2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)
July 6-11, 2014, Beijing, China

Investigating Distance Metric Learning in Semi-supervised Fuzzy c-means Clustering

Daphne Teck Ching Lai, Jonathan M. Garibaldi and Jenna Reps

***Abstract*— The idea behind distance metric learning (DML) is to accentuate the distance relations found in the training data, maintaining whether the data patterns are similar or dissimilar. In this paper, we investigate in using DML (GDML, LMNN, MCML and NCA) in semi-supervised Fuzzy c-means clustering and apply them on a real, biomedical dataset and on UCI datasets. We used a cross validation setting with varying amount of labelled data to test our methodology. Out of eight datasets, statistical significant improvement was found on five datasets using ssFCM with DML. This shows that DML can improve ssFCM clustering for some datasets. Further analysis using 2D PCA projection and sum of squared distances before and after DML transformation of the original data are carried out. Interestingly, DML was found to worsen ssFCM clustering in the NTBC dataset with hierarchical clusters.**

I. Introduction

Distance metrics are critical in data mining algorithms because they reflect the important structure within the data. Traditionally, users manually adjust the metric or experiment with several different metrics until adequately good clusters are found. This has motivated researchers to learn a distance metric using labelled data as examples. Distance Metric Learning (DML) techniques allow us to learn a distance metric of the data such that the distance relation among the training data is preserved by using labelled data to indicate whether they are similar and dissimilar. Many studies have shown that the use of a learned metric can improve the performance of classification and clustering tasks. A review on distance metric learning is found in [1].

The idea behind DML is such that if the algorithm is informed which data patterns are similar and which are dissimilar, the algorithm will learn a distance metric that maintains this relationship, where smaller distances are assigned to similar distances and bigger distances are assigned to dissimilar distances. This suggests that the separability between clusters can be increased using DML, particularly for datasets where clusters overlap. This in turn can improve clustering for algorithms such as semi-supervised Fuzzy c-means (ssFCM) where the similarity of data patterns are defined by distance metrics. In [2], Reps *et al.* employed DML into a semi-supervised clustering framework with integration of existing techniques and information from the web for the identification of rare adverse side effects of drugs. Similarly, we wish to employ DML to ssFCM to improve clustering results.

The popularity of distance metric learning (DML) is rapidly increasing. While DML has been largely applied to K-means and constrained K-means clustering [3], [2], a few studies have been done on application of DML techniques to semi-supservised Fuzzy c-means (ssFCM) algorithms [4], [5], [6]. Ceccarelli and Maratea [4] applied GDML [3] to ssFCM. In this paper, we aim to expand this work by exploring other DML algorithms such as NCA, MCML and LMNN applied to ssFCM with investigation in distance metric, which, to the best of our knowledge, have not been done.

Semi-supervised fuzzy c-means (ssFCM) is an extended form of fuzzy c-means [7], which uses available prior knowledge in the form of labelled data to guide the clustering of unlabelled data. This is beneficial as labelled data are scarce and expensive to collect. Our study is focused on ssFCM as it can represent data in more than one clusters using membership values and can learn from labelled data patterns. Furthermore, ssFCM has been demonstrated to perform well in many clustering problems, such as traffic classification in [8] and identifying biological clusters in [9], to name a few.

To fulfill our overarching research objective [10] of exploring different techniques to incorporate into an ssFCM framework for improving its clustering, in this paper, we investigate the application of DML techniques Global Distance Metric Learning (GDML) [3], Neighbourhood Components Analysis (NCA) [11], Maximally Collapsing Metric Learning (MCML) [12] and Large Margin Nearest Neighbour (LMNN) [13] in ssFCM. Although NCA, MCML and LMNN are designed for k-nearest neighbour (KNN), where data patterns located close together are assigned the same labels and those far away with different labels, clustering algorithms are recognised to work on similar concepts of similarity using distance metric. Thus, this paper focuses on investigating the performance of ssFCM with the application of DML on the NTBC and UCI datasets.

The paper is organised as follows: We review ssFCM and DML in Section II and III respectively. The experimental methods are found in Section IV. This is followed by results and discussion in Section VI and VII before we reach conclusions in Section VIII.

II. Algorithms

In this section, we briefly describe the four DML algorithms and ssFCM algorithm selected for our investigation.

A. Distance Metric Learning

1) Global Distance Metric Learning: The Global Distance Metric Learning (GDML) technique [3] learns a global distance metric that that minimises the distance between the data patterns that are similar.

Suppose there are some points $\{\mathbf{x}_i\}_{i=1}^{m} \subseteq \mathbb{R}^n$, and those that are similar are expressed as:

$$S : (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{S} \qquad \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are similar.} \tag{1}$$

Daphne Teck Ching Lai, Jonathan M. Garibaldi and Jenna Reps are with the School of Computer Science, University of Nottingham, United Kingdom (email: {dtl, jmg, jzr}@cs.nott.ac.uk).

978-1-4799-2072-3/14/$31.00 ©2014 IEEE

To calculate similarity, the distance metric of the following form is used:

$$d(\mathbf{x},\mathbf{y}) = d_{\mathbf{A}}(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||_{\mathbf{A}} = \sqrt{(\mathbf{x}-\mathbf{y})^T\mathbf{A}(\mathbf{x}-\mathbf{y})}. \quad (2)$$

For a similarity measure to be a metric, it needs to be non-negative and conform to the triangle inequality. Thus, $\mathbf{A}$ has to be positive semi-definite, $\mathbf{A} \succeq 0$ in constraint (5). The constraint in (4) ensures that $\mathbf{A}$ is not zero since this will cause the dataset to collapse to a single point, which is not useful. The objective function is expressed as an optimisation problem as follows:

$$\max_{\mathbf{A}} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{S}} ||\mathbf{x}_i-\mathbf{x}_j||^2_{\mathbf{A}} \quad (3)$$

$$\text{s.t.} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{D}} ||\mathbf{x}_i-\mathbf{x}_j||_{\mathbf{A}} \geq 1, \quad (4)$$

$$\mathbf{A} \succeq 0. \quad (5)$$

To learn the diagonal $\mathbf{A}$, the Newton-Raphson method is used. This would rescale the data and replace each point $\mathbf{x}$ with $\mathbf{A}^{1/2}\mathbf{x}$. To learn a full matrix $\mathbf{A}$, the constraint $\mathbf{A} \succeq 0$ becomes a hard problem. Gradient descent and iterative projection are used to solve the objective function for the full matrix $\mathbf{A}$.

