not affect the classification accuracy in many scenarios even if the assumption is violated.

With the assumption of the outlier model, m_p produces many small $P_i\left(\mathbf{z} \in R(\mathbf{x}, \mathbf{y})|\phi(\mathbf{x})\right)$ if $\mathbf{x}$ and $\mathbf{y}$ are similar. In other words, instances which are similar are assumed to have small $|R_i|$ in many dimensions. It is not an unrealistic assumption in high dimensional problems.

In practice, $P_i\left(\mathbf{z} \in R|\phi(\mathbf{x})\right)$ can be estimated as:

$$P_i\left(\mathbf{z} \in R|\phi(\mathbf{x})\right) = \frac{|R_i|}{n} \tag{10}$$

Hence, Eqn. 9 and Eqn. 10 lead to m_p-dissimilarity defined in Eqn. 4. The role of parameter p is similar to that in ℓ_p, i.e., p controls the influence of a dimension by scaling up the degree of dissimilarity.

Figure 2 shows the contours of dissimilarity measured from an instance at (0.5,0.5) based on m_2 (m_p with $p = 2$) in three different data distributions (uniform, normal and bimodal). In contrast, ℓ_p and cosine distance would produce the same contour in all three distributions. Under uniform distribution and infinite samples, m_p will yield the same result as ℓ_p because the data mass in R_i will be proportional to $x_i - y_i$. This is depicted in the first contour plot in Figure 2 where it approaches the contour plot of ℓ_2.

Complexity:

Computationally, m_p is more expensive than ℓ_p as it requires a range search in each dimension. One dimensional range search can be done in $O(\log n)$ using binary search trees. Hence, the dissimilarity of a pair of instances can be computed in $O(d \log n)$ against $O(d)$ of ℓ_p. In sparse data, the unique values in each dimension will be a lot less than n. Hence, the average case run time will be a lot less than $O(d \log n)$. Also, it requires $O(dn)$ time and $O(d \log n)$ space to build and store d binary search trees, respectively.

IV. Empirical evaluations

This section presents the results of the experiments conducted to evaluate the performance of m_p against ℓ_p and cosine distance in kNN classification and information retrieval.

Eleven data sets from different domains with different sizes ($1000 \leq n \leq 9100$), number of dimensions ($188 \leq d \leq 10000$) and number of classes ($2 \leq c \leq 52$) were used.

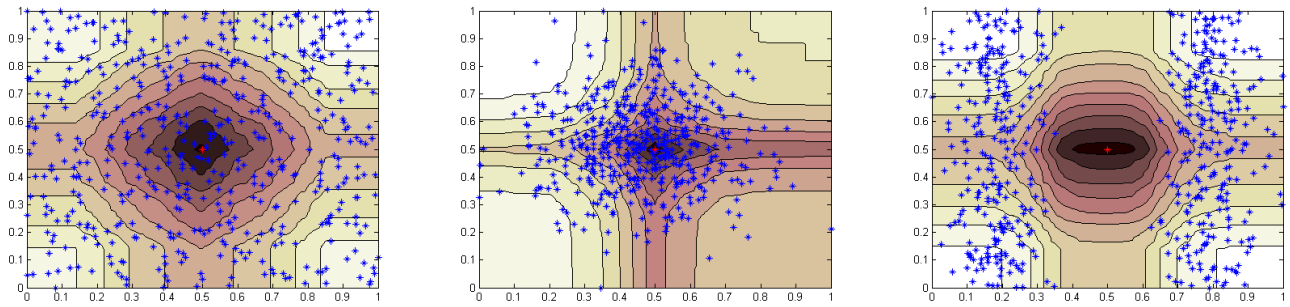


Fig. 2: Contour plots of dissimilarity based on m_2-dissimilarity to the instance at (0.5,0.5) in three different data distributions: uniform, normal and bimodal.

All the attributes in the data sets are numeric. Out of 11 data sets used, six are from text mining domain, two from music classification and retrieval domain, 2 from character recognition and the last one is a synthetic data set from UCI machine learning repository [7]. Text data were represented by TFIDF [8] weighted 'bag of words' vectors. Other data sets (non-text) were normalised to the range of [0,1]. The properties and references of the data sets are provided in Table I.

We discuss the experimental set-ups and results in classification and information retrieval tasks in the following two subsections.

