2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Multidimensional Scaling with Multiswarming

Thomas A. Runkler
Siemens AG
Corporate Technology
Otto-Hahn-Ring 6, 81739, Munchen, Germany
thomas.runkler@siemens.com

James C. Bezdek
Dept. of Electrical and Electronic Engr.
University of Melbourne
Melbourne, Victoria, 3053, Australia
jcbezdek@gmail.com

***Abstract*— We introduce a new method for multidimensional scaling in dissimilarity data that is based on preservation of metric topology between the original and derived data sets. The model seeks neighbors in the derived data that have the same ranks as in the input data. The algorithm we use to optimize the model is a modification of particle swarm optimization called multiswarming. We compare the new method to three well known approaches: Principal component analysis, Sammon's method, and (Kruskal's) metric MDS. Our method produces feature vector realizations that compare favorably with the other approaches on three real relational data sets.**

Keywords*— *metric topology preservation, multidimensional scaling, multiswarm optimization, Sammon's algorithm

I. INTRODUCTION

Let $O = \{o_1, o_2, \ldots, o_n\}$ be a set of n objects. Each object is a physical entity (a soccer player, fish, guitar, type of beer, etc.). When $o_i \in O$ has a *physical label*, O is *labeled data*; otherwise, O is unlabeled. The objects in O can be represented by *feature data* $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \Re^p$, or by relational data R on $O \times O$. In the first case, object o_j has vector $\mathbf{x}_j$ as its numerical representation; x_{jk} is the k-th feature (or attribute) associated with object j. In the second case, we have a relation $\rho : O \times O \mapsto \Re$ whose n^2 values $\{\rho(o_i, o_j)\}$ are arrayed as relation matrix $R = [r_{ij}] = [\rho(o_i, o_j)]$. In this note we deal with dissimilarity data D, which exhibit pairwise dissimilarities on n vectors $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \Re^p$ The elements of D may have four properties:

$$p1: \ \sqrt{d_{ij}} \geq 0 \ \forall \ i, j \tag{1a}$$

$$p2: \ \sqrt{d_{ii}} = 0 \Leftrightarrow \mathbf{x}_i = \mathbf{x}_j \tag{1b}$$

$$p3: \ \sqrt{d_{ij}} = \sqrt{d_{ji}} \ \forall \ i \neq j \tag{1c}$$

$$p4: \ \sqrt{d_{ij}} \leq \sqrt{d_{ik}} + \sqrt{d_{kj}} \ \forall \ i \neq j \neq k \tag{1d}$$

D is said to be *metric* when its entries satisfy (1a)-(1d). $M_n = \{D = [d_{ij}] \in \Re^{nn} : (1a) - (1d) \text{ hold}\}$ is the set of all metric matrices, $M_n^+ = \{D = [d_{ij}] \in \Re^{nn} : (1a) - (1c) \text{ hold}\}$ is the set of positive, hollow, symmetric matrices. $D \in M_n$ is a *Euclidean distance matrix* (EDM) if and only if there is a set of vectors $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ in some Euclidean space $\Re^q$, $q \leq n-1$, such that $d_{ij} = (\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ between pairs of vectors in X is *squared* Euclidean distance. When D is Euclidean, X is called a *realization* of D. The *smallest* q for which there is a Euclidean realization of D is the *embedding dimension*. D is metric when it is Euclidean, but not necessarily conversely, so the set of Euclidean matrices M_n^e, is a proper subset of M_n, $M_n^e \subset M_n$. The literature on EDMs is not consistent with respect to this terminology: some authors use *unsquared* distances in the definition. We use the squared definition and theory as given, for example, in [1-3]; see [4-6] for the alternate formulation using unsquared distances. The two formulations are equivalent. but it is very important to make sure which type of D you are dealing with when performing a cluster analysis or multidimensional scaling using D as input.

Let $\Re_+^{pp}$ be the set of *positive definite matrices* in $\Re^{pp}$. For vectors $\mathbf{x}, \mathbf{v} \in \Re^p$ and $A \in \Re_+^{pp}$, the inner product distance is $d_A(\mathbf{x}, \mathbf{v}) = \|\mathbf{x} - \mathbf{v}\|_A = \sqrt{(\mathbf{x} - \mathbf{v})^T A(\mathbf{x} - \mathbf{v})}$. When A is the identity matrix, $d_A(\mathbf{x}, \mathbf{v})$ is Euclidean distance. $D_A \in \Re^{nn}$ is *A-Euclidean* if there is an $A \in \Re_+^{pp}$ and a set of feature vectors $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \Re^p$ such that $D_A = \left[d_A(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_A^2 ; \ 1 \leq i, j \leq n \right]$. It is not hard to show that D is Euclidean if and only if it is A-Euclidean. Thus, $D_A \in M_n^e \ \forall \ A \in \Re_+^{pp}$.

Let $HP(0, \mathbf{1}) = \{\mathbf{z} \in \Re^n : \langle \mathbf{z}, \mathbf{1} \rangle = 0\}$ denote the (n-1) dimensional hyperplane through the origin of $\Re^n$ that is perpendicular to the vector $\mathbf{1}^T = (1, 1, \ldots, 1) \in \Re^n$. Schoenberg [7] proved that D is Euclidean *if and only if* (iff) D is *negative semi-definite* (n.s.d.) on this hyperplane, i.e., iff $\mathbf{z}^T D \mathbf{z} \leq 0 \ \forall \ \mathbf{z}$ in $HP(0, \mathbf{1})$. (cf. p. 418, eqn 9.19 in [1]). This condition is difficult to verify, so we instead look at the eigenstructure of the matrix PDP, where $\hat{D} = [\hat{d}_{ij}] = [d_{ij}^2]$ and the *centering matrix*, $P = I - 1/n[\mathbf{1}\mathbf{1}^T]$, is the orthogonal projector of $\Re^n$ onto $HP(0, \mathbf{1})$. P has only 2 distinct

eigenvalues: $\lambda_1 = 1,\ \text{mult.} = n-1; \lambda_2 = 0,\ \text{mult.} = 1$, so the spectrum of P is $\Lambda = \{\underbrace{1,1,\ldots,1}_{n-1},0\}$.