2) Neighbourhood Components Analysis: The Neighbourhood Components Analysis (NCA) [11] learns a Mahalanobis distance metric specifically for the KNN classifier by optimising the expected leave-one-out (LOO) performance on the training data using stochastic neighbour selection rule. Let a labelled data set be $\mathbf{x}_1,...,\mathbf{x}_n$ in $\mathcal{R}^{\mathcal{D}}$ with corresponding class labels $c_1,...,c_n$. To ensure the metric to be learned symmetric positive semi-definite, it is in the form $\mathbf{Q} = \mathbf{A}^T\mathbf{A}$ such that $d(x,y) = (x-y)^T\mathbf{Q}(x-y) = (\mathbf{A}x-\mathbf{A}y)^T(\mathbf{A}x-\mathbf{A}y)$.

In leave-one-out cross validation setting, the authors consider the entire transformed data set as stochastic nearest neighbours. Given a point $\mathbf{x}_i$, a stochastic ("soft") neighbour of $\mathbf{x}_i$ is defined by p_{ij}, which is the probability of $\mathbf{x}_i$ selecting $\mathbf{x}_j$ as its neighbour and sharing the same class label. The probability p_{ij} is defined as:

$$p_{ij} = \frac{\exp(-||\mathbf{A}\mathbf{x}_i-\mathbf{A}\mathbf{x}_j||^2)}{\sum_{k=i}\exp(-||\mathbf{A}\mathbf{x}_i-\mathbf{A}\mathbf{x}_k||^2)}. \quad (6)$$

We denote the set of points in the same class as i by $C_i = \{j|c_i = c_j\}$. Thus, the probability p_i that $\mathbf{x}_i$ will be correctly classified is $p_i = \sum_{j\in C_i} p_{ij}$ and the expected number of points correctly classified is $f(A) = \sum_{i=1}^{n} p_i$. To maximise the objective $f(A)$, the first-order derivative of $f(A)$ is taken with respect to A.

3) Maximally Collapsing Metric Learning: The Maximally Collapsing Metric Learning (MCML) [12] proposed a convex optimisation problem which learns the Mahalanobis distance metric that collapses data from the same class to a point and push data in other classes far apart.

Given a data pattern x_i, a conditional distribution over points $i \neq j$ is defined:

$$p^A(j|i) = \frac{1}{Z_i}e^{-d^A_{ij}} = \frac{e^{-d^A_{ij}}}{\sum_{k\neq i} e^{-d^A_{ik}}} \qquad i \neq j \quad (7)$$

where $d^A_{ij} = D_A(X_i,x_j)$ and A is a positive semi-definite matrix. An ideal "bi-level" distribution expressing the ideal case where all data patterns from the same class are mapped to a single point and data patterns in other classes are separated as much as possible is used. To make the conditional distribution as close as possible to the ideal case, the KL divergence between the two distributions, the conditional distribution and "bi-level" distribution, is minimised. Gradient descent and iterative projections, similar to [3] is used to solve this convex optimisation problem.

4) Large Margin Nearest Neighbour: The Large Margin Nearest Neighbour (LMNN) technique [13] learns a Mahalanobis distance metric by enforcing the KNN classifier to always belong to the same class while data patterns that belong to other classes are separated by a large margin. A cost function is used to penalise large distances between each input $\mathbf{x}_i$ and its target neighbours and small distances between each input and all other inputs that do not belong to the same class as follows:

$$\begin{aligned}\varepsilon(\mathbf{L}) &= \sum_{ij}\eta_{ij}||\mathbf{L}(\mathbf{x}_i,\mathbf{x}_j)||^2 \\ &\quad +c\sum_{ijl}\eta_{ij}(1-y_{il})[1+||\mathbf{L}(\mathbf{x}_i,\mathbf{x}_j)||^2 \\ &\quad -||\mathbf{L}(\mathbf{x}_i,\mathbf{x}_l)||^2]_+ \end{aligned} \quad (8)$$

where $||\mathbf{L}(\mathbf{x}_i,\mathbf{x}_j)||^2$ is the squared distance between two data patterns, $y_{ij} \in 0,1$ indicates the labels of two data patterns, y_i and y_j match, $\eta_{ij} \in \{0,1\}$ indicates whether $\mathbf{x}_j$ is a target neighbour of $\mathbf{x}_i$, $[z]_+ = \max(z,0)$ in the second term denotes the standard hinge loss and $c > 0$ is a positive constant. The first term only penalises large distances between data patterns and target neighbours, and not between all data patterns that have similar labels. The second term incorporates the idea of a margin such that for each input $\mathbf{x}_i$, the hinge loss is incurred by differently labelled data patterns whose distances do not exceed the distance from $\mathbf{x}_i$ to any of its target neighbours by one absolute unit of distance. Thus, the cost function favours distance metrics in which differently labelled data maintain a large margin and do not threaten to to "invade" each other's neighbourhoods.

5) Differences: While the four techniques have the same objective of preserving the distance relation between data patterns, they take on different learning strategies. For GDML, Xing *et al.* [3] focused on minimising the distance between similar data patterns. NCA, MCML and LMNN are designed based on the KNN algorithm. For NCA, a distance metric is learned by finding a linear transformation of input data such that the average LOO classification performance of stochastic nearest neighbours is maximized in the transformed space. MCML attempts to learn a distance metric which map similar data patterns to a single point and dissimilar data far apart using "bi-level" distribution. LMNN introduces a cost function which punishes dissimilar data pattern with small distances, ensuring a large distance is maintained between dissimilar data.

B. Semi-supervised Fuzzy c-means

Let N, n, c and $\mathbf{U}$ denote number of data patterns, number of dimensions, number of clusters and partition matrix containing memberships of data patterns respectively. u_{ij} is membership value of data pattern j in cluster i, v_i is the cluster centre

Algorithm 1 Semi-supervised fuzzy c-means [7]

1: Initialise c, labelled data membership matrix F and initial membership matrix U^0

2: Calculate cluster centres using

$$v_i = \frac{\sum_{j=1}^{N} u_{ij}^2 \mathbf{x}_j}{\sum_{k=1}^{N} u_{ij}^2}, \ 1 \leq i \leq c. \tag{10}$$

3: Compute fuzzy covariance matrices.

4: Compute squared distances d_{ij}^2 between cluster centres and data patterns.

