國立交通大學

統計學研究所

碩士論文

分群與合併的多元尺度分析法之最佳分群決策與遺失值問題的討論

Optimal Grouping and Missing Data Handling for Split-and-Combine Multidimensional Scaling

研 究 生：陳珮琦

指導教授：盧鴻興　教授

中華民國九十七年六月

Optimal Grouping and Missing Data Handling for Split-and-Combine Multidimensional Scaling

研究生：陳珮琦	Student：Pei-Chi Chen
指導教授：盧鴻興	Advisor：Henry Horng-Shing Lu

國立交通大學

統計學研究所

碩士論文

A Thesis

Submitted to Institute of Statistics

College of Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Statistics

June 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年六月

分群與合併的多元尺度分析法之最佳分群決策

與遺失值問題的討論

學生：陳珮琦　　　　　　　　　　　　　　　　指導教授：盧鴻興　博士

國立交通大學統計學研究所碩士班

摘　　要

MDS (Multidimensional scaling)為資料探礦中的重要方法之一。MDS 的主要目的為二:(一)使資料能夠在空間中，以點座標形式來表示而不失其差異性。(二)降低資料維度，讓資料得以視覺化的形式呈現，更容易找出資料潛在的特徵。傳統的 CMDS (Classical multidimensional scaling)計算量非常大，對於樣本過大的資料，計算不易。因此曾正男博士在 2008 年提出了一個新的方法 SC-MDS (Split-and-combine multidimensional scaling)來解決 CMDS 計算量龐大的問題。而在計算 SC-MDS 過程中，有兩個重要的參數必須要決定：分群時應與鄰近的集合交疊多少個點（N_I），以及每個集合應該要多大（N_g）。因此，該如何選擇兩參數才能將 SC-MDS 方法的表現最佳化則是本文討論的主要重點之一。這裡我們建議N_I至少要是資料維度加一，而N_g大約是 1.51 倍的N_I能讓 SC-MDS 有最佳的表現。另外，文中也討論各種 SC-MDS 在分群時的可能情況，並修正原本集合合併的方法，不應以一特定集合為中心，將其他集合往同一集合合併；而應該要考慮兩集合各自的維度，將低維度的集合往高維度的集合合併。因此，群集中只要任一群集的維度，與全部資料維度相同，則 CMDS 與 SC-MDS 將會在同一個空間下被展開，這部份會在文中有詳細的證明與討論。本文中另一個討論的主題，便是運用 SC-MDS 的基本想法處理遺失值的問題。在計算的過程中，我們不去補遺失值，而是將所有的點重新排序，使得每一個子群內，在計算 MDS 的時候沒有遺失值。在此種方法下，我們可以將遺失值的容許比例由 20%提升至 30%。

Optimal Grouping and Missing Data Handling for Split-and-Combine Multidimensional Scaling

Students: Pei-Chi Chen   Advisor: Henry Horng-Shing Lu

Institute of Statistics
National Chiao Tung University

ABSTRACT

Multidimensional Scaling (MDS) is one of many important methods in data mining. It has two main purposes: (1) Express data in coordinate points in spatial configuration from given pair-wise distances between data. (2) Reduce data dimensions and find hidden features of data through visual display. We focus on discussing Classical multidimensional scaling (CMDS) in this paper since there are many types of MDS methods. CMDS faces some challenges. One of them is that CMDS's calculation time is huge. So it's hard to calculate data with a large sample size. Therefore, Tzeng, Lu and Li (2008) proposed split-and-combine multi-dimensional scaling (SC-MDS) to figure out this problem. However, in the process of SC-MDS, there are two important parameters to be decided: (1) the number of overlapping points with the neighboring groups, N_I, and (2) the size of each group, N_g. These two parameters have great effects on the performance of SC-MDS. Thus, the main topic of this paper discusses how to best choose these two parameters. We suggest that N_I should be at least the dimensionality of the data plus one and N_g be about 1.51 times N_I to make SC-MDS perform optimally. In addition, we revise the method for combining groups. When combining two groups, we should consider their own dimension and then align the group with lower dimensionality to the group with higher dimensionality; instead, we randomly choose one group as the center then align the other group to it. Therefore, CMDS and SC-MDS will be spanned by the same space as long as any one group has the same dimension as the whole data's. There is a proof and discussion in this article. Another main topic in this article is using the SC-MDS concept to solve the missing value problem. We did not refill the missing value; instead, we permute over the index of objects, in which the subgroups in SC-MDS processing have no missing value. Then, it can raise the tolerance of ratio of missing values from 20% to 30% by simulation result.

誌　　　謝

首先我要誠摯的感謝我的指導教授盧鴻興老師，這兩年給予我耐心與細心的指導。讓我有機會接觸到許多不一樣的領域與方法，教導我正確的學習精神與嚴謹的研究態度。另外特別感謝曾正男學長，在研究時，能夠悉心的引導我，提供我方向，即使遇到瓶頸逃避時，也完全不責備而能以夠體諒的心，幫助我。感謝口試委員，徐南蓉、王秀瑛、黃冠華老師對這篇論文的指導與建議，讓這篇研究更完整。

謝謝在研究過程中幫助過我的同學們，夙吟、香菱、郤嵐、佩芳、姿蒨、宜德，在我遇到問題的時候，陪我一起討論，一起翻書找答案。還有在盧老師實驗室一起努力的同學們，政言、瑜達、健鈞。也謝謝交大統研所的所有同學，在兩年的碩士生活中，給我很多難忘的回憶。還有急救社的朋友，不時的關心，在精神上給我很大的支持。

此外，感謝我的家人不斷的鼓勵我，給予我充足的資源，讓我在學習之餘沒有後顧之憂。謝謝紗布幫我畫圖，滴滴隨時提供給我精神糧食，聽我抱怨。

最後將此篇文章獻給所有曾經幫助我的人。

陳珮琦　謹誌于

國立交通大學統計學研究所

中華民國九十七年六月

Contents

1 Introduction

1.1 Origination

Multidimensional scaling (MDS) is a method developed in behavioral and social sciences. It was proposed by Torgerson (1952) based on Young and Householder (1938). Psychologists usually want to know the relationship between stimuli or objects, which are characterized by some subjects. For instance, stimuli may be countries, and each country may record its population, location, weather and so on to describe it. Dissimilarity (or similarity) will be defined through difference of these subjects. For exploratory purposes, psychologists observe the dissimilarity (or similarity) of these stimuli. They want to transform the dissimilarity into a point configuration such that the distance between pairs of points is consistent with dissimilarity. Consequently, they can find a rule to describe the point spread. Let $\mathbf{X}_{p\times n}$ be a data set; each column vector x_i can be represented as $\mathbf{x}_i$=$(\mathbf{x}_{1i}, \mathbf{x}_{2i}, \ldots, \mathbf{x}_{pi})^T \in \mathbb{R}^p$, denoting the i-th stimulus characterized by p subjects. $D = \{d_{ij}\}_{n\times n}$ is the distance of i-th object and j-th object in $\mathbf{X}$. In Young and Householder (1938) mentioned that ”a necessary and sufficient condition for a set of numbers $d_{ij} = d_{ji}$ to be the mutual distances of a real set of points in Euclidean space is that the matrix $\mathbf{B}$=$\mathbf{X^T X}$ be semi-positive definition; and in this case the set of points is unique apart from a Euclidean transformation.” After that, Torgerson constructed the algorithm of MDS based on Young and Householder (1938).

To introduce MDS, we must start with proximities. Proximities record the similarity or dissimilarity between objects. They can be in a distance matrix, correlation matrix, or so on. MDS is performed on proximities. MDS is aimed to transform proximities into a spatial configuration and find the underlying dimension to describe the similarity or dissimilarity. MDS can be applied to project a high dimension matrix into a low dimension matrix, which is not the original row data; instead, it could be a reduced data (dimensional reduction), but we can’t really specify the meaning of these orientations without changing the relative dissimilarity of object pairs. Every object would be expressed as a point so that we can easily give a graphical description of objects. There are many kinds of models in MDS; which model is more appropriate depends on your input data.