Theorem 1 [cf. eqn. (947), p. 425 in [1]]. Let $P = I - 1/n[\mathbf{1}\mathbf{1}^T]$, $D \in M_n$, and $\hat{D} = [\hat{d}_{ij}] = [d_{ij}^2]$. Then

$$D \in M_n^e \Leftrightarrow -P\hat{D}P \text{ is } \textit{negative semidefinite} \text{ (n.s.d.)} \quad (2)$$

If $-P\hat{D}P$ has one or more positive eigenvalues, D is not Euclidean. The *number* of strictly negative eigenvalues of $-P\hat{D}P$ equals the (minimum) dimension s required for a realization of D. Note especially that the test in Theorem 1 uses $\hat{D}$, the Hadamard product of D with itself. This presents a small dilemma for the user: given a dissimilarity or distance matrix D, how do you know if its entries are already squared? You will be certain only if you know that D is constructed by computing pairwise squared distances between pair of vectors in $X = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\} \subset \Re^p$. If D arises from a transformation of similarity matrix S, e.g., $D = \mathbf{1}\mathbf{1}^T - S$, its entries will not be squares, and you will need to build $\hat{D}$ as in Theorem 1. In all other cases, you won't be certain. References [4-6] assume that D is NOT squared, and present the test in this alternate, equivalent form

Theorem 2 [Theorem 4 of [6]]. Let $P = I - 1/n[\mathbf{1}\mathbf{1}^T]$, $D \in M_n$, and $\hat{D} = [\hat{d}_{ij}] = [d_{ij}^2]$. Then

$$D \in M_n^e \Leftrightarrow -P\hat{D}P/2 \text{ is } \textit{positive semidefinite} \text{ (p.s.d.)} \quad (3)$$

The eigenvalues of $-P\hat{D}P/2$ are (-1/2) times the eigenvalues of $P\hat{D}P$. The constant (-2) multiplies the cross product of the inner product $\langle \mathbf{x}_i - \mathbf{x}_j, \mathbf{x}_i - \mathbf{x}_j \rangle$, and is carried to the other side of the equation for the Gram Matrix B in classical multidimensional scaling (cf. Section IIIA).

When D is non-Euclidean, $D \in (M_n - M_n^e)$, there are several transformations $\Psi : (M_n - M_n^e) \mapsto M_n^e$ which convert D from metric to Euclidean, $\Psi(D) \in M_n^e$. Benasseni et al. [8] discuss some methods for transformation of a metric D to a Euclidean D.

II. Multidimensional Scaling

Let $D \in M_n$. The basic idea in *multidimensional scaling* (MDS) is to find a set $Y = \{\mathbf{y}_1,\ldots,\mathbf{y}_n\} \subset \Re^q$ (with specified dimension q) so that the found dissimilarities $\{d'(\mathbf{y}_i, \mathbf{y}_j) = \tau(d_{ij}) : 1 \le i, j \le n\}$ between pairs of vectors in Y, arrayed as D' (approximately) match those in D, i.e., $D \approx D'$. When q = 1 the problem is called *unidimensional* scaling. Schoenberg [7] and Young and Householder [9] planted the seeds of MDS in 1938. Torgerson [10] germinated the idea in 1952. Rapid growth into a mature plant over the next 30 years is summarized in Davison [11]. Many relatives of MDS are presented in recent texts [5,12].

III. Five Methods for MDS

Two main types of MDS are distinguished by the type of objective function used: metric and non-metric [5]. It is easy to confuse these terms. Metric and non-Metric MDS both fit a metric to the data. Metric MDS refers to the case where J(D, D') is a least-squared error criterion, while non-metric MDS uses an objective function that assesses only ordinal values.

A. Classical metric MDS, $\mathbf{D \in M_n^e}$. Is there an input D for which an *exact* solution of MDS (D = D') is guaranteed? Yes. If D is Euclidean, and we choose the function τ as the identity, we can construct Y as in Section II.A, Y is a (non-unique) realization of D, and q is the *embedding dimension* for D. When D is Euclidean, X can be completely recovered, up to rotation, translation and reflection, about the origin. The solution is well known [5]. Let $B = \left[\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right] = XX^T$ denote the Gram matrix of inner products corresponding to $X = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\} \subset \Re^p$. The elements of B are related to D using the centering matrix P, $B = -P\hat{D}P/2 = XX^T$. B has rank p, with p positive eigenvalues $\Lambda_p = \{\lambda_1 \ge,\ldots,\ge \lambda_p\}$ and corresponding orthonormal eigenvectors $V_p = \{\mathbf{v}_1,\ldots,\mathbf{v}_p\}$. Define the matrix $\Lambda = \text{diag}\left[\sqrt{\lambda_1},\ldots,\sqrt{\lambda_p}\right]$, array the p eigenvectors as columns of matrix V. The spectral decomposition of B then yields $B = V\Lambda V^T$. The columns of the matrix $X = V\sqrt{\Lambda}$ are the recovered vectors, unique up to rotation, translation, and reflection about the origin. The solution in the Euclidean case is equivalent to principal component analysis [5], which we will use as one of the four methods in this article. In other cases, the search for Y is often guided by an objective function J(D, D') which assesses the goodness of fit between D and D'. Different cases of MDS arise by choosing different J's, and different algorithms to optimize J.