5: Update partition matrix, U using equation :

$$u_{ij} = \frac{1}{1+\alpha} \left\{ \frac{1 + \alpha(1 - b_j \sum_{l=1}^{c} f_{lj})}{\sum_{l=1}^{c} \left(\frac{d_{ij}}{d_{lj}}\right)^2} + \alpha f_{ij} b_j \right\} \tag{11}$$

6: If $||U' - U|| < \epsilon$, stop. Else, go to Line 2 with $U = U'$

TABLE I

DATASET SPECIFICATIONS SHOWING NUMBER OF DATA PATTERNS (N), NUMBER OF DIMENSIONS (n), NUMBER OF CLASSES (c) AND THE NUMBER OF FOLD (k) USED IN CROSS-VALIDATION (CV)

Dataset	N	n	c	k-fold CV
Nottingham Tenovus Breast Cancer (NTBC)	663	25	6	10
Wisconsin Original Breast Cancer (WOBC)	699	8	2	10
Arrhythmia	420	277	3	10
Pima Indian Diabetes (PID)	768	8	2	5
Cardiotocography	2126	21	3	10
Yeast	1484	8	10	2
Wisconsin Diagnostic Breast Cancer (WDBC)	569	30	2	10
Dermatology	366	33	6	5

(prototype) for cluster i, d_{ij} is distance between data pattern j and cluster centre v_i, fp is fuzzifier parameter, f_{ij} is membership value of labelled data pattern j in cluster i, $\mathbf{b}$ is a boolean vector indicating if a pattern is labelled and α is a parameter for maintaining balance between the supervised and unsupervised learning components.

In [7], the ssFCM objective function is as follows:

$$J = \sum_{i=1}^{c} \sum_{j=1}^{N} u_{ij}^{fp} d_{ij}^2 + \alpha \sum_{i=1}^{c} \sum_{j=1}^{N} (u_{ij} - f_{ij} b_j)^{fp} d_{ij}^2, \tag{9}$$

The algorithm uses labelled data as training examples to classify unlabelled data. The algorithm involves iteratively calculating the cluster centres and partition matrix to minimise the objective function until a termination criterion is satisfied. The algorithm is summarised in Algorithm 1.

III. EXPERIMENT

A. Datasets

The specifications of the datasets used are shown in Table I. For the Nottingham Tenovus Breast Cancer data, the 663 labels were obtained from [14], which are also used in our previous work [10]. For Arrhythmia, feature 14 is removed as it contains many missing values. Data patterns in class 2 to 15 have been combined together as class 2 as there is too little data patterns in classes 7, 8 and 11 to carry out 10-fold cross validation properly. 22 data patterns which are unclassified are classed as class 3. For Cardiotocography, we used the 3-class labels instead of the 10-class labels. As there is not enough data patterns from some classes to be split into 10 folds in Yeast, we carry out 2-fold cross validation. For Dermatology, age in column 34 is removed due to missing values.

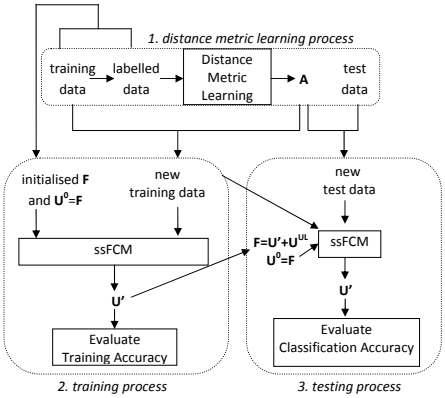


Fig. 1. Experimental set-up using distance metric learning with ssFCM

B. Methodology

We run our methodology in a cross validation (CV) setting using varying amounts of labelled data 10%, 20%, 30%, 40%, 50% and 60% of the training data, repeated 30 times for each amount. The labelled data are selected randomly based on stratified sampling to ensure each cluster is represented. Prior to running ssFCM, we first perform DML using the selected labelled data to learn $\mathbf{A}$, a positive-definite matrix, which is found in the Mahalanobis distance metric between two points x_i and x_j in the form:

$$d_{(i,j)} = (\mathbf{x_i} - \mathbf{x_j})^T \mathbf{A} (\mathbf{x_i} - \mathbf{x_j}). \tag{12}$$

The DML codes are implemented in MATLAB and are available online in [15], [16], [17].

Using $\mathbf{A}$, we transform the training and test data into new (transformed) training and test data. The transformation would cause the similar data patterns to become closer and/or the dissimilar data patterns are pushed further apart. We run ssFCM using the new training data in the training process. The updated partition matrix $\mathbf{U}'$ is then used with the new test data in the testing process to classify the unlabelled data. This experimental set-up is illustrated in Figure 1.

C. ssFCM set-up

α is set as N/M where M is the number of labelled data. In the original ssFCM [7], all data patterns are assigned memberships based on given labels and stored in $\mathbf{F}$. They are then selected to be labelled or unlabelled using the boolean vector $\mathbf{b}$ in (11). In our case, we select the labelled data and generate their memberships prior to running the algorithm. We set our $\mathbf{F} = \mathbf{U^0}$, which contain memberships of labelled and unlabelled data and b_j is 1 for all j (11). A high 0.9 membership value is arbitrarily chosen to indicate a data pattern's high possibility of belonging to the cluster while a 0.02 value indicates otherwise. The memberships are assigned as follows:

$$f_{ik} = \begin{cases} 0.9 & \text{if } x_j \text{ is labelled and in class } i \\ \frac{(1-0.9)}{(c-1)} & \text{if } x_j \text{ is labelled and not in class } i \\ 1/c & \text{if } x_j \text{ is unlabelled} \end{cases}$$

We assumed the number of cluster c to be the number of classes. Instead of the Mahalanobis distance metric with fuzzy covariance originally used [7], we experiment with Euclidean, Mahalanobis and kernel-based distance metrics instead as we found that the fuzzy Mahalanobis ssFCM did not always produce the best accuracy [10]. A one-tailed Mann-Whitney test

TABLE II

ACCURACY USING SSFCM WITH DML ON VARIOUS DATASETS IN A CV SETTING. EUCLIDEAN (E), MAHALANOBIS (M) OR KERNEL-BASED (K) INDICATE THE BEST PERFORMING DISTANCE METRIC FOR SSFCM