TABLE I: Characteristics of data sets

Name [Ref]	n	d	c	Domain
Amazon [7]	1500	10000	50	text
CNAE [7]	1080	856	9	text
Reuter [7]	5000	9288	50	text
R8 [9]	7674	3497	8	text
R52 [9]	9100	7369	52	text
Webkb [9]	4199	1818	4	text
HBA [10]	1500	188	15	music
GTZAN [11]	1000	230	10	music
Gisette [7]	7000	5000	2	digit recognition
Mfeat [7]	2000	649	10	digit recognition
Madelon [7]	2600	500	2	artificial data

A. *kNN classification*

In the kNN classification context, in order to predict a class label for a test instance $\mathbf{x}$, its k nearest neighbours were searched in the training set based on a dissimilarity measure and the most frequent label of the kNN instances was predicted.

All classification experiments were conducted using a 10-fold cross validation. We used four settings of p $(2.0, 1.0, 0.5, 0.1)$ in ℓ_p and m_p and two settings of k ($k = 1$ and $k = 10$) for all classifiers. The average accuracy (%) over a 10-fold cross validation is reported. The accuracies of two algorithms are considered to be significantly different if their confidence intervals (based on $\pm$ one standard error) do not overlap.

The best average classification accuracy over a 10-fold cross validation achieved by m_p, ℓ_p and cosine distance in all 11 data sets is presented in Figure 3. A red dot on the top of the bar indicates that the best performer had significantly better classification accuracy than the other two contenders.

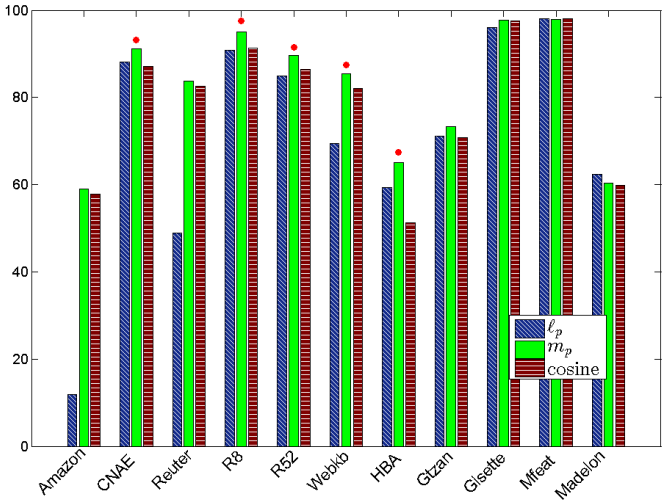
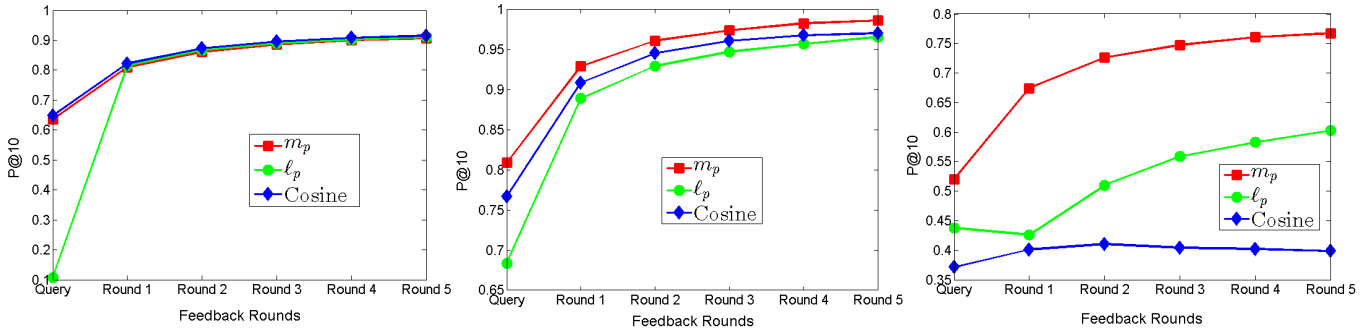


Fig. 3: The best classification accuracies of ℓ_p, m_p and cosine distance in kNN classifier. A red dot on the top signifies that the best performer had significantly better classification accuracy than the other two contenders.