1.2 Introduction to classical MDS

Assume there are N observations in an investigation; each observation would be described using p variables. Thus we can define a p by n data matrix $\mathbf{X}$, with each column of $\mathbf{X}$ denoted as $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{pi})$, in a p-dimensional Euclidean space without loss generation. To ensure a unique solution for an arbitrary translation when performing MDS on matrix $\mathbf{X}$, we shift the row mean to zero, denoted as $\sum_{i=1}^{n}\mathbf{x}_i = 0$, because we care about relative location instead of absolute position of points. First, we have to define the distance matrix or dissimilar matrix of $\mathbf{X}$. Here, we use the Euclidean distance to define the dissimilar matrix, so the square distance matrix $\mathbf{D}$ can be expressed as $\mathbf{D} = [d_{ij}]_{n\times n} = (\mathbf{x_i} - \mathbf{x_j})^T(\mathbf{x_i} - \mathbf{x_j})$. Besides, we define the inner product matrix $\mathbf{B} = \mathbf{X^TX}$.

Deriving:

$$\frac{1}{n}\sum_{i=1}^{n} d_{ij} = \frac{1}{n}(\sum_{i=1}^{n}\mathbf{x}_i^T\mathbf{x}_i - \mathbf{x}_j^T\sum_{i=1}^{n}\mathbf{x}_i - \mathbf{x}_j^T\sum_{i=1}^{n}\mathbf{x}_i + n\mathbf{x}_j^T\mathbf{x}_j) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^T\mathbf{x}_i + \mathbf{x}_j^T\mathbf{x}_j$$

$$\frac{1}{n}\sum_{j=1}^{n} d_{ij} = \frac{1}{n}(\sum_{j=1}^{n}\mathbf{x}_j^T\mathbf{x}_j + \mathbf{x}_i^T\sum_{j=1}^{n}\mathbf{x}_j + \mathbf{x}_i^T\sum_{j=1}^{n}\mathbf{x}_j + n\mathbf{x}_i^T\mathbf{x}_i) = \frac{1}{n}\sum_{j=1}^{n}\mathbf{x}_j^T\mathbf{x}_j + \mathbf{x}_i^T\mathbf{x}_i$$

$$\begin{aligned}\frac{1}{n^2}\sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij} &= \frac{1}{n^2}\sum_{j=1}^{n}(\sum_{i=1}^{n}\mathbf{x}_i^T\mathbf{x}_i + n\mathbf{x}_j^T\mathbf{x}_j) = \frac{1}{n^2}(n\sum_{i=1}^{n}\mathbf{x}_i^T\mathbf{x}_i + n\sum_{j=1}^{n}\mathbf{x}_j^T\mathbf{x}_j) \\ &= \frac{1}{n}(\sum_{i=1}^{n}\mathbf{x}_i^T\mathbf{x}_i + \sum_{j=1}^{n}\mathbf{x}_j^T\mathbf{x}_j)\end{aligned}$$

implies

$$b_{ij} = x_i^T x_j = -\frac{1}{2}(d_{ij} - \frac{1}{n}\sum_{i=1}^{n} d_{ij} - \frac{1}{n}\sum_{j=1}^{n} d_{ij} + \frac{1}{n^2}\sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij}) = -\frac{1}{2}(d_{ij} - d_{i\cdot} - d_{\cdot j} + d_{\cdot\cdot})$$

Then, matrix $\mathbf{B}$ can be expressed as

$$\mathbf{B} = -\frac{1}{2}(\mathbf{D} - \mathbf{\bar{D}_r} - \mathbf{\bar{D}_c} + \mathbf{\bar{D}_g}) = \mathbf{HPH}$$

where $\bar{\mathbf{D}}_r = [d_{i\cdot}]$ be column means of $\mathbf{D}$, $\bar{\mathbf{D}}_c = [d_{\cdot j}]$ be row means of $\mathbf{D}$, $\bar{\mathbf{D}}_g = [d_{\cdot\cdot}]$ be grand mean of $\mathbf{D}$, $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ and $\mathbf{P} = [p_{ij}]_{n\times n} = -\frac{1}{2}\mathbf{D}$. $\mathbf{B}$ could be obtained from double centering of distance matrix $\mathbf{D}$ multiplying by $-\frac{1}{2}$.

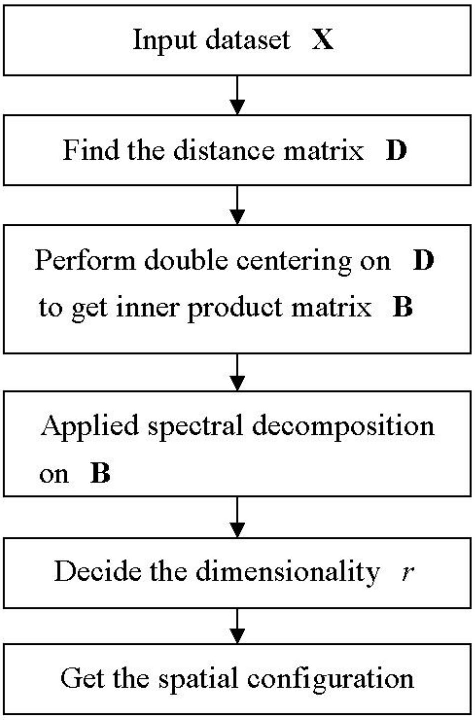

Input dataset **X**
Find the distance matrix **D**
Perform double centering on **D** to get inner product matrix **B**
Applied spectral decomposition on **B**
Decide the dimensionality *r*
Get the spatial configuration

Fig. 1: Flowchart for CMDS

Second, apply SVD to $\mathbf{B}$, then we can get $\mathbf{B} = \mathbf{U\Lambda U^T}$ (because $\mathbf{B}$ is a symmetric matrix, SVD is identical with spectral decomposition), where $\mathbf{\Lambda}$ is a diagonal matrix with eigenvalues on its diagonal terms, and $\mathbf{U}$ is an orthogonal matrix, $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, hence $\mathbf{X} = \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{U}^T$. Using the same concept with PCA, we truncate N-r components with (relatively) small eigenvalues or with eigenvalues less than one and keep r main components such that the dissimilarity among objects can be maintained as far as possible.

$$\tilde{\mathbf{X}} = \mathbf{\Lambda}_r^{\frac{1}{2}}\mathbf{U}_r^T \ \textit{where}\ r < p$$

Although truncating the components with small eigenvalues (< 1) is not the only criterion, another principal will be introduced in the following.

1.3 Introduction to nonmetric MDS

Nonmetric MDS was proposed by Shepard (1962) and Kruskal (1964). Proximities in nonmetric MDS don't offer distance values; instead, they offer only ordinal information. For example, when we measure the perceptual space of human subjects, it is hard to tell how much difference there is between two objects, but it is easy to say that comparing with object B and C, object A is more close to object B. Hence, dissimilar rank can be defined.

1.3.1 How to sort the dissimilarity for the proximities?

There are many ways to rank the similarities. One method is to prepare cards for each pair of objects and let subjects arrange every card according to their similarity. Another method is to divide the cards into two groups, one with higher similarity and another with lower similarity, and repeat the procedure on each pile until the similarity of object pairs in the same pile are approximately equal. Furthermore, still another method is to write one object on a card and put the similar objects in the same pile, and then count the number of times that two objects occur in the same pile as the proximities. This kind of definition of proximities is very intuitive.