		10%	20%	30%	40%	50%	60%
NTBC	Euclidean (E)	96.12±2.04	96.86±1.94	97.22±1.77	97.54±1.61	97.64±1.55	97.84±1.53
	GDML+E	91.22±5.63	92.64±4.42	93.52±3.28	93.72±3.07	93.52±3.21	94.09±2.56
	LMNN+E	56.56±11.24	84.16±6.74	90.85±5.61	84.62±15.00	73.94±15.36	62.87±10.29
	MCML+E	87.31±5.43	88.66±5.08	89.23±4.55	89.66±4.66	89.69±4.41	90.60±4.29
	NCA+E	78.03±8.33	80.37±6.53	81.76±5.74	82.35±5.23	83.12±5.64	83.84±5.15
WOBC	Kernel-based (K)	96.22±1.67	95.98±1.56	95.94±1.54	95.76±1.41	95.72±1.38	95.71±1.37
	GDML+K	94.99±1.82	94.51±1.97	94.11±2.04	93.81±2.02	93.69±2.03	93.69±2.53
	LMNN+K	95.16±2.24	95.50±1.76	95.65±1.87	95.57±1.82	95.56±1.74	95.38±1.80
	MCML+K (4)	95.80±2.10	95.88±1.75	95.78±1.82	**95.85±1.64**	**95.77±1.59**	**95.79±1.53**
	NCA+K	93.99±3.92	94.72±2.31	94.72±2.36	95.13±2.02	94.90±1.88	95.27±1.70
Arrhythmia	Euclidean (E)	38.75±7.59	40.32±7.99	42.40±8.16	43.68±8.13	43.90±7.89	44.33±8.28
	GDML+E	**41.52±8.01**	**44.47±7.91**	**46.91±8.28**	**47.98±7.83**	**48.75±7.49**	**49.98±7.36**
	LMNN+E	*44.17±9.14*	*41.35±8.15*	41.02±8.17	43.68±7.90	*44.84±7.83*	*46.90±8.22*
	MCML+E	*39.89±7.89*	*41.46±8.61*	*43.58±8.44*	*44.67±8.12*	*44.69±7.65*	*45.25±8.13*
	NCA+E	*39.70±8.19*	*40.93±8.09*	*42.56±8.42*	42.92±8.55	43.87±8.55	43.78±8.19
PID	Mahalanobis(M)	72.53±2.80	72.95±2.58	73.46±2.54	73.90±2.77	73.87±2.67	74.11±2.57
	GDML+M	69.12±5.84	70.84±4.89	71.01±5.36	71.35±5.15	71.66±4.85	71.87±4.66
	LMNN+M	72.25±3.39	**73.00±2.71**	**73.54±2.55**	**73.99±2.72**	**74.22±2.78**	**74.34±2.73**
	MCML+M	72.52±2.82	72.95±2.59	73.45±2.54	73.90±2.78	73.87±2.67	74.12±2.58
	NCA+M	71.34±4.34	72.66±3.25	73.32±3.01	73.36±2.85	73.30±3.00	73.51±2.80
Cardiotocography	Euclidean (E)	47.60±3.91	48.92±3.00	49.31±3.03	49.94±3.05	50.44±3.08	51.18±3.00
	GDML+E	**63.41±8.22**	**68.58±6.67**	**71.06±5.45**	**74.46±5.11**	**74.33±4.03**	**75.89±4.31**
	LMNN+E	*50.76±6.45*	*50.98±5.03*	*52.25±5.24*	*52.84±4.88*	*54.07±4.90*	*55.12±5.14*
	MCML+E (10)	47.58±6.58	*49.23±5.98*	*49.81±6.24*	*50.99±6.54*	*51.21±6.64*	*51.60±6.33*
	NCA+E	*51.77±9.86*	*54.12±8.44*	*56.28±8.34*	*56.52±7.94*	*57.88±8.43*	*60.83±8.90*
Yeast	Euclidean (E)	33.34±3.51	35.28±2.99	36.94±2.72	37.67±3.05	38.06±2.66	38.21±2.55
	GDML+E	29.06±5.39	29.92±4.79	30.83±3.60	31.43±4.16	31.55±4.09	32.28±4.81
	LMNN+E	*34.54±3.14*	*37.13±2.75*	*38.26±2.69*	*39.48±2.98*	*40.39±2.75*	*40.38±2.64*
	MCML+E	**37.66±3.76**	**39.55±3.11**	**41.24±3.14**	**42.26±2.80**	**42.81±2.90**	**43.66±2.35**
	NCA+E	25.92±2.79	27.62±2.79	28.26±3.05	28.52±3.25	28.88±3.95	27.95±3.31
WDBC	Mahalanobis (M)	89.03±4.70	90.41±4.48	88.73±5.74	84.70±4.68	85.70±5.40	91.01±5.59
	GDML+M	**89.60±4.87**	**91.26±4.64**	**92.29±4.03**	**92.87±4.23**	**93.10±3.83**	*92.71±4.13*
	LMNN+M	84.87±5.15	88.88±4.68	*90.93±4.13*	*92.30±3.76*	*92.89±3.91*	**93.64±3.44**
	MCML+M (20)	88.83±4.80	*90.78±4.35*	*92.01±4.30*	*92.36±4.01*	*92.88±4.13*	*93.22±3.94*
	NCA+M	87.47±5.15	89.78±4.64	*90.87±4.65*	*91.33±4.30*	*91.67±4.74*	*92.24±3.98*
Dermatology	Euclidean (E)	94.33±2.18	94.54±2.15	94.61±2.11	94.63±2.12	94.63±2.02	94.62±1.92
	GDML+E	92.81±3.11	93.25±2.54	93.32±2.64	93.57±2.66	93.92±2.39	93.97±2.46
	LMNN+E	80.61±7.13	89.32±4.13	92.59±2.86	*94.74±2.13*	*95.28±2.12*	*95.58±1.85*
	MCML+E (20)	92.51±3.19	93.95±2.57	**95.22±2.30**	**95.60±2.44**	**95.73±2.39**	**96.04±2.04**
	NCA+E	89.92±4.47	91.67±4.38	91.55±4.25	91.14±4.64	91.72±4.07	92.21±3.28

[18] is used to demonstrate significant improvement between using ssFCM and DML with ssFCM (with p-value < 0.05).