B. Classical metric MDS, $\mathbf{D \in M_n}$. If D is metric, but not Euclidean, construct B as in method A. If rank(B) = p, we again have $B = V\Lambda V^T = XX^T$, and can recover X as before. An algorithm for cases *A* and *B* is given on p. 38 of Cox and Cox [5].

C. Metric MDS, $\mathbf{D \in M_n}$. An alternative to *A* and *B*, whether D is Euclidean or metric, is based on optimizing an objective function J(D,D'). Why seek an alternative? Two reasons: (i) for large n, finding the spectral decomposition of B is a difficult and notoriously unstable numerical problem; and (ii) even if the transformation of D into Eucidean form, say $\hat{D} \in M_n^e$, is feasible, the vectors $\hat{Y}$ found by applying method *A* to $\hat{D}$ may be a considerable distortion of the vectors Y obtained as a realization of D. Since our objective in applying MDS to D is to find a reasonable visual representation that may help us interpret cluster substructure in the *input data*, we turn to another method.

Sammon's model [13] looks for a set $Y_S \subset \Re^q$, q < p, whose elements have the same pairwise distances as their pre-images in X. Let $D = [d_{ij}]$ be a distance matrix for X in $\Re^p$ and $D' = [d'_{ij}]$ be a distance matrix whose ij-th entry is the distance between the (unknown) vectors $\mathbf{y}_j = A_S(\mathbf{x}_j)$ in $\Re^q$, where $A_S : \Re^p \mapsto \Re^q$ is our notation for any algorithm that attempts to solve Sammon's optimization problem

$$\underbrace{\min}_{Y \subset \Re^q} \left\{ J_S(D' \mid D) = \frac{1}{\sum_{i<j} d_{ij}} \sum_{i<j} \frac{(d_{ij} - d'_{ij})^2}{d_{ij}} \right\}. \quad (4)$$

Our notation emphasizes that D is fixed (possibly by X), while Y (i.e., D') is variable. $J_S(D' \mid D) = 0 \Leftrightarrow (D' = D)$. When J_S is zero all n(n-1)/2 distances are preserved *exactly*. Thus, A_S attempts to find a Y_S that is *isometric* to X. This is a *stronger* property than the one sought by our new MTP-MSO algorithm, which has the less ambitious objective of finding Y_ρ^* that preserves *neighbor ranks*. There are two cases of (4) depending on the type of input data (D or X). Both cases choose a function η' to convert Y_S into D', and both are initialized by guessing a first set Y_{S0} and computing $D'_0 = \eta'(Y_{S0})$. If the input data is X, we also must choose η so that $D = \eta(X)$. It would be unusual – but not unheard of - in this case to choose $\eta \neq \eta'$. After guessing an initial Y_{S0}, Sammon used the method of steepest descent as A_S to iteratively minimize J_S. In this paper we use Newton's method to minimize J_S.

D. Non-Metric MDS, $\mathbf{D} \in \mathbf{M_n}$. There are many forms for this type of MDS [5]. We will use Kruskal's original version of nonmetric MDS [14]. In this case J(D, D') is an objective function that assesses ordinal values. To begin, reindex the dissimilarities in $D = [d_{ij}]$ to the vector **d** which has entries $d_k = d_{ij}$, where $k_{ij}=(i-1)((2n-i)/2)+(j-i)$. Relabel the entries of D' to get a vector **d'** in the same way. Because D and D' are symmetric, and have zero diagonals, these vectors are in $\Re^T$, $T = n(n-1)/2$. Kruskal's stress function defines the optimization problem

$$\underbrace{\min}_{Y \subset \Re^q} \left\{ J_K(D' \mid D) = \sum_{k_{ij}=1}^{T} (d_{k_{ij}} - d'_{k_{ij}})^2 \Big/ \sum_{k=1}^{T} d_{k_{ij}} \right\}. \quad (5)$$

From **d** and **d'** compute the rank vectors **r** and **r'**. It is convenient to represent these conversions as **r** = F(D), **r'**=F(D'), F denoting the composition of the two operations just described. If the ranks of **d** and **d'** are the same, then **r** = **r'**, so approximate solutions Y_K of (3) attempt to replicate the monotonic order structure of D in D'. A detailed description of Kruskal's iterative algorithm for approximating solutions of (5) appears in [5, pp. 69-71].

Our implementation begins by choosing q, randomly initializing $Y_{K0} \in \Re^q$, and computing $D'_0 = \eta'(Y_{S0})$. Let $\mathbf{d}'_{Kt}$ denote the vector of distances, t = 0, 1, … t_{max}. At step t, find all successively indexed subsequences in $\mathbf{d}'_{Kt}$ that are not monotonically increasing, and replace each of these subsets with the average distance in the subsequence. Let the new set of distances be $\hat{\mathbf{d}}'_{Kt}$. Then update Y_{Kt} for i = 1 to n with

$$\mathbf{y}_{Ki,t+1} = \mathbf{y}_{Ki,t} + \gamma \bullet \sum_{i=1}^{n} \sum_{j=1}^{n} \left(1 - (\hat{d}'_{k_{ij}} / d'_{k_{ij}})\right)(\mathbf{y}_{Ki,t} - \mathbf{y}_{Kj,t}). \quad (6)$$

Kruskal's algorithm is terminated with output set $Y=Y_{Kt}$, when $J_K(D'_{Kt} \mid D) < \varepsilon$. All of our examples use the Euclidean norm for η', $\gamma = 0.2$ and $\varepsilon = 10^{-4}$.