As shown in Figure 3, m_p produced better classification accuracies than ℓ_p and cosine distance in eight data sets and similar results in the other three data sets. The result is statistically significant in five data sets (CNAE, R8, R52, Webkb and HBA) and not significantly worst in any data set.

It is interesting to note that m_p produced significantly better classification accuracy than ℓ_p in all six text (sparse) data sets; and better than cosine distance in four out of six. This is because m_p assigns (i) the maximum dissimilarity (of a dimension) if the majority of instances have the same value which is often the case in sparse text data where term frequencies are zeros in many dimensions; and (ii) the minimum dissimilarity if the value has the least number of training instances in the local neighbourhood.

In terms of p, m_p produced better results with $p = 2$ in



(a) Reuter (b) CNAE (c) HBA

(d) Amazon (e) R8 (f) Gtzan

Fig. 4: Precision at top 10 retrieved results (P@10).

eight out of 11 data sets used with the exceptions of Amazon ($p = 0.5$), CNAE ($p = 0.1$) and Madelon ($p = 0.1$). The result with ℓ_p, was mixed: $p = 0.1$ produced better classification result in four data sets, $p = 2$ was better in four, $p = 1$ was better in two and 0.5 was better in one data set.

Generally, we observed that $p = 2$ is a reasonable setting in m_p, but we can not say anything about setting p in ℓ_p as the accuracy varies significantly with p.

B. Information retrieval

In information retrieval, given a query $\mathbf{q}$, the relevance of a database instance $\mathbf{x}$, $Rel(\mathbf{x}|\mathbf{q})$, was measured using dissimilarity measure f as:

$$Rel(\mathbf{x}|\mathbf{q}) = -f(\mathbf{x}, \mathbf{q}) \tag{11}$$

In a relevance feedback process [12], a user examines the current retrieval result and provides some 'relevant' and 'irrelevant' examples to the retrieval system. Let $\mathcal{Q} = \mathcal{P} \cup \mathcal{N}$ be a set of feedback instances to the query $\mathbf{q}$ where $\mathcal{P}$ and $\mathcal{N}$ are the sets of positive and negative feedback, respectively. Note that $\mathcal{P}$ includes $\mathbf{q}$. In a relevance feedback round, the relevance score was estimated as follows:

$$Rel(\mathbf{x}|\mathcal{Q}) = \frac{1}{|\mathcal{P}|} \sum_{\mathbf{y}^+ \in \mathcal{P}} Rel(\mathbf{x}|\mathbf{y}^+) - \gamma \frac{1}{|\mathcal{N}|} \sum_{\mathbf{y}^- \in \mathcal{N}} Rel(\mathbf{x}|\mathbf{y}^-) \tag{12}$$

where $0 \leq \gamma \leq 1$ is a trade-off parameter for the relative contribution of positive and negative feedback.

We used text and music information retrieval data sets (Reuter, CNAE, HBA, Amazon, R8 and Gtzan) with more than five classes in information retrieval. R52 was not used in information retrieval as the class distribution is heavily skewed and many classes have a few instances, which are not enough for query and feedback.

Initially five queries were chosen randomly from each class. For each query, instances from the same class were regarded as relevant and those from the other classes were irrelevant. At each round of feedback, two relevant (instances from the same class) and two irrelevant (instances from the other classes) instances were provided. Five rounds of feedback were conducted for each query. An instance was not used in ranking if it was used as a feedback instance in current or previous feedback rounds. The feedback process was repeated five times with different relevant and irrelevant feedback. This process was repeated 10 times with different queries from each class. The average precision at top 10 (P@10) returned results was reported.

We used the same four settings of p (2.0, 1.0, 0.5, 0.1) and two settings of γ (0,1). Note that when $\gamma = 0$, no negative feedback was needed. The best result achieved at the end of the fifth round of feedback is shown in Figure 4. m_p had produced either better than or similar results to ℓ_p and cosine distance in five data sets. The only exception is in R8 where cosine distance was better than m_p.

It is interesting to note that m_p produced better results with $\gamma = 0$. Its performance degraded in all cases when negative

feedback were given. This is because m_p considers the probability of two instances being different and assigns dissimilarity score according to the distribution of other instances already. Hence, deducting the average relevance score w.r.t irrelevant feedback affects the relevance score of an instance w.r.t $\mathbf{q}$. An instance relevant to a negative feedback may not be equally irrelevant to the query.