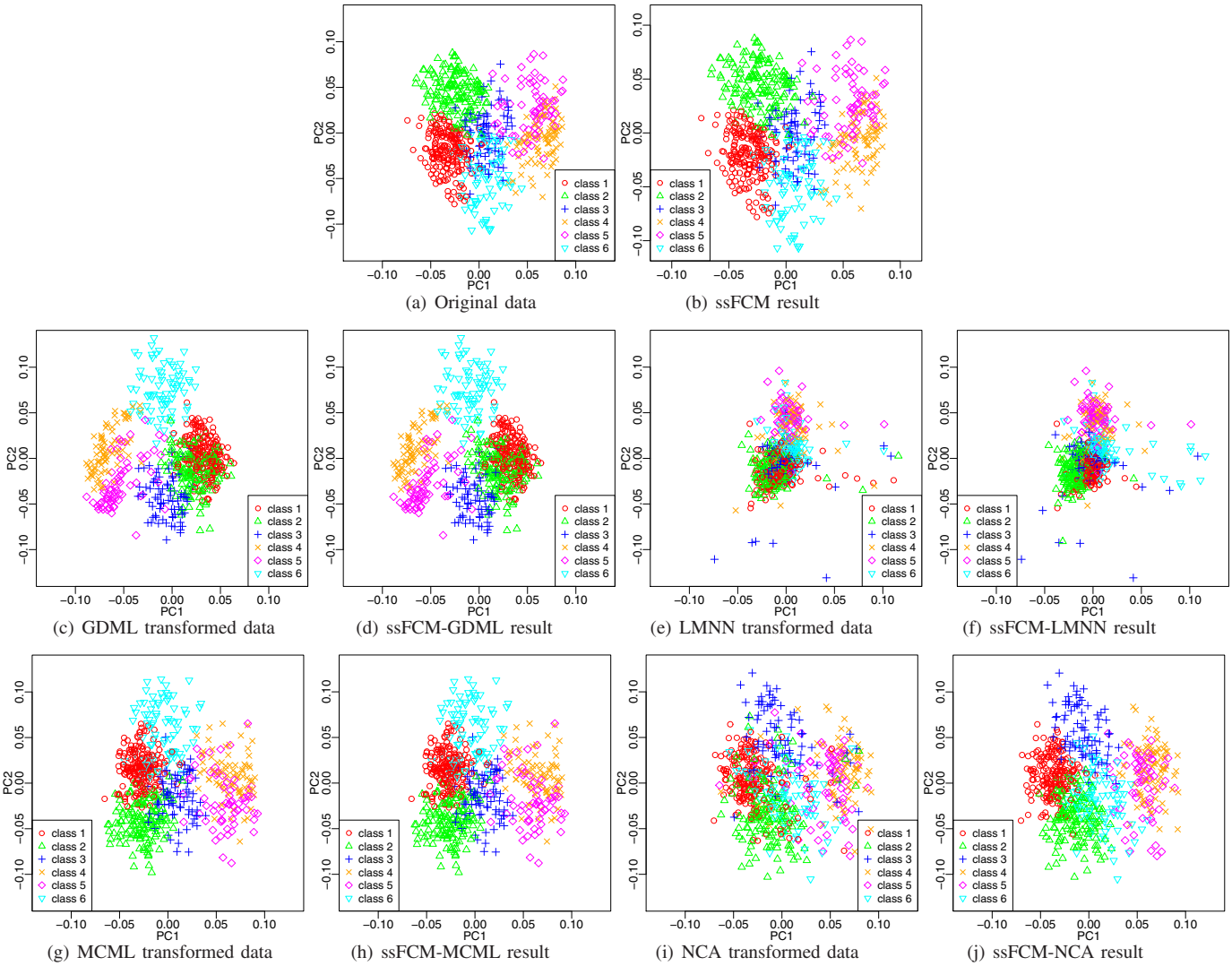
IV. RESULTS

Table II shows the accuracy of test data using ssFCM with DML. Only ssFCM results with the best performing distance metric is shown. The best results are highlighted in bold and results which are better than ssFCM are italicised. Results that are underlined indicate statistically significant improvement in accuracy when compared to using ssFCM alone. The accuracy is calculated based on the average number of correct assignments presented in percentage followed its standard deviation.

Out of the eight datasets used to test our methodology, significant improvement using ssFCM with DML was found on five datasets, particularly for Cardiotocography, WDBC and Dermatology. While improvement was found on WOBC and PID, it is considered not statistically significant. This is not to say there is no improvement but, the improvement is not significant. For the NTBC dataset, we observed no improvement at all using ssFCM with DML. ssFCM with LMNN showed significant improvement for more datasets than ssFCM with GDML, on five out of eight datasets. ssFCM with GDML produced the highest significant improvement for three datasets. ssFCM with DML appears to show improvement when the amount of labelled data are at least 20%, 30% or 40% of training data for Dermatology, PID and WOBC respectively. This was also observed in WDBC using LMNN, MCML and NCA.

For dataset WOBC, PID, Cardiotocography, WDBC and Dermatology where MCML did not improve ssFCM clustering results, we experiment with reduced number of dimensions indicated by a numerical in brackets specified in Table I. For PID, experimentation using ssFCM and MCML with 4 dimensions was conducted but the results did not improve, and was thus not shown.

Principal Component Analysis (PCA) is used to present a 2D projection view of the resulting DML transformation on the original datasets and the clustering results based on the transformed data. Due to space constraints, we show only the datasets with the worst and best results NTBC, Dermatology and Cardiotocography, as shown in Figure 2, 3 and 4 respectively. By worst, we mean the dataset where ssFCM produced good results but, worsen drastically using ssFCM with DML. The DML-ssFCM results displayed are based on one of the runs after training and testing using 60% labelled data. Note



(a) Original data (b) ssFCM result

(c) GDML transformed data (d) ssFCM-GDML result (e) LMNN transformed data (f) ssFCM-LMNN result

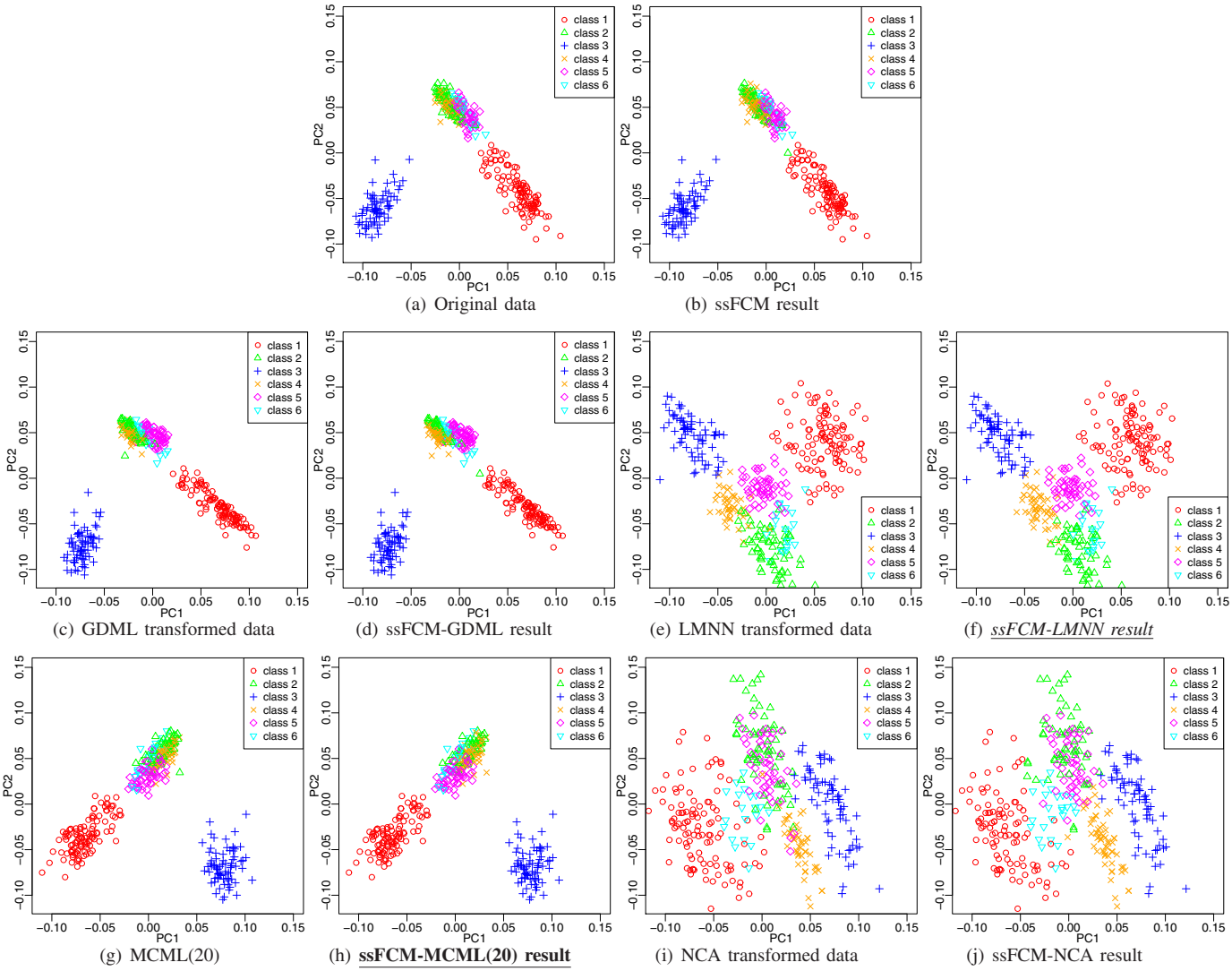
(g) MCML transformed data (h) ssFCM-MCML result (i) NCA transformed data (j) ssFCM-NCA result