E. Non-Metric MDS with MTP-MSO. The input data need only satisfy properties (p1)-(p3) of (1), which includes Euclidean and metric inputs as special cases. Runkler and Bezdek [15] recently proposed a new algorithm for feature extraction called *metric topology preservation - multiswarm optimization* MTP-MSO. This algorithm can also be used for MDS but has not been previously discussed in this context. The long acronym MTP-MSO indicates that this algorithm combines metric topology preservation as defined in [16] with a variation of multi (particle) swarm optimization, introduced in [17]. MSO is itself a generalization of the original form of particle swarm optimization (PSO, [18]). Here is the definition of MTP:

Definition MTP. $\mathcal{P}(\Re^p)$ and $\mathcal{P}(\Re^q)$ are the power sets of $\Re^p$ and $\Re^q$, $\phi : \mathcal{P}(\Re^p) \mapsto \mathcal{P}(\Re^q)$, $\mathbf{y}_i = \phi(\mathbf{x}_i)\ \forall\, i$, $Y = \phi[X]$, $|X| = |Y| = n > 1$. If d is a metric for $\Re^p$ and d' is a metric for $\Re^q$, ϕ is a *metric topology preserving* (MTP) transformation if and only if, for any $\mathbf{x}_i$ in X, whenever $\mathbf{x}_j$ is the k-th nearest (in the sense of d) neighbor of $\mathbf{x}_i$, then $\mathbf{y}_j$ is the k-th nearest (in the sense of d') neighbor of $\mathbf{y}_i$ in Y.

Any feature extractor ϕ that preserves in its range the *relative positions* of (all) neighbors of every point in its domain, has the MTP property. That is, neighbors in $\Re^p$ are still neighbors in $\Re^q$ that have the same relationship to each other in the two sets. A function that has this property lies in between *continuity*, which preserves neighborhoods but not distance order; and *isometry*, which preserves not only distance order, but actual distances.

Using the same relabeling schemes for D and D' as in method D above results in the vectors **d**, **d'** and **r**, **r'** from **d**, **d'**, all in $\Re^T$, $T = n(n-1)/2$, **r** = F(D), **r'**=F(D'), Now compute Spearman's rho [19] between **r** and **r'**,

$$\rho_{Sp}(\mathbf{r}, \mathbf{r}') = \rho_{Sp}(F(D), F(D')) = 1 - \left(6 \sum_{k=1}^{T} (r_k - r'_k)^2 \Big/ (T^3 - T) \right). \quad (7)$$

This is a correlation coefficient, so $-1 \le \rho_{Sp} \le 1$. Bezdek and Pal proved that ϕ is a *metric topology preserving* (MTP) transformation with respect to (D, D') if and only if $\rho_{Sp}(\mathbf{r},\mathbf{r}') = \rho_{Sp}(F(D),F(D')) = 1$. To convert the MTP statistic into an objective function suitable for MDS, let $D = \eta(X)$, $D' = \eta'(Y)$ and write (5) as

$$J_\rho(F(D') \mid F(D)) \quad . \tag{8}$$

The MDS model based on (8) is the optimization problem

$$\underbrace{\max}_{Y \subset \Re^q} \{J_\rho(F(D') \mid F(D))\} . \tag{9}$$

We use (particle) *multi-swarm optimization* (MSO) to look for solutions of (9). MSO is a stochastic optimization algorithm that attempts to maximize the fitness value of an objective function $f : \Re^{qn} \mapsto \Re$ by considering candidates of a population that evolves over many iterations (generations in time, called t below). In our approach each sub-swarm focusses on one fixed part of the solution (a specific feature vector). Our setup is somewhat different from dynamic multi-swarm particle swarm optimization (DMS-PSO) [17], where each sub-swarm focuses on a specific region of the solution space and where sub-swarms are dynamically re-grouped. The (unknown) set of q-vectors to be constructed from D are $Y_\rho = \{\mathbf{y}_1,\ldots,\mathbf{y}_n\} \subset \Re^q$. As shown in Fig. 1, MTP-MSO assigns a swarm of m particle vectors to each of the n feature vectors we seek, so we have n sets of m particle vectors (subswarms), say $Y_{k\rho} = \{\mathbf{y}_{k1},\ldots,\mathbf{y}_{km}\} \subset \Re^q$. Each (unknown) feature vector y_k is associated with $Y_{k\rho}$, which has velocity vectors $\Delta Y_{k\rho} = \{\Delta\mathbf{y}_{k1},\ldots,\Delta\mathbf{y}_{km}\} \subset \Re^q$. The total number of particles (vectors) is mn, as seen in Fig. 1.

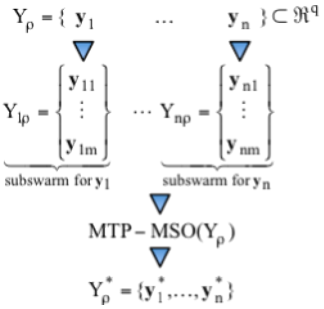

Fig. 1. Architecture of the MTP-MSO Model

Fig. 2 is an abbreviated form of the MTP-MSO algorithm given in [15]. The particle vectors and their velocities for each of the n subproblems are initialized randomly with uniformly distributed random vectors in $[0,1]^q$ for each $\mathbf{y}_k$, and in $[-1,1]^q$ for each $\Delta\mathbf{y}_k$. The set of best particle vectors at the t-th step is $\{\mathbf{y}_{1,t}^*,\ldots,\mathbf{y}_{n,t}^*\} \subset \Re^q$. Instead of accepting $\{\mathbf{y}_{1,t}^*,\ldots,\mathbf{y}_{n,t}^*\}$ as the best we can do, we also compute the local best for each particle vector $\mathbf{y}_{ik,t}$ of the current populations. Then for each k = 1 to n we pick the particle vector that maximizes the MTP objective function for that k (with the best particle vectors for all other k) and, if necessary, rename it $\mathbf{y}_{k,t}^*$. In this way we maintain a set of currently best *overall* candidates. At termination, compute the terminal value $J_\rho(D^* \mid D)$, and we are done.