1.3.2 How does nonmetric MDS operate?

In nonmetric MDS, the distance between two objects is meaningless to us. To be more precise, the value of proximities only matters in their relative sizes; the distance between two values does not have any meaning. Hence, transforming proximities into spatial points needs to preserve the rank order of pairs of points. So, we want to find a monotonic function such that proximities transformed by this function could still preserve their dissimilar order. Distances of points which transform from proximities through the optimal monotonic function, denoted as $\hat{\mathbf{d}} = f(\mathbf{p})$, are called disparities. The problem of nonmetric MDS transfers to how to arrange the points configuration and how to find an optimum transformation so that the order of disparities and proximities will be consistent as well as possible. Here, Kruskal (1964) proposed an iterative technique to find the transformation and suggested minimizing STRESS to access how well the configuration fits.

$$STRESS = \sqrt{\frac{\sum_{i,j} (p_{ij} - \hat{d_{ij}})^2}{\sum_{i,j} p_{ij}^2}}$$

1. Given an initial configuration

2. Find the distance d_{ij} of the configuration

3. Find the optimum transformation and calculate $\hat{\mathbf{d}}$

4. Use steepest descent to find a new configuration

5. Compare the stress to one iteration forward. If it is smaller than some criterion, end this algorithm or go back to step 2

More details are in *Multidimensional Scaling*, Cox and Cox, chapter3.

Although Multidimensional scaling has many types of models for each kind of data, here we focus on classical MDS. Let's go back to the classical MDS and confront some challenges.

1. How many components should we keep?

   In other words, how many dimensions does this data set need in order to at least keep it dissimilarity? There are lots of suggestions, such as picking up those components with eigenvalue > 1, using the minimal number of dimensions such that stress is less than 0.05, or deleting the components with a small eigenvalue relative to others. Some of these suggestions are from researchers' experience according to huge numbers of trials. No one can ensure that estimating the number of dimensions is the best estimator according to these suggestions. Another well-known method is the scree test or elbow test: plot the scatter plot of dimension vs. stress, observe variety of stress as dimension changing, and choose the point which doesn't have a significant decrease as the dimension increases. This method has a disadvantage. If the curve shows a mild decrease when the dimension gets large, it is hard to choose which point is the elbow point, and the method is then inactive. You can see more principals by refering to "*Stopping Rules in Principal Components Analysis: A Comparison of Heuristical and Statistical Approaches*" , Donald A. Jackson (1993).

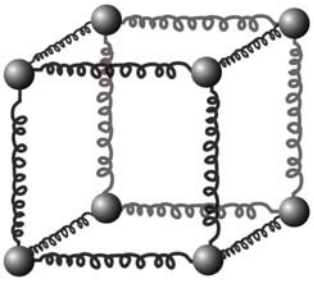
2. Classical MDS algorithm is slow.

   The computation complexity is $O(N^3)$. It would cost a lot of time to calculate when the sample size is large. Many kinds of MDS methods are proposed for large data sizes, such as Chalmer's linear iteration algorithm, anchor point method, relative MDS, landmark MDS, or Diagonal majorization algorithm (DMA).

3. Missing data of distance matrices are not allowed. CMDS is a PCA based method. It does not allow any missing value in the matrix when we reconstruct the data coordinate. There are many methods to refill the missing data, such as shortest path. However, the computation cost is huge.

2 Literature Review

2.1 Chalmers' Linear Iteration Algorithm



Many kinds of MDS models are developed. The spring model is one kind of MDS model. It is a force-directed model. Proximities are considered as a physical system. Imagine that objects are connected with each other using springs. The proximity of two objects is considered the length of a relaxed spring. At first, we will initialize positions of objects arbitrarily so these springs will be stretched or compressed. If we have these springs oscillate liberally, the system will eventually get into equilibrium with a minimal energy. Stress is a suitable measurement here to measure the energy of the system. In other words, the spring model for MDS aims to minimize the stress. This process will be achieved by an iterative algorithm. However, the computation complexity is $O(N^3)$.

The spring model is based on a method proposed by Chalmers (2006). Its time cost is better than the spring model's time cost. The computation complexity is reduced to $O(N^2)$. The Chalmers linear iteration model reduces the number of calculations for forces. Two sets are defined. For an object i, the first set, V, collects the neighbor objects of object i. We randomly select some objects from those objects out of a neighbor set and check the proximity of object i and itself. If the proximity is less than any one of its current neighbors, then swap it into set V, else collect it in a second set S. The set S will be reconstructed in each iteration. Thus, in the iteration of each object, not all the force information is used. This will reduce the order of computation complexity.

In this case, the spring model is good for adding new points into system. Still, it is not stable for the general solution; therefore, good initial values are needed.

2.2 Anchor Point Methed

The anchor point method is proposed by Andreas Buja, Deborah F. Swayne, Michael L. Littman, and Nathaniel Dean (1998). In the anchor point method, the distance between objects in the same cluster is important, and the distance between objects in different clusters is less important. So, we define objects in the same cluster as anchors, and the others as the floaters. Information about similarity for anchors is used to construct the framework of the structure, and information about similarity for floaters is used to adjust the fine structure. The number of anchors, k, should be larger than the dimensionality of dataset, and it only needs to apply modified MDS on the $N \times k$ matrix. This method will not work if we select an anchor randomly, which was mentioned in Buja et al (1998). Prior information about grouping for the anchor point method is necessary.

3 Split and Combine MDS

3.1 Introduction to Split and combine MDS

Split-and-Combine Multidimensional Scaling (SC-MDS), which was proposed by Tzeng, Lu, and Li (2008), is a modified MDS method to solve the computation problem of MDS for large data sizes. In these cases, it is reasonable to assume that the data size is huge and that each data is defined by $\mathbb{R}^p$, $p \ll N$. The main idea of SC-MDS is that the orientation in spatial configuration does not need such a great deal of information from all the points. For instance, in two-dimensional space, if we want to add a new object onto a plane by only distance information, we just need to know the distance between new point and at least three arbitrary non-colinear points on the plane, then we can set the position precisely. If we want to set a point in a q-dimensional space, we only need the distance from arbitrary non-colinear $q+1$ points. Instead, too much information will cause a huge computing complexity in dealing with the amount of information. To apply this main ideal in both split step and combine step, we divide data into K groups. Each group has at least $r+1$ points, where r is the dimensionality of data. Then, find the spatial configuration to each part. At the moment, it faces two problems: (1) How to combine each group and make each set of points defined in the same axis space? (2) What sufficient conditions are needed such that this method can perform reasonably?

First, let's answer the first question. In this method, we would ideally like to combine each group by overlapping points. When we divide the data, we arrange each group to have N_i overlapping points with its neighbor group, so that we can make use of information from the overlapping part to combine each groups. Notice that the Combine step uses the same idea as the Split step: N_i must larger than r. Figure 2 shows the ideal way to combine two groups by overlapping and what will occur if the number of overlapping points, N_i, is less than $r+1$. In the following, we will simplify the problem to $K=2$, and discuss the detailed process of SC-MDS.

SC-MDS has two steps. The first step is *Split* and the second step is *Combine*. Here, we extend the notation from the above. Let $\mathbf{X_{p\times n}}$ be a data set which can be expressed as $\mathbf{X} = [\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_n}]$ and each column of $\mathbf{X}$, denoted as $\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{ip})$ is defined in

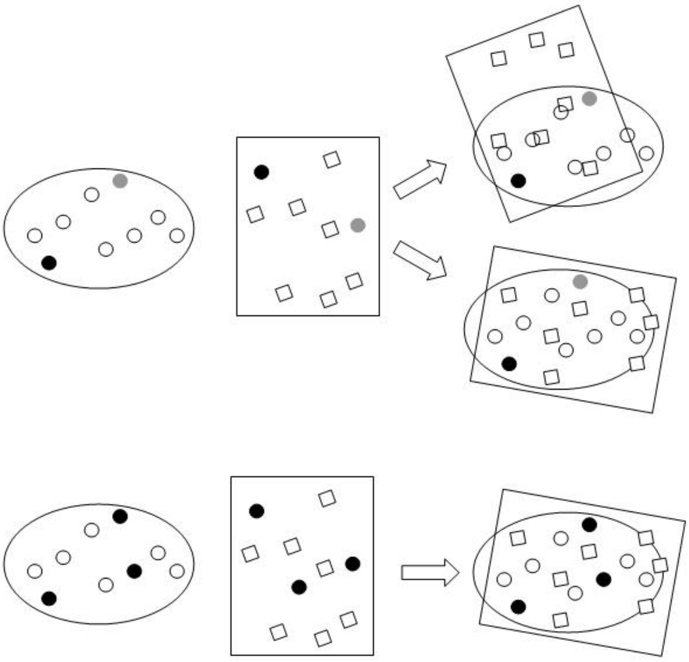


Fig. 2: Groups combine in two-dimension space. Elliptic frame and quadrate frame represent two point groups. The solid points represent overlapping part.

a p-dimensional Euclidean space.