Fig. 2. PCA biplots to show DML transformation with original labels in (c), (e), (g), (i) and ssFCM clustering results in (d), (f), (h), (j) for NTBC dataset

that no clustering using PCA is performed. PCA is solely used to extract the first two principal components of the original and transformed datasets for providing visualisation. To reflect the findings from Table II on the figures, the best performing ssFCM with DML methods for the dataset are indicated in bold, and those with improvement italicised and those that are statistically significant are underlined. Unfortunately, the 2D projection does not always show exactly the clusters DML or ssFCM recognise in terms of cluster separability. For instance, in Figure 3, although transformation by LMNN shows better cluster separability than MCML (MCML has not push apart clusters for class 2, 4, 5 and 6), ssFCM-MCML on average performed better than ssFCM-LMNN. Despite this limitation, the general idea of DML for projecting distance relations can still be studied.

To further analyse the DML transformation, the sums of squared Euclidean distances (SSD) of a train set between similar data (data with same labels) and between dissimilar data (data with different labels), before and after DML (using 60% of labelled data) transformation are also studied, shown in Table III. The factor SSD is reduced or increased by after DML transformation is also calculated. Ideally, the larger the factor SSD for similar data is reduced by after DML transformation, the more compact the clusters and thus, further away from other clusters. The train datasets transformed after DML used in these calculations correspond to the same ones presented in Figures 2, 3 and 4 (c), (e), (g) and (i) for the respective DML and datasets.

For NTBC in Figure 2, it appears that GDML increase the separability of the three known main groups [14], where classes 1-3 belong to one group, classes 4-5 to another and finally class 6. The classes within these three main groups can be observed to be made closer by GDML. But, separability between classes within a group does not appear to have increased. Similar observation is found for GDML on Dermatology based on PCA projection in Figure 3(c). Classes 2 and 4-6 appear to belong to a group. However, unlike in NTBC, it can be observed that GDML puts the similar data closer for these classes (classes 2 and 4-6 of Dermatology) as they appear to overlap less as compared to the original data in Figure 3(a). For Cardiotocography in Figure 4(c), we observed that class 2 is better separated from class 1, and class 3 from the other two



(a) Original data (b) ssFCM result

(c) GDML transformed data (d) ssFCM-GDML result (e) LMNN transformed data (f) *ssFCM-LMNN result*

(g) MCML(20) (h) **ssFCM-MCML(20) result** (i) NCA transformed data (j) ssFCM-NCA result

Fig. 3. PCA biplots to show DML transformation with original labels in (c), (e), (g), (i) and ssFCM clustering results in (d), (f), (h), (j) for Dermatology dataset

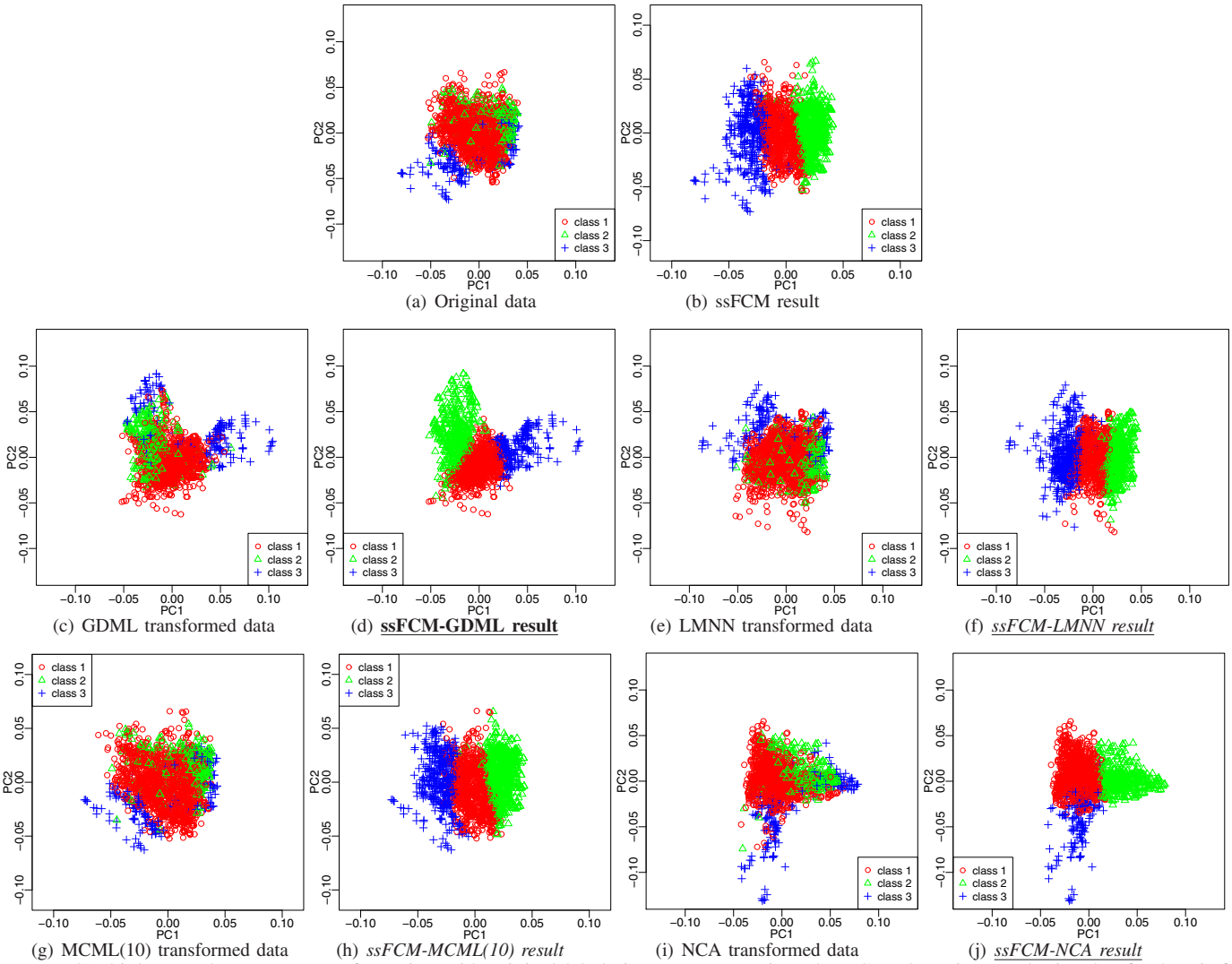
classes. These observations based on GDML transformation is also reflected in Table III where the SSD of data with same labels in the train set is reduced by a greater factor than for SSD of data with different labels.