Algorithm MTP-MSO

Inputs: $D \in M_n^+$; $\alpha \in [0,1]$; $a_1, a_2 \in \Re^+$; m, q, $t_{max} \in \mathcal{N}$. MSO fitness $f(Y) = J_\rho(F(D) \mid F(D'))$.

Initialize $Y = \hat{Y}, \Delta Y$ ***For*** k = 1,…, n : $\mathbf{y}_k^* \in \{\mathbf{y}_{1k},\ldots,\mathbf{y}_{mk}\}$: ***Next*** k

For k = 1,…, n
  $Y_{temp} = Y^*$
  For i = 1,…,m : $Y_{temp,k} = \hat{\mathbf{y}}_{ik}$: $\hat{f}_{ik} = f(Y_{temp})$: ***Next*** i
  $j = \arg\min\{\hat{f}_{1k},\ldots,\hat{f}_{mk}\}$: $\mathbf{y}_k^* = \hat{\mathbf{y}}_{jk}$: $f_k^* = \hat{f}_{jk}$
Next k

For t = 1 to t_{max}
  $\Delta Y = \alpha\Delta Y + a_1 rand(\hat{Y} - Y) + a_2 rand(Y^* - Y)$
  $Y = Y + \Delta Y$
  For k = 1 to n
    $Y_{temp} = Y^*$
    For i=1 to m
      $Y_{temp,k} = \mathbf{y}_{ik}$
      If $f(Y_{temp}) > \hat{f}_{ik}$ ***Then*** $\hat{\mathbf{y}}_{ik} = \mathbf{y}_{ik}$: $\hat{f}_{ik} = f(Y_{temp})$
      If $f(Y_{temp}) > f_k^*$ ***Then*** $\mathbf{y}_k^* = \mathbf{y}_{ik}$: $f_k^* = f(Y_{temp})$
    Next i
  Next k
Next t

Out $\{\mathbf{y}_1^*,\ldots,\mathbf{y}_1^*\} = Y_\rho^* \subset \Re^q$: $J_\rho(F(D) \mid F(D'))$

Fig. 2. The MTP-MSO Algorithm

IV. Numerical Examples

Experiment 1. Anderson's Iris data comprises n =150 feature vectors in p=4 dimensions [20]. Each vector in Iris has one of three (crisp) physical labels corresponding to the subspecie it belongs to; Setosa, Versicolor, or Virginica. The input data for this experiment is Iris 149, which is Iris without feature vector 143, which duplicates vector 102. This accomodates our implementation of Sammon's algorithm that requires $d_{ij} > 0$ for all $i \neq j$ (cf. the denominator of (4)).

We computed dissimilarity data for Iris 149 with the Euclidean ($D_2 \in M_n^e$) and Sup ($D_\infty \in M_n$) norms. D_2 is Euclidean, D_∞ is not: $-PD_\infty P/2$ has 75 positive, 1 zero, and 73 negative eigenvalues. Fig. 3 shows the 2D data sets produced by the four methods for these inputs. The class labels shown are (1=Setosa), (2=Versicolor), (3=Virginica). The scatterplots all show linear separability between class 1 and the mixed classes 2/3. The main points of this example are (i) to verify that MTP-MSO produces 2D realizations of Iris 149 that are in every way comparable to the other three methods; and (ii) that MTP-MSO does not falter on non-Euclidean inputs (the sup norm dissimilarity data D_∞). The Sammon and Kruskal algorithms produce virtually identical

outputs for these three inputs. This happens often, but not always.