Split Assuming a large sample size, N is a large integer, and $p \ll N$. Under this condition, applying CMDS will cost extensive time. We try to split the data set into two overlapping groups $\mathbf{X_1}$ and $\mathbf{X_2}$, $\mathbf{X_1} \cap \mathbf{X_2} = \mathbf{Y} \neq \emptyset$. We will discuss the reasons later. After applying MDS on $\mathbf{X_1}$ and $\mathbf{X_2}$ individually, we can obtain two point sets defining in the same dimension space, $\mathbf{X}'_1$ and $\mathbf{X}'_2$. $\mathbf{Y}_1$ and $\mathbf{Y}_2$ represent the spatial structure of the intersection points after applying the MDS in two groups seperately.

Combine Subsequently, we want to utilize the overlapping part to connect two groups. According to Young and Householder (1938), these two spatial configurations must be consistent through Euclidean transformation. Thus, we want to find an affine mapping $\mathbf{U}(\cdot)+\mathbf{b}$ such that each overlapping point $x'_{1j} \in \mathbf{Y}_1$ can find the corresponding point in $\mathbf{Y}_2$ satisfying $\mathbf{x}'_{1j} = \mathbf{U}x'_{2j} + \mathbf{b}$, $x'_{2j} \in \mathbf{Y}_2$. Assume $\mathbf{\bar{Y}_1}$ and $\mathbf{\bar{Y}_2}$ represent the average of $\mathbf{Y}_1$ and $\mathbf{Y}_2$. Shift the row mean of each point sets to zero, $\mathbf{Y}_1 - \mathbf{\bar{Y}_1}\mathbf{1}^T$ and $\mathbf{Y}_2 - \mathbf{\bar{Y}_2}\mathbf{1}^T$ and apply QR factorization on $\mathbf{Y}_1 - \bar{\mathbf{Y}}_{\mathbf{1}}\mathbf{1}^T$ and $\mathbf{Y}_2 - \bar{\mathbf{Y}}_{\mathbf{2}}\mathbf{1}^T$, then it can obtain $(\mathbf{Y}_1 - \bar{\mathbf{Y}}_{\mathbf{1}}\mathbf{1}^T) = \mathbf{Q_1R_1}$ and $(\mathbf{Y}_2 - \bar{\mathbf{Y}}_{\mathbf{2}}\mathbf{1}^T) = \mathbf{Q_2R_2}$. $\mathbf{R}_1$ and $\mathbf{R}_2$ should be equal although the same point set is expressed by different orientations. Thus, we have to contrast $\mathbf{R}_1$ with $\mathbf{R}_2$ to adjust positive and negative signs in columns of $\mathbf{Q}_1$ and $\mathbf{Q}_2$. These two equal equations can combine into $\mathbf{Q}_1^T(\mathbf{Y}_1 - \bar{\mathbf{Y}}_{\mathbf{1}}\mathbf{1}^T) = \mathbf{Q}_2^T(\mathbf{Y}_2 - \bar{\mathbf{Y}}_{\mathbf{2}}\mathbf{1}^T)$.

Extending the equation above, we will have

$$\mathbf{Y}_1 = \mathbf{Q_1Q_2^TY_2} - \mathbf{Q_1Q_2^T\bar{Y}_2 1^T} + \mathbf{\bar{Y}_1 1^T} \tag{1}$$

$$\mathbf{U} = \mathbf{Q_1Q_2^T} \quad \text{and} \quad \mathbf{b} = -\mathbf{Q_1Q_2^T\bar{Y}_2 1^T} + \mathbf{\bar{Y}_1 1^T}$$

$$\tilde{\mathbf{X}}_2 = \mathbf{UX_2'} + \mathbf{b}$$

Then, $\tilde{\mathbf{X}}_{r\times n} = [\mathbf{X}_1'\tilde{\mathbf{X}}_{\mathbf{2}}]$. However, a rounding error usually exists in practical calculation processes. $\mathbf{R}_1$ and $\mathbf{R}_2$ will not be identical altogether. Besides, even if we randomly split the matrix into groups, two groups with different dimensionalities are still possible to occur. We will discuss some probable situations in the following.

3.2 More discussion about combine step

- $dim(\mathbf{X}_1) = dim(\mathbf{X}_2 = dim(\mathbf{X}_1 \cap \mathbf{X}_2)) = r$

  This is the basic situation; the above derivation is under this assumption.

- $dim(\mathbf{X}_1) = r_1 > dim(\mathbf{X}_2) = dim(\mathbf{X}_1 \cap X_2) = r'$

  Assume the dimensionality of two groups is different even by random grouping, such that the same points (overlapping part) are expressed as different dimensions. In this case, $\mathbf{Q}_1$ and $\mathbf{Q}_2$ will become $r \times r'$ and $r' \times r'$ matrix after QR decomposition. It means that only an r' basis can span these points (overlapping part) in r-dimension space, and the corresponding $\mathbf{R}_1$ and $\mathbf{R}_2$ will drop the dimension to r' being $r' \times N_I$ matrices. Then, we can find the affine transformation based on the same dimension such that equation (1) will hold. After $\mathbf{U} = (\mathbf{Q}_1)_{r\times r'}(\mathbf{Q}_2^T)_{r'\times r'}$ operate on $(\mathbf{X}_2')_{r'\times N_I}$, $\tilde{\mathbf{X}}_2 = \mathbf{U}_{r\times r'}(\mathbf{X}_2')_{r'\times n_2} + \mathbf{b}_{r\times n_2}$ to move up to r-dimension space. Inversely, if $dim(\mathbf{X}_1) = r' < dim(\mathbf{X}_2) = r$, in equation (1) $\mathbf{U} = (\mathbf{Q}_1)_{r'\times r'}(\mathbf{Q}_2^T)_{r'\times r}$ operating on $(\mathbf{X}_2)_{r\times N_I}$ will project the higher dimension part into lower dimension. As a result, dimensionality of $\mathbf{X}_2$ will descend and lose partial information of $\mathbf{X}_2$. Consequently, if the dimensionality of two groups is different, it is more reasonable to transform the lower dimension part into higher dimension space by affine mapping.

- $dim(\mathbf{X}_1) = dim(\mathbf{X}_2) = r > dim(\mathbf{X}_1 \cap \mathbf{X}_2) = r'$

  If two overlapping parts are colinear, $dim(\mathbf{X}_1 \cap \mathbf{X}_2)$ will less than r. In this case, there are two possible conditions. Let's repeat the example above to explain. Assume there are two groups in two-dimension space. There are three intersection points for these two groups. These three points are collinear so that they are located on a line in one-dimension space, as Figure 3 shows. Fix the first group and align the second group through affine mapping. This may occur in two situations. One is that these two sets will be aligned through a rotation and shift, and another is that two sets will be aligned through a rotation or shift other than a reflection in respect to the line. These two conditions will create two different results, and one of them would accord with the original configuration. As a result, there is not enough information to distinguish which one will be correct. The same confusion would happen under $dim(\mathbf{X}_1) \geq dim(\mathbf{X}_2) > dim(\mathbf{X}_1 \cap \mathbf{X}_2)$.