For LMNN on NTBC , shown in Figure 2 (e), the separability between the clusters have reduced, causing clusters to overlap. This observation is also reflected in Table III where SSD of data with different labels are reduced by a greater factor than SSD for data with the same labels. This means that data with different labels are placed closer together than data with same labels. This led to the poorer ssFCM accuracy as compared to GDML+E in Table II. However, on Dermatology and Cardiotocography, LMNN reduced SSD of similar data by a greater factor than SSD of data with different labels. In Figure 3(e), LMNN was able to discriminate between classes 2, 4 and 5, increasing separability between these classes as compared to GDML and MCML. LMNN

For MCML, it is not visually obvious whether the separability between clusters have increased based on the PCA 2D projections. For NTBC in Figure 2(g), it can be observed that class 6 is slightly further away from class 3 as compared to the original data. For Dermatology, it can be observed that data with same labels in classes 2 and 4-6 are placed closer together as compared to the original data. Both these observations are consistent with SSD results in Table III. For Cardiotocography, however, SSD of data with same labels are reduced by a smaller factor than SSD of data with the different labels using MCML, causing poor ssFCM accuracy in Table II.

For NCA, the change in between cluster separability can be observed in the PCA projections and SSD analysis. While ssFCM with NCA performed moderately well in comparison with ssFCM with other DML techniques, particularly in Cardiotocography and WDBC, it did not produce the best significantly improved results.

For Cardiotocography, while the DML techniques were able to put data with the same labels closer together for class 1 and 2, class 3 remained split in two. Based on the 2D projections in Figure 4, the clustering results from ssFCM and ssFCM with LMNN and MCML appear wrong. 3D projections are required to further analyse the results.



(a) Original data (b) ssFCM result

(c) GDML transformed data (d) **ssFCM-GDML result** (e) LMNN transformed data (f) *ssFCM-LMNN result*

(g) MCML(10) transformed data (h) *ssFCM-MCML(10) result* (i) NCA transformed data (j) *ssFCM-NCA result*

Fig. 4. PCA biplots to show DML transformation with original labels in (c), (e), (g), (i) and ssFCM clustering results in (d), (f), (h), (j) for Cardiotocography dataset

V. Discussion

DML has been shown to significantly improve ssFCM clustering for five out of the eight datasets tested. For datasets such as WDBC and Dermatology, significant improvement was found when the amount of labelled data are at least 20% of training data. This suggests that for some datasets, a larger amount of labelled data is required for DML to improve ssFCM clustering. All DML used were able to reduce the SSD of similar data by a greater factor than dissimilar data, for most datasets.

No improvement to ssFCM clustering using ssFCM with DML on the NTBC. Based on observation of 2D projections on the DML transformed data, the subgroups belonging to the same main group are being placed closer together. This type of hierarchical grouping with main groups and subgroups appears to be a challenge for DML techniques. ssFCM performed with high accuracy on NTBC but, DML actually worsens ssFCM clustering on this dataset. We suspect that ssFCM with LMNN (as well as other DML techniques) performed poorly on NTBC due to the hierarchical nature of its classes.

Interestingly, the contrary was found with Dermatology where classes 2 and 4-6 appear to belong to a larger common group on the 2D projection. LMNN was able to separate these classes well. Similarly for NCA, it could separate the classes 2, 4-6 in Dermatology but, causes more overlapping for classes in NTBC. The classes of NTBC appear to have a conflicting effect on DML transformation. Further investigation is required to study this effect.

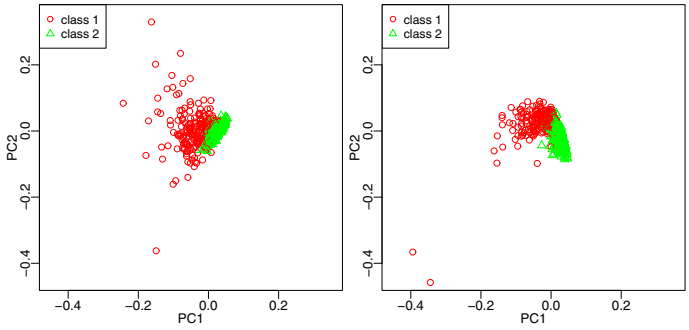
ssFCM with MCML have produced some of the best results, such as for Yeast and Dermatology. But, MCML is found to be the least visually informative as compared to the other DML techniques on 2D projection.