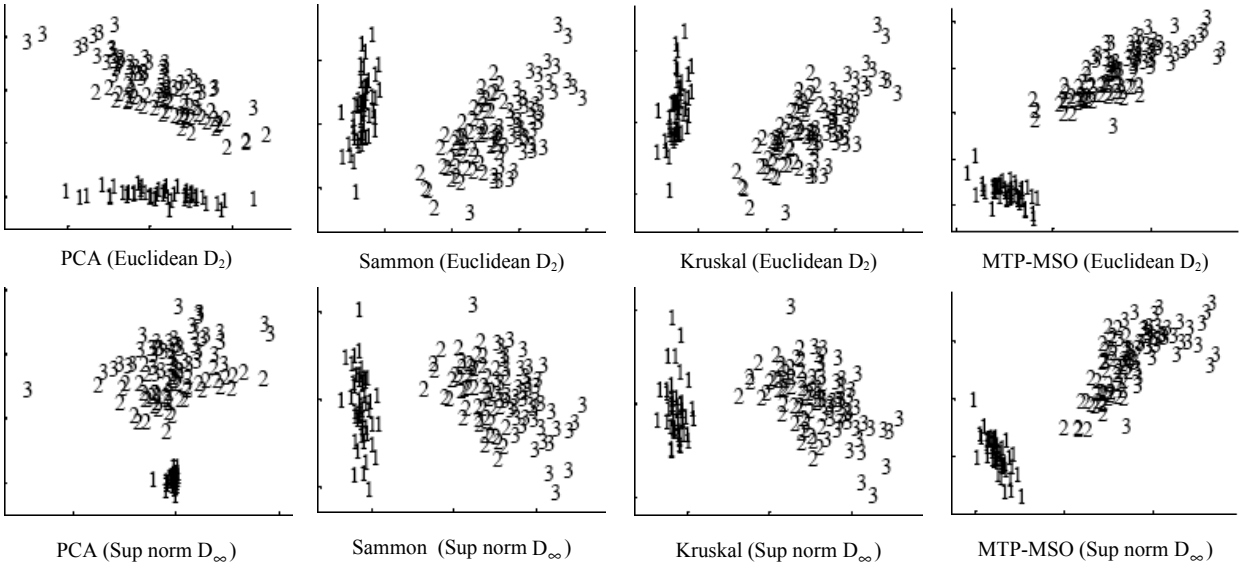

Fig. 3. 2D sets of feature vectors extracted from Euclidean (D_2) and Sup norm (D_∞) distance matrices on Iris 149

Experiment 2. Helm [21] discussed ways to visually represent the perception of colors by normal (not color blind) and abnormal (color blind) subjects. Subjects were asked to place colored tiles in a physical arrangement that showed perceived color similarity as well as relative distance between adjacent tile pairs. Helm found that the average distance matrix $\overline{D}$ for the 10 normal subjects had two large eigenvalues (223, 175) and 8 much smaller ones in the interval [-5.84, 9.43]. This suggested to him that a two-dimensional PCA plot corresponding to the two large eigenvalues would capture the essential structural relationships between the 10 colors. Fig. 4, adapted from Fig. 5 of [16], shows this representation, with color 1="A", color 2 = "C", and on up to to color 10 = "S". The longest distance is from 1 to 2, then 2-3 and 3-4 are roughly equal, and so on, the closest pairs being 8-9 and 9-10 ("S" and "Q"). We processed a slightly different version of Helm's data listed as Table 10 in [8]. The matrix $-P\hat{D}P/2$ of Theorem 2 for this input D has 6 positive, 1 zero, and three negative eigenvalues. Thus, D is not Euclidean, and the minimum embedding dimension is three.

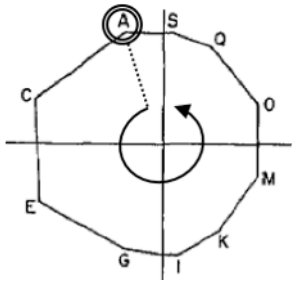

Fig. 4. Helm's plot for 10 combined subjects on first two PCs of $\overline{D}$.

We exhibit scatterplots in Fig. 5 for q = 2 to afford comparisons to Helm's solution shown in Fig. 4. Sammon and Kruskal MDS again yield strikingly similar plots to each other, and this pair of plots are also quite similar to Helm's result. Our PCA and MTP-MSO plots are similar to each other, but less so than Sammon is to Kruskal. Ignoring the "direction of connection," which is not relevant to interpretation of the data, we see that all four algorithms produce similar results: (i) pathwise linear connections in chromatic order; and (ii) relative distances between color pairs that agree with those seen in Fig. 4.

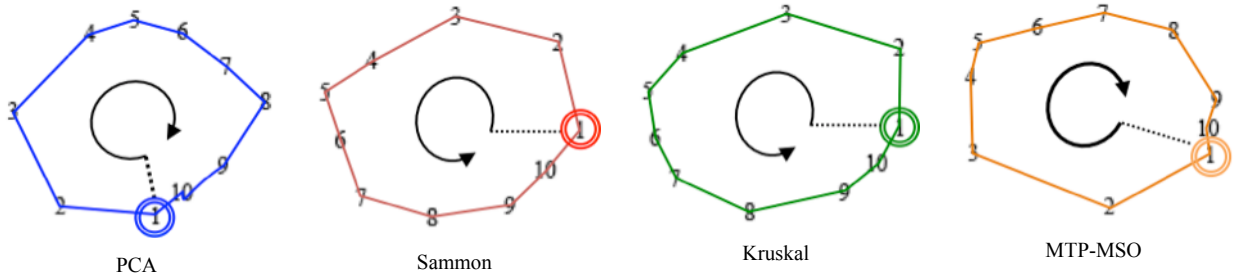

Fig. 5. 2D sets of feature vectors extracted from Helm's data on color perceptions of 10 normal (not color blind) subjects

Experiment 3. Fitch and Margoliash [22] present the phylogentic tree shown in Fig. 6 (Fig. 2, [22]) which is based on average mutation distances between minimum numbers of mutations required to interrelate pairs of cytochromes c (Table 3, [22]).

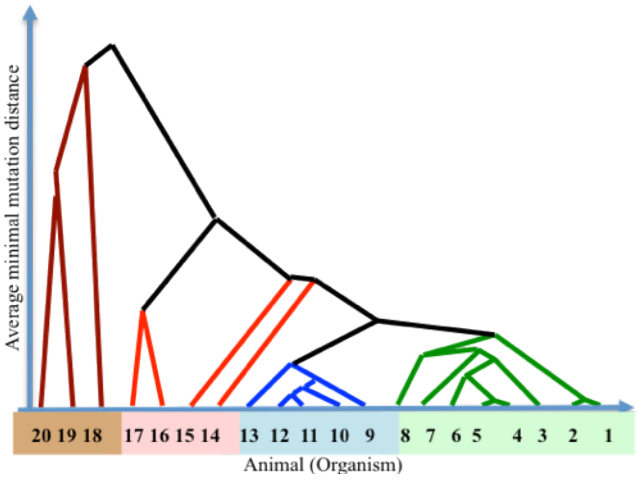

Fig. 6. Mutations of cytochrome c in 20 animal species (Fig. 2, [17])