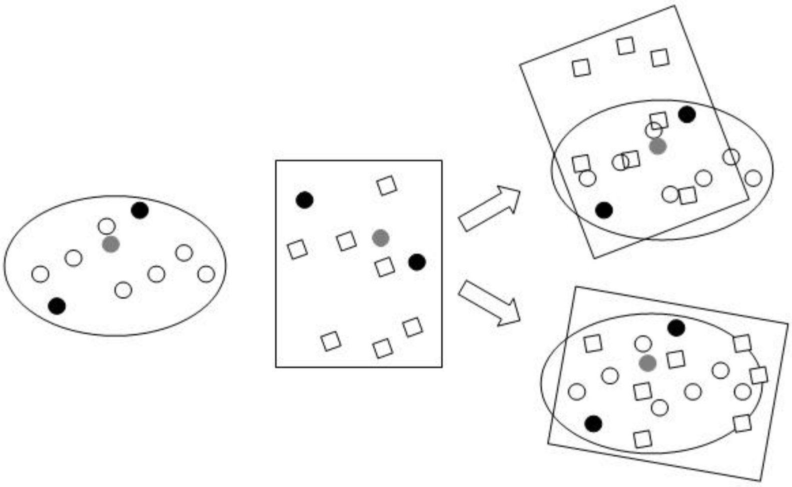


Fig. 3: Three solid points represent overlapping points of two groups. These three points has collinearity and allocate in a line.

In the following, we will show another simple example to explain the second condition:

Matrix

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{bmatrix}$$

is a five-dimensional set. Divide matrix $\mathbf{X}$ into two sets, $\mathbf{X}_1$ and $\mathbf{X}_2$. $\mathbf{X}_1$ and $\mathbf{X}_2$ are spanned by three basis with $dim(\mathbf{X}_1) = dim(\mathbf{X}_2) = 4$. Figure 4 shows the condition. In matrix $\mathbf{X}$, $\mathbf{x}_1$ and $\mathbf{x}_5$ are orthogonal to the set $\mathbf{X}_1 \cap \mathbf{X}_2$, and $\mathbf{x}_1$ is also orthogonal to $\mathbf{x}_5$, so the space spanned by $\mathbf{X}_2$ projects on the space spanned by $\mathbf{X}_1$ should be a three-dimensional plane. However, $\mathbf{x}_1$ and $\mathbf{x}_5$ will be considered as the same component and aligned through affine mapping in the process of combination step in SCMDS. Doing the wrong space alignment will cause matrix $\mathbf{X}$ to be reduced to a four-dimensional matrix and lose information from one dimension. So, choosing a suitable N_I (the number of overlapping points) and N_g (the size of each group) is very important. We hope that N_I and N_g are large enough so that we can lower the chance of the appearance of colinearity for the intersection part as well as the chance of dimensionality of the intersection part so that it is less than the dimensionality of total data when we do the random grouping.

In conclusion, the number of overlapping points would be at least the minimal of dimensionality of two groups. This can be denoted as
$dim(\mathbf{X}_1 \cap \mathbf{X}_2) \geq min\{dim(\mathbf{X}_1), dim(\mathbf{X}_2)\}$.

Remarks:

1. In the process of SCMDS, combining two overlapping groups should fix one group with larger dimensionality and operate affine mapping on another group with smaller dimensionality, and then align it with the former. If two groups have the same dimensionality,

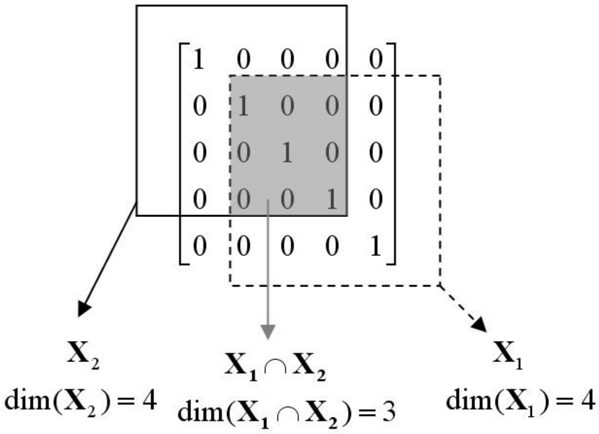


Fig. 4: Split a dataset into two overlapping groups. Each groups have four dimensionality. The overlapping part has only three dimensionality.

choose any one as the central group and align the two groups through applying affine mapping on the other one.

2. In SCMDS, split data set $\mathbf{X}$ into two overlapping groups, $\mathbf{X}_1$ and $\mathbf{X}_2$. The number of intersection points, N_I, should be large enough such that $dim(\mathbf{X}_1 \cap \mathbf{X}_2) = min\{dim(\mathbf{X}_1), dim(\mathbf{X}_2)\}$.

Theorem: Let there be a matrix $\mathbf{X}_{p\times n} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$, $\mathbf{x}_i \in \mathbb{R}^p$, $i = 1, 2, \cdots, n$, $\mathbf{X}_1$ and $\mathbf{X}_2$ are two subsets of $\mathbf{X}$, $\mathbf{X} = \mathbf{X}_1 \cup \mathbf{X}_2$, $\mathbf{X}_1 \cap \mathbf{X}_2 \neq \emptyset$. On $\mathbf{X}_1$ and $\mathbf{X}_2$ apply MDS separately, resulting in new matrices denoted as $\mathbf{X}'_1$ and $\mathbf{X}'_2$. There exist minimal orthogonal sets such that $\mathbf{X}'_1 \subset span\{v_1, v_2, \cdots, v_{r_1}\}$ and $\mathbf{X}'_2 \subset span\{w_1, w_2, \cdots, w_{r_2}\}$, $\{v_i\}_{i=1}^{r_1}, \{w_j\}_{j=1}^{r_2} \in \mathbb{R}^r$, $r < p$, where r is the number of main component we keep, $dim(\mathbf{X}'_1) = r_1 \geq r_2 = dim(\mathbf{X}'_2)$. $\mathbf{Y}_1$ and $\mathbf{Y}_2$ represent the overlapping part after applying MDS in two sets, respectively. If $rank(\mathbf{X}_1 \cap \mathbf{X}_2) = r_2$, the affine mapping of recombination process in SC-MDS will transform $\mathbf{X}'_2$ to a subset of $span\{v_1, v_2, \cdots, v_{r_1}\}$.

Proof:

Because $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are not centralized in the same center, we shift these two sets to the same center, say original point. Then, we rotate them to the same configuration expension. As the introduction of the previous recombination method, we apply QR decomposition to both $\mathbf{Y}_1 - \bar{\mathbf{Y}}_1\mathbf{1}^T$ and $\mathbf{Y}_2 - \bar{\mathbf{Y}}_2\mathbf{1}^T$. We have

$\mathbf{Y}_1 - \bar{\mathbf{Y}}_1\mathbf{1}^T = \mathbf{Q}_1\mathbf{R}_1$ and $\mathbf{Y}_2 - \bar{\mathbf{Y}}_2\mathbf{1}^T = \mathbf{Q}_2\mathbf{R}_2$

$\because$ $\mathbf{Y}_1$ is the representation of $\mathbf{X}_1 \cap \mathbf{X}_2$ in $\mathbb{R}^{r_1}$

$\therefore$ $\mathbf{Q}_1$ only has r_2 orthogonal column vectors

$\therefore \mathbf{Q}_1 \in \mathbf{M}_{r_1 r_2}(\mathbb{R})$

Similarly, $\mathbf{Q}_2 \in \mathbf{M}_{r_2}(\mathbb{R})$

$\because$ $\mathbf{R}_1 = \mathbf{R}_2$ are the triangular matrix of $\mathbf{M}_{r_2}(\mathbb{R})$

We have

$\mathbf{Q}_1^T(\mathbf{Y}_1 - \bar{\mathbf{Y}}_1\mathbf{1}^T) = \mathbf{Q}_2^T(\mathbf{Y}_2 - \bar{\mathbf{Y}}_2\mathbf{1}^T)$