We found that when SSD of similar data is reduced by a factor smaller than the SSD of data with different labels, this usually indicates that the DML will not improve ssFCM for that dataset. This was observed in LMNN for NTBC and in MCML(10) for Cardiotocography. This may seem an intuitively obvious way to check if a particular DML will improve clustering for a dataset. But, it is not a foolproof method, as was found with LMNN for WDBC. For WDBC, ssFCM with LMNN produced significant improvement. But, based on the analysis of SSD where the factor SSD of similar data after DML transformation is reduced by (4.5E+02) is found smaller than for

TABLE III

SUM OF SQUARED EUCLIDEAN DISTANCES FOR DATA WITH SAME (S) AND DIFFERENT (D) LABELS BEFORE AND AFTER DML TRANSFORMATION AND SSD CHANGE FACTOR (BEFORE÷AFTER)

			before DML	after DML	factor
NTBC	GDML	S	9.2E+09	6.5E+09	1.42
		D	8.0E+10	7.9E+10	1.01
	LMNN	S	9.2E+09	3.6E+07	255
		D	8.0E+10	2.1E+08	388
	MCML	S	9.2E+09	6.1E+09	1.50
		D	8.0E+10	5.7E+10	1.40
	NCA	S	9.2E+09	7.6E+04	1.2E+05
		D	8.0E+10	9.4E+05	8.5E+04
Dermatology	GDML	S	3.7E+05	2.8E+05	1.33
		D	3.9E+06	4.4E+06	0.881
	LMNN	S	3.7E+05	2.3E+04	15.7
		D	3.9E+06	5.7E+05	6.77
	MCML(20)	S	3.7E+05	1.9E+05	1.94
		D	3.9E+06	3.2E+06	1.21
	NCA	S	3.7E+05	1.5E+06	0.247
		D	3.9E+06	2.6E+07	0.149
Cardiotocography	**GDML**	S	1.8E+10	1.4E+10	1.33
		D	1.8E+10	1.9E+10	0.965
	LMNN	S	1.8E+10	3.2E+07	580
		D	1.8E+10	3.2E+07	563
	MCML(10)	S	1.8E+10	1.3E+10	1.43
		D	1.8E+10	1.2E+10	1.48
	NCA	S	1.8E+10	6.1E+09	3.04
		D	1.8E+10	8.4E+09	2.17



(a) Original data (b) LMNN-transformed data

Fig. 5. PCA biplots to show of original data (a) and LMNN-transformed data (b) for WDBC

dissimilar data (7.1E+02)), this would suggest poor clustering results. On analysis of the 2D projection in Figure 5, the class 1 data patterns have been placed closer together, producing a more compact cluster and thus, improving clustering results. Thus, it is crucial to conduct both analysis using visual projections and SSD, before and after DML transformation to study the DML effects in clustering techniques.

VI. CONCLUSION

The NTBC and seven popular UCI datasets have been tested using ssFCM with DML techniques. Comparison between accuracy of test results are conducted and statistical tests from using DML in ssFCM are analysed. ssFCM with DML was found to produce significant improvement to ssFCM clustering for five out of the eight datasets. Furthermore, DML was found not to always improve ssFCM clustering on datasets with hierarchical clusters such as the NTBC.

Based on our observation using comparisons of ssFCM accuracy, SSD measures and 2D projection between different DML techniques, we found that further information about the internal structure that are useful for clustering can be gained. In fact, the SSD measures and 2D projection provide important analysis as further support to findings in the ssFCM accuracy.

As preliminary work in the application of DML to ssFCM, these findings are considered promising. As our future work, we hope to perform 3D projections on the DML transformed datasets to get a better view of how the separability between clusters have increased or decreased. To further support our findings, DML transformations on datasets with hierarchical clusters need to be further investigated.

REFERENCES

[1] L. Yang and R. Jin, "Distance metric learning: A comprehensive survey," *Michigan State Universiy*, pp. 1–51, 2006.

[2] J. Reps, J. Garibaldi, U. Aickelin, D. Soria, J. Gibson, and R. Hubbard, "A novel semi-supervised algorithm for rare prescription side effect discovery," *IEEE Journal of Biomedical and Health Informatics*, vol. preprint, no. 99, pp. 1–1, 2013.

[3] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning, with application to clustering with side-information," *Advances in Neural Information Processing Systems*, vol. 15, pp. 505–512, 2002.

[4] M. Ceccarelli and A. Maratea, "Improving fuzzy clustering of biological data by metric learning with side information," *International Journal of Approximate Reasoning*, vol. 47, no. 1, pp. 45 – 57, 2008.

[5] G. Beliakov, S. James, and G. Li, "Learning choquet-integral-based metrics for semisupervised clustering," *Fuzzy Systems, IEEE Transactions on*, vol. 19, no. 3, pp. 562–574, 2011.

[6] X. Yin, T. Shu, and Q. Huang, "Semi-supervised fuzzy clustering with metric learning and entropy regularization," *Knowledge-Based Systems*, vol. 35, no. 0, pp. 304 – 311, 2012.

[7] W. Pedrycz and J. Waletzky, "Fuzzy clustering with partial supervision," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 27, no. 5, pp. 787–795, May 1997.

[8] C. Stutz and T. A. Runkler, "Classification and prediction of road traffic using application-specific fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 3, pp. 297–308, June 2002.

[9] L. Tari, C. Baral, and S. Kim, "Fuzzy c-means clustering with prior biological knowledge," *Journal of Biomedical Informatics*, vol. 42, no. 1, pp. 74–81, 2009.

[10] D. T. C. Lai and J. M. Garibaldi, "Improving semi-supervised fuzzy c-means classification of breast cancer data using feature selection," in *Proceedings of IEEE International Conference on Fuzzy Systems*, 2013, pp. 1–8.

[11] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," *Advances in Neural Information Processing Systems*, pp. 513–520, 2004.

[12] A. Globerson and S. Roweis, "Metric learning by collapsing classes," *Advances in Neural Information Processing Systems*, vol. 18, pp. 451–458, 2006.

[13] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Advances in Neural Information Processing Systems*, pp. 1473–1480, 2006.

[14] D. Soria, J. M. Garibaldi, F. Ambrogi, A. R. Green, D. Powe, E. Rakha, R. D. Macmillan, R. W. Blamey, G. Ball, P. J. Lisboa, T. A. Etchells, P. Boracchi, E. Biganzoli, and I. O. Ellis, "A methodology to identify consensus classes from clustering algorithms applied to immunohistochemical data from breast cancer patients," *Computers in Biology and Medicine*, vol. 40, no. 3, pp. 318–330, 2010.

[15] "GDML's matlab code," Last assessed: 10th September 2013. [Online]. Available: http://www.cs.cmu.edu/~epxing/papers/Old_papers/code_Metric_online.tar.gz

[16] "LMNN's matlab code," Last assessed: 10th September 2013. [Online]. Available: http://www.cse.wustl.edu/~kilian/code/code.html

[17] "NCA and MCML's matlab code," Last assessed: 10th September 2013. [Online]. Available: http://cseweb.ucsd.edu/~lvdmaaten/dr/download.php

[18] R. C. Team and contributors worldwide, "Wilcoxon rank sum and signed rank tests," 2013 Last assessed: 28th November 2013. [Online]. Available: http://stat.ethz.ch/R-manual/R-patched/library/stats/html/wilcox.test.html