The 20 species include 1= man, 2=monkey, 3= dog, etc. The tree in Fig. 8 suggests two tight clusters at relatively small average mutation distances, $C_1 = \{1,\ldots,8\}$ and $C_2 = \{9,\ldots,13\}$. The tightest couplings (minimum average joining distances) appear to be the pairs{1, 2} and {4,5} in cluster C_1. Four relatively "anomalous" species seem to form a third cluster, C_3 = {14=snake, 15=tuna, 16=fly, 17=moth}. The last three organisms form a fourth cluster C_4 = {18, 19, 20}, where 18 = baker's mould, 19 = bread yeast, and 20= skin fungus. These non-sentient organisms join the first 17 species at almost twice the mutation distances that bind the others together. $-\hat{P}DP/2$ for these data has 6 negative eigenvalues, so the input data are not Euclidean.

Fig. 7 displays scatterplots of the 2D sets of feature vectors extracted from the data with our four study models. There are four rows in Fig. 7. The top row exhibits all 20 vectors derived from each model. Note that 19 and 20 are superposed in the Kruskal diagram in this row. All four models place the outlier cluster $C_4 = \{18, 19, 20\}$ apart from the other 17 organisms. To see more clearly how the MDS vectors separate the remaining input species, we made successively finer "zooms" that increased the resolution of the remaining clusters to make further distinctions possible. The blue boxes in the top row are zoomed to produce the scatterplots in the second row. Here we see the anomalous points cluster $C_3 = \{14, 15, 16, 17\}$ scattered about, outside of the red boxes. Zooming the red boxes yields the third row of Fig. 9, where the cluster $C_2 = \{9,\ldots,13\}$ lies outside the green bounding boxes in all four views. Finally, expansion of the green boxes yields the graphs in the bottom row of Fig. 7, where all four models have grouped together $C_1 = \{1,\ldots,8\}$.

So, all four models produce 2D models that offer visual evidence that agrees with the stucture of the data captured by Fitch and Margoiash in Fig. 6. Please notice the highlighted pairs {1, 2} and {4,5} in the bottom right view of Fig. 7. These are the most tightly coupled pairs in the tree of Fig. 6, and MTP-MSO produces vectors for these pairs that are closer to each other than they are in the other three views in tha last row. Thus, in this example anyway, we feel that our new approach to MDS performs a bit better than the comparison algorithms.

V. Conclusions and Discussion

We reviewed three well known approaches to MDS (PCA, Sammon, Kruskal), and compared them to outputs from our new MTP-MSO algorithm on three real data sets. The results suggest that our model is generally comparable to Sammon's model, and both of these seem slightly better than PCA and Kruskal's approaches. Moreover, our new algorithm yields vectorial representatives for the mutation data that are closer to the original interpretation of this data set than the other three models. These examples suggest that there is merit in further investigations of MTP-MSO.

One aspect of our method that might lead to better MDS solutions is that optimization of the MTP objective function is done with multi (particle) swarm optimization. This type of evolutionary computation often avoids local extrema that stall the Kruskal and Sammon algorithms at undesirable minima. On the other hand, traditional MDS is significantly faster than our implementation of MTP-MSO. As often happens, a new algorithm such as MTP-MSO raises more questions than it answers. What is the exact time and memory complexity of MTP-MSO and how can it be decreased? Can we adapt it for MDS in big data? What can we do for visualization when the data is sparse or incomplete? Is there a better way to optimize the MTP stress function? Our immediate aim will be to adapt MTP-MSO to address some of these questions.

References

[1] J. Dattorro. "Convex Optimization and Euclidean Distance Geometry," Meboo Publishing USA, Palo Alto, 2005.

[2] N. Krislock and H. Wolkowicz. "Euclidean distance matrices and applications," *Handbook on Semidefinite, Conic and Polynomial Optimization*, eds. M. F. Anjos and J. B. Lasserre, Springer Int. Series on OR and MS, v166, 879-914, 2012.

[3] R. J. Hathaway and J. C. Bezdek, J. C. "NERF c-Means : Non-Euclidean relational fuzzy clustering," *Pattern Recogition*, 27(3), 429-437, 1994.

[4] K. V. Mardia, J. T. Kent and J. M. Bibby, "Multivariate Analysis," Academic Press, New York, 1979.

[5] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling*, Chapman and Hall, 2001.

[6] J. C. Gower and P. Legendre. "Metric and Euclidean Properties of Dissimilarity Coefficients," *J. Classification*, 3, 5-48, 1986.

[7] I. J. Schoenberg, "Remarks to Maurice Frechet's article "Sur la definition axiomatic d'une classe d'espace distancies vector-iellement applicable sur l'espace de Hilbert," Ann. Math, 36(3), pp. 724-732, 1935.

[8] J. Benasseni, M. B. Dosse and S. Joly. "On a General Transformation Making a Dissimilarity Matrix Euclidean," *J. Classification*, 24, 2007, pp.33-51.

[9] G. Young and A. S. Householder. Discussion of a set of points in termsof their mutual distances. Psychometrika, 3:19–22, 1938.

[10] W. F. Torgerson. "Multi-dimensional Scaling: I Theory and Method," *Psychometrika*, 17(4), 401-419, 1952.

[11] M. L. Davison, *Multidimensional Scaling*, J. Wiley and Sons, NY, 1983.

[12] Borg, I., Groenen, P. (2005). *Modern Multidimensional Scaling: theory and applications* (2nd ed.). New York: Springer-Verlag.