$\mathbf{Y}_1 = \mathbf{Q}_1\mathbf{Q}_2^T\mathbf{Y}_2 - \mathbf{Q}_1\mathbf{Q}_2^T\bar{\mathbf{Y}}_2\mathbf{1}^T + \bar{\mathbf{Y}}_1\mathbf{1}^T$

$\mathbf{U} = \mathbf{Q}_1\mathbf{Q}_2^T$, $\mathbf{b} = -\mathbf{Q}_1\mathbf{Q}_2^T\bar{\mathbf{Y}}_2\mathbf{1}^T + \bar{\mathbf{Y}}_1\mathbf{1}^T$

$\therefore$ $\mathbf{U} = \mathbf{Q}_1\mathbf{Q}_2^T$ is an operator that maps vectors from $\mathbb{R}^{r_2}$ to $\mathbb{R}^{r_1}$, and the span of $\{q_1, q_2, \cdots, q_{r_1}\} \subset \mathbf{C}(\mathbf{X}_1')$ (column space of $\mathbf{X}_1'$)

So as $\mathbf{Q}_1\mathbf{Q}_2^T\bar{\mathbf{Y}}_2\mathbf{1}^T$, and $\bar{\mathbf{Y}}_1$ is the linear assembly of $\mathbf{X}_1'$ and is contained in $span\{v_1, v_2, \cdots, v_{r_1}\}$.

$\therefore \tilde{\mathbf{X}}_2 = \mathbf{U}(\mathbf{X}_2') + \mathbf{b} \subset \mathbf{C}(\mathbf{X}_1')$

$\therefore$ $\tilde{\mathbf{X}}_2$ is a subset of $span\{v_1, v_2, \cdots, v_{r_1}\}$

Now, extend the simplified case into K groups. Apply the combining method above, we combine the first two groups and obtain a new spatial configuration $\tilde{\mathbf{X}}_{(1)}$. Then, combine the new spatial configuration with the next groups $\mathbf{X}_3'$ with the same rule. If $dim(\mathbf{X}_3') \geq dim(\tilde{\mathbf{X}}_{(1)})$, align $\tilde{\mathbf{X}}_{(1)}$ with $\mathbf{X}_3'$ based on $\mathbf{X}_3'$ and so forth. Low dimensional groups will be absorbed into high dimensional groups. Repeat until all groups combine in an identity space. In the end, the whole spatial configuration will be spanned by basis of one of the groups. In other words, as long as, the space spanned by a group whose basis is identical with the space spanned by the basis of data $\mathbf{X}$, the result of SCMDS should be consistent with CMDS.

Remark:

If the dimensionality of at least one of the groups is equal to the dimensionality of the total data set, the result of SCMDS is the same as the result of CMDS apart from a rotation effect.

3.3 Computation Complexity

When comparing SC-MDS with CMDS, computation complexity is reduced to $O(N)$ from $O(N^3)$. Let's use the same notation with above. Assume there are K chain subsets belong to a dataset with N points. Let N_I represent the intersection size for a two-neighbor subset; the size of each subset is $N_g = \alpha(r+1)$, where α is a constant. Then, $KN_g - (K-1)N_I = N$ and $K = \frac{(N-N_I)}{(N_g-N_I)}$. In SC-MDS, we apply CMDS on each subset with computation complexity $O(N_g^3)$, and we use QR decomposition to combine each subset with computation complexity $O(N_I^3)$. The total computation complexity of SC-MDS is

$$\begin{aligned} KO(N_g^3) + (K-1)O(N_I^3) &= \frac{(N-r-1)}{(\alpha-1)(r+1)}O(\alpha^3(r+1)^3) + \frac{(N-\alpha(r+1))}{(\alpha-1)(r+1)}O((r+1)^3) \\ &\sim O((r+1)^2N) \end{aligned}$$

Computation complexity of SC-MDS converges to $O(N)$ as $(r+1) \ll N$. In Tzeng, Lu and Li (2008), it assumed that $p \ll N$. Actually, if we have some prior information about the small rank of the data set, this assumption $p \ll N$ is not necessary; if we do not have any, p is the upper bound of essential dimension. For insurance, the assumption is needed.

Some recommended conditions for SCMDS

1. When grouping dataset $\mathbf{X}$ into K chain subsets $\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_K$, we should notice that the intersection of each subset with its neighbor should not be empty; i.e., $\mathbf{X}_i \cap \mathbf{X}_{i+1} \neq \emptyset$.

2. Each group size N_g can't ne less than the dimensionality of $\mathbf{X}$ data. The front section discusses the effect when $N_g < r$. Some information will be lost, which affects the accuracy.

3. The intersection size between two neighboring groups in N_I should be at least $p+1$. Defining any points in p-dimension space needs $p+1$ pieces of distance information. If $N_I < p+1$, then there may be more than one location that satisfies the distance measurement.

4. Overlapping points in groups should contain points not only in the neighborhood but also in the distance. Under the same error disturbance when performing rotation, adjacent points will cause larger effects than distant points will, as shown in Figure 5.

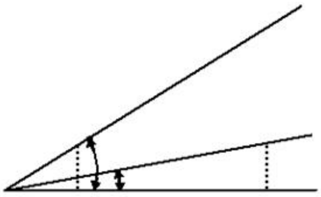


Fig. 5: With the same length of two dotted line as the error, the effect on the rotation angle is slight when the point far away from the center of rotation.

Now that there are some constraints on choosing N_I, N_g by the dimensionality of data r, does it mean that larger N_I, N_g means better performance for SC-MDS? It is an important task to estimate p and choose appropriate N_I, N_g values.

3.4 How to choose optimal values for N_I, N_g

It has been mentioned that N_g is at least equal to the dimensionality of data r in Tzeng, Lu and Li (2008). Once N_g is less than r, the dimension of space constructed by SC-MDS will be less than an essential dimension and will cause inaccuracy. $N_I > p + 1$ because we want to allocate a new point on p-dimension space, so we need information about relativity with $p+1$ non-independent points or else affine transformation would not be unique. There should be enough overlapping points offering information to decide the correct spatial configuration. All in all, N_I, N_g affect the computation complexity and the accuracy of SC-MDS.

By simulation, we try to find some relationship between N_I, N_g, time cost and error. Observe if they have some special features or patterns that we are interested in, and further check what information they could offer. Here, we assume $\mathbf{X}_i = (x_{1i}, x_{2i}, \cdots, x_{pi})^T$ comes from mixture multivariate normal without covariance, $\mathbf{X}_i \sim MVN_1(\mu_1, \sigma_1^2) + MVN_2(\mu_2, \sigma_2^2)$. Generate N data randomly, and obtain an $N \times p$ data set $\mathbf{X}$. We consider $\mathbf{X}$ as the final spatial configuration we would obtain after applying MDS. Apply SC-MDS on $\mathbf{X}$, then compare the result with $\mathbf{X}$ and record the time cost and error for SC-MDS. We don't generate distance matrices instead of generating a spatial configuration because it is hard to find the true dimensionality for a larger size distance matrix, and we can't see the effect of data dimensionality on different N_I, N_g values.