On the other hand, ℓ_p-norm significantly improved its performance when negative feedback were given. The performance was improved drastically even in the first round of feedback in the sparse text data sets (Reuter, CNAE, Amazon and R8) whereas this was not the case in the non-sparse music data sets (HBA and Gtzan). In text data, instances are similar in many dimensions with zero values. Initially, in the query round, many irrelevant instances get a high relevance score as ℓ_p assigns zero distance in many dimensions because of zero frequency. They also have high similarity with negative feedback. Hence, deducting the average relevance w.r.t negative feedback compensates well for the high relevance score given in the first place to irrelevant instances. With negative feedback, ℓ_p produced competitive retrieval results with m_p and cosine distance in the Amazon and Reuter data sets. In the other four data sets, ℓ_p was significantly worse than m_p.

Cosine distance produced significantly worst results in the music data sets. In text retrieval, it produced better result than m_p in subsequent feedback rounds in R8 but was worse than m_p in CNAE. In Amazon and Reuter, they produced similar retrieval results. Note that, cosine distance also produced better results with $\gamma = 1$, i.e., with negative feedback. m_p produced significantly better retrieval performance than ℓ_p and cosine distance in the music (non-sparse) data sets (HBA and Gtzan).

Again, $p = 2$ was better in all six data sets for m_p in information retrieval. For ℓ_p-norm, $p = 1$ or 2 achieved the best retrieval results.

V. Conclusions and future work

In this paper, we proposed a new dissimilarity measure called 'm_p-dissimilarity' that mainly utilises data distribution in its dissimilarity calculations. It estimates the dissimilarity between two instances in each dimension as a probability of data mass that falls in a region enclosing the two instances. The final dissimilarity between the two instances is estimated as the power mean of all single dimensional dissimilarities as in the case of ℓ_p. The fundamental difference between the formulations of m_p and ℓ_p is the replacement of the geometric difference with the probability mass.

Our empirical evaluations in classification and information retrieval suggest that m_p provides more meaningful closest neighbours than those provided by ℓ_p and cosine distance in high dimensional space, especially in text data sets where sparsity is a dominant data characteristic.

The potential avenue for future work includes investigating an efficient implementation of m_p-dissimilarity, its strengths and limitations along with theoretical analysis and applying m_p to tasks such as clustering, anomaly detection and kernel learning.

Acknowledgement

Sunil Aryal is supported by Australian Postgraduate Award (APA), Monash University. This project is partially supported by a grant from the U.S. Air Force Research Laboratory, under agreement# FA2386-13-1-4043, awarded to Kai Ming Ting.

References

[1] M. M. Deza and E. Deza, *Encyclopedia of Distances*. Springer Berlin Heidelberg, 2009.

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

[3] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the Surprising Behavior of Distance Metrics in High Dimensional Space," in *Proceedings of the 8th International conference on Database Theory*. Springer Berlin Heidelberg, 2001, pp. 420–434.

[4] A. Tversky, "Features of similarity," *Psychological Review*, vol. 84, no. 4, pp. 327–352, 1977.

[5] C. L. Krumhansl, "Concerning the applicability of geometric models to similarity data: The interrelationship between similarity and spatial density," *Psychological Review*, vol. 85, no. 5, pp. 445–463, 1978.

[6] T. P. Minka, "The 'summation hack' as an outlier model," http://research.microsoft.com/en-us/um/people/minka/papers/minka-summation.pdf, 2003, Microsoft Research.

[7] K. Bache and M. Lichman, "UCI machine learning repository," http://archive.ics.uci.edu/ml, 2013, University of California, Irvine, School of Information and Computer Sciences.

[8] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[9] A. Cardoso-Cachopo, "Improving Methods for Single-label Text Categorization," Ph.D. dissertation, Instituto Superior Tecnico, Technical University of Lisbon, Lisbon, Portugal, 2007.

[10] H. B. Ariyaratne and D. Zhang, "A novel automatic hierachical approach to music genre classification," in *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops*. IEEE Computer Society, Washington DC, USA, 2012, pp. 564–569.

[11] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[12] Y. Rui, T. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: a power tool for interactive content-based image retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 644–655, 1998.