[13] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Trans. Comput.,* 18, 1069, pp. 401-409.

[14] J. B. Kruskal. "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, 29(1), 1964, pp.1-27.

[15] T. Runkler and J. C. Bezdek, "Topology Preserving Feature Extraction with Multiswarm Optimization," *Proc. 2013 IEEE Int. Conf. on Systems, Man and Cybernetics*, in press.

[16] J. C. Bezdek and N. R. Pal, "An index of topological preservation for feature extraction," *Pattern Recogition*, 28(3), 1995, pp. 381-391.

[17] S. Z. Zhao, J. J. Liang, P. N. Suganthan, and M. F. Tasgetiren. Dynamic multi-swarm particle swarm optimizer with local search for large scale global optimization. In *Proc. IEEE Congress on Evolutionary Computation*, 2008, pp.3845–3852.

[18] Kennedy and R. C. Eberhart, "Particle swarm optimization," In IEEE International Conference on Neural Networks, pp. 1942–1948, Perth, Australia, 1995.

[19] M. Kendall and J. D. Gibbons, Rank Correlation Methods, Oxford University Press, NY, 1990.

[20] E. Anderson, "The Irises of the Gaspe peninsula," *Bull. Amer. Iris Soc.*, 59, pp. 2-5, 1935.

[21] C. E. Helm, "Multidimensional ratio scaling analysis of perceived color relations," *J. Opt. Soc. America*, 54(2), pp. 256-262, 1964.

[22] W. M. Fitch and E. Margoliash, "Construction of phylogentic trees," *Science*, 155, n. 3760, pp. 279-284, 1967.

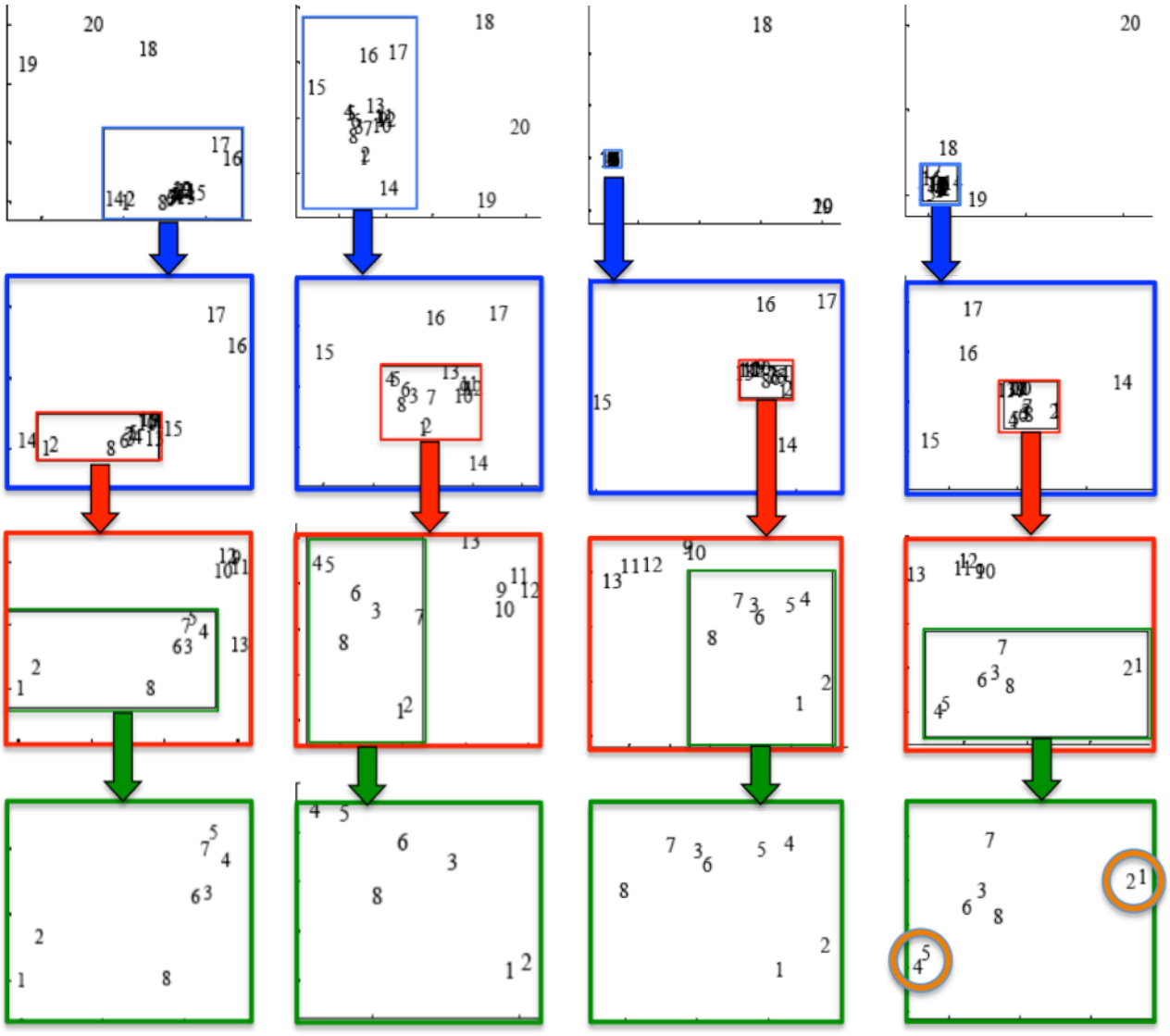

Fig. 7. Feature vectors extracted from Fitch's mutation data: top row = full scale: row 2 = zoom blue: row 3 = zoom red: bottom row = zoom g