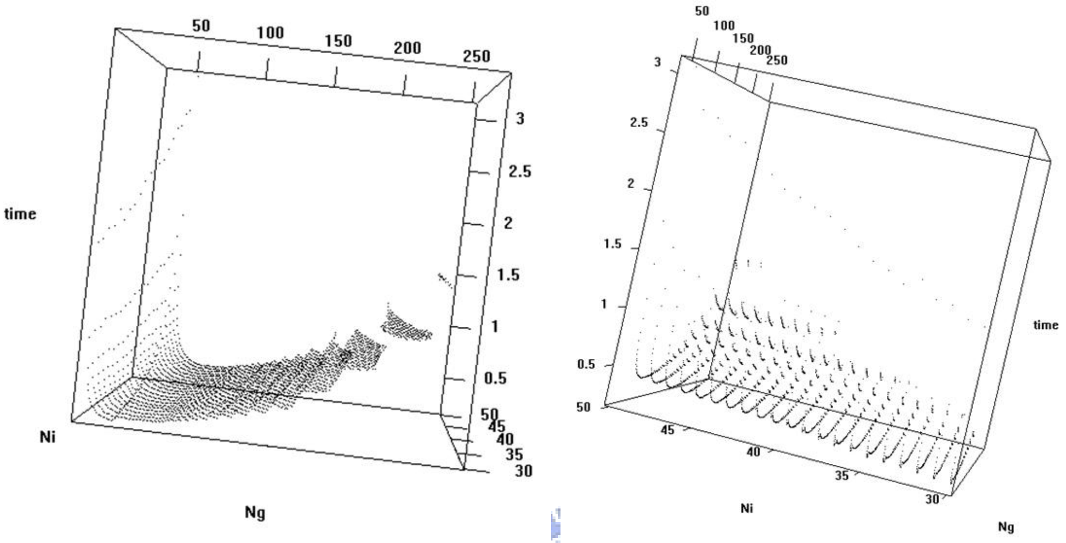


Fig. 6: The time cost for SC-MDS with variety of N_I and N_g

Figure 6 is the time cost of applying SC-MDS on X with $N = 1000$, $p = 30$ under different parameters N_I, N_g. Repeat the simulation process under the same parameter settings, then take an average for the time cost and have a scatter plot for N_I, N_g and time cost. The outcome is shown in Figure 6. We can observe some features from the figure. It costs less time as N_I is small. We can explain the phenomenon from the computation complexity equation, $KO(N_g^3)+(K-1)O(N_I^3)$. If N_I is large, the number of groups would become small under fixed N_g. However, the increasing rate of $O(N_I^3)$ is larger than the decreasing rate of $K = \frac{N-N_I}{N_g-N_I}$ affected by N_I. So, when N_I increases, the time cost will also increase.

Now, observe how N_g affects time cost. We can observe that N_g will impact time cost more significantly than N_I. It seems that the optimal decision for N_g will have a minimal time cost. Let's go back to the equation

$$KO(N_g^3) + (K-1)O(N_I^3) = \frac{N-N_I}{N_g-N_I}O(N_g^3) + \frac{N-N_g}{N_g-N_I}O(N_I^3) \tag{2}$$

We try to do some reasonable explanation. If N_g is too small, the time cost on each group applying MDS will be efficient; still a small N_g will make grouping number increase. In this case, the total time cost will increase. On the other hand, if N_g is too large, then even if the grouping number is decreased, the time cost in applying CMDS on each group will increase. This will increase the computing time cost. Figure 7 shows the rough relationship between N_g and time cost.

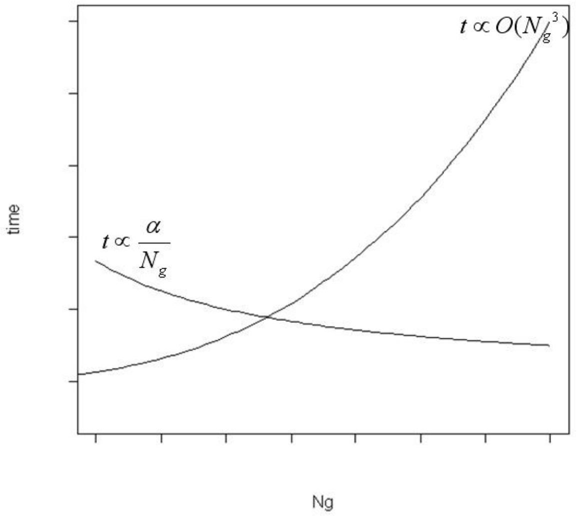


Fig. 7: Rough relationship between N_g and time cost.

We want to find the optimal N_g so that SC-MDS has the most efficient performance. So, we do some simulations on a different N_g but fix N_I, N, and p values, where N_I, N and p are chosen randomly from intervals $[p+1, 0.1N]$, $[1000, 3000]$, and $[20, 50]$, respectively. Also, we record the time cost and all the parameter information. Each point that we record is the average of ten times the simulation result under the same parameter setting. Find the points with shortest computing time in fixed N_I, N, p values and various N_g values. The result is shown in Figure 8. We also fit these points by a linear model in Figure 8.

In the following, we try to derive the theoretical value and compare the results by simulation. Assume $N_g = \alpha N_I$. s and q are third order coefficients for computation complexity equation in CMDS and QR decompositions, respectively. Here, we ignore other terms except the leading term. Substitute into equation (2) to obtain

$$\frac{N-N_I}{(\alpha-1)N_I}O(\alpha^3 N_I^3) + \frac{N-N_g}{(\alpha-1)N_I}O(N_I^3) = \frac{N-N_I}{(\alpha-1)N_I}s\alpha^3 N_I^3 + \frac{N-N_g}{(\alpha-1)N_I}qN_I^3$$

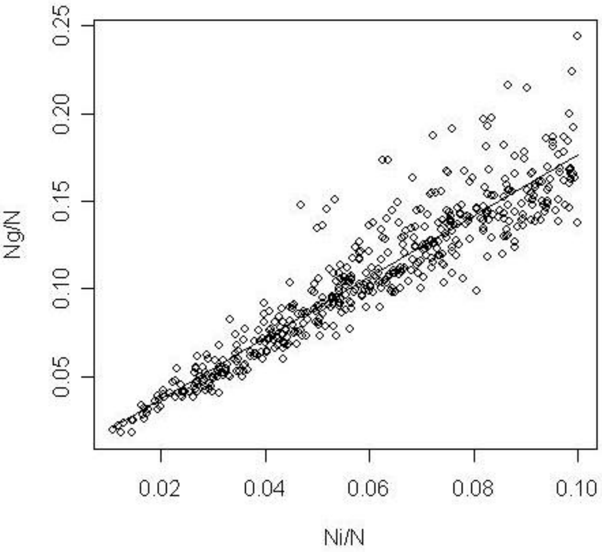


Fig. 8: Each points represent the minimal time cost for fixed N_I, N_g and N.

Let $g(\alpha) = \frac{N-N_I}{(\alpha-1)N_I} s\alpha^3 N_I^3 + \frac{N-N_g}{(\alpha-1)N_I} qN_I^3$ and we want to find α such that $g(\alpha)$ has a minimum.

$$\frac{dg(\alpha)}{d\alpha} = \frac{3(N-N_I)sx^2N_I^2}{(x-1)} - \frac{(N-N_I)sx^3N_I^2}{(x-1)^2} - \frac{N_I^3 q}{(x-1)} - \frac{(N-xN_I)q}{(x-1)^2} N_I^2$$

$$g'(\alpha) = 0$$

$\alpha = \frac{((s+2q+2\sqrt{q(q+s)})s^2)^{1/3}}{2s} + \frac{s}{2((s+2q+2\sqrt{q(q+s)})s^2)^{1/3}} + \frac{1}{2}$(there are still two imaginary roots)

$$g''(\alpha) = \frac{6(N-N_I)sxN_I^2}{(x-1)} - \frac{6(N-N_I)sx^2N_I^2}{(x-1)^2} + \frac{2(N-N_I)sx^3N_I^2}{x(x-1)} + \frac{2N_I^3 q}{(x-1)^2} + \frac{2(N-xN_I)q}{(x-1)^3} N_I^2$$

$$g''(\alpha|s=26, q=2/3) = 454.20(N-N_I)N^2 + 5.21N_I^3 + 10.31(N-1.51N_I)N_I^2 > 0$$

The leading coefficient is $\frac{2}{3}$ for QR-Decomposition by the Householder transformation (reference, Golub's book, page no.225), and the leading coefficient is 26 for the R-SVD algorithm (reference, Golub's book, page no.254). Hence we can have the result $\alpha = 1.51$. We think this result is quite close to the result $\alpha = 1.73$ by simulation.

In conclusion, the best choice for Ng is about 1.51 times Ni.

We continue on to compare variations in parametersN_I, N_g. Assume $\mathbf{D} = [d_{ij}]_{n\times n}$ is the Euclidean distance matrix of original dataset, and $\hat{\mathbf{D}} = [\hat{d}_{ij}]_{n\times n}$ is the Euclidean distance matrix of a new spatial configuration by SC-MDS. We define STRESS to measure the error to SC-MDS as

$$STRESS = \sqrt{\frac{\sum_{i,j}(d_{ij} - \tilde{d}_{ij})^2}{\sum_{i,j} d_{ij}^2}}$$

where $\tilde{d}_{ij} = \frac{\hat{d}_{ij}}{s}$, $s = \frac{\max_{i,j}\hat{d}_{ij}}{\max_{i,j} d_{ij}}$. The higher STRESS is, the larger error we have.

In the same simulation process, the result is shown in Figure 9. STRESS declines sharply on $N_I = 31$, and has a smooth and mild decrease in error after $N_I = 31$. In Figure 9, it is apparent that error is maintained in the 10^{-11} level. This outcome is consistent with what we discussed. N_I is at least equal to the dimensionality of data plus one. Of course, to avoid the colinearity problem as we discussed in last section, choosing a large N_I is a good method. So we can consider choosing a large N_I in the tolerance of computation cost.

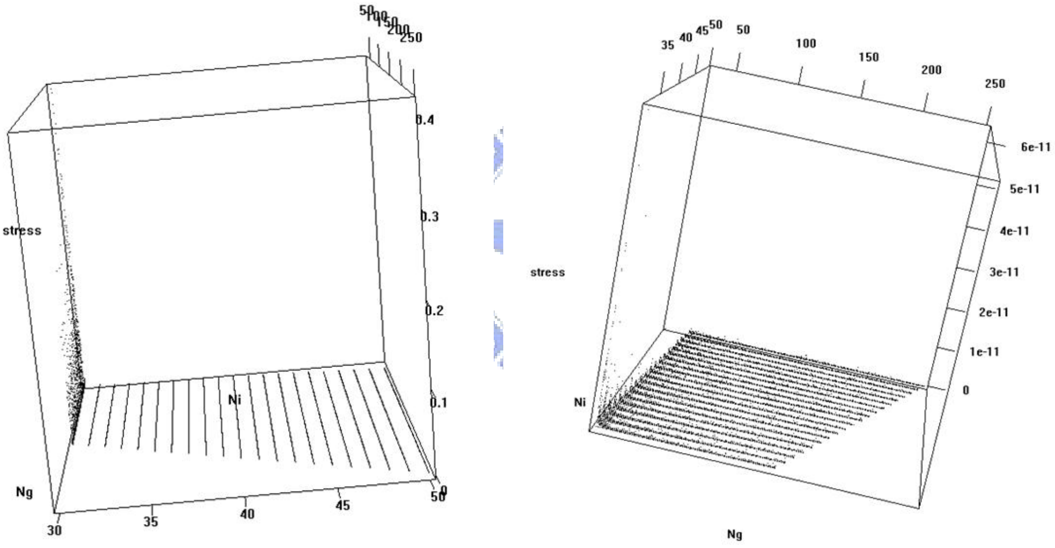


Fig. 9: The stress for SC-MDS with variety of N_I and N_g.

3.5 Hidden dimensionality of sample

Now that we know that the choice of N_I will strongly affect the accuracy of SC-MDS, we can use the information from the variety of stress to decide the hidden dimensionality of data. Scree plot would be a good tool to help us solve this problem. Let r be the hidden dimensionality of sample. Error will decrease significantly as $N_I = r + 1$. As the result shows in Figure 10 where $r = 30$, when $N_I > 31$ the error of SC-MDS decrease to 1.0E-13 level.

However, variation is quite large after $N_I > 31$. It is hard to find a criterion to judge which N_I is large enough. But it is easy to observe the relative variation through the scree plot.

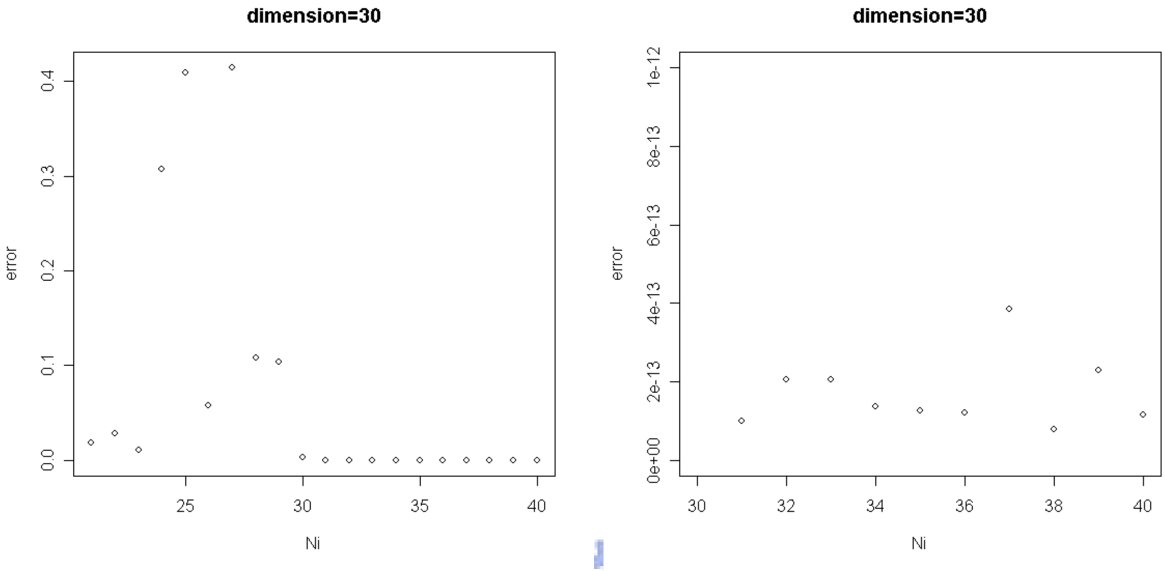


Fig. 10: Variety of error (stress) with different N_I

4 Missing value

4.1 An intuitive method

The third challenge for MDS is dealing with the missing value. The most common solution is filling up missing part with a substitute value. The substitute value can be any kind of estimator. However, estimators may cause some bias. This would disturb the MDS process. SC-MDS has an advantage for dealing with missing data, because SC-MDS doesn't need the entire distance matrix. Only the data in the neighborhood of main diagonal are needed. If we can shift the missing part to an off-diagonal region, missing values will have no impact on the SC-MDS process. So how can we permute the index of data such that the missing part will be away from the main diagonal part (the gray region showed in Figure 11). A straight idea is to check whether there is missing value in the diagonal region that is showed as gray shadow in figure 11. If there is a missing value, find a column such that after swapping, the missing value can be moved to the white region. Notice that we should swap the rows and the columns at the same time to keep the symmetric property of the distance matrix. Repeat the process until all the missing values are moved off the diagonal area. Then we can process SC-MDS to reconstruct data coordinate. The flow chart is shown in Figure 12 to describe the process more clearly.

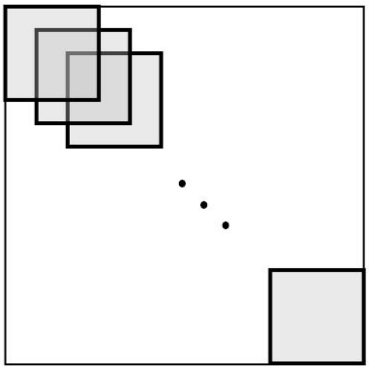


Fig. 11: Perfect permutation for SC-MDS. There is no missing value in the gray area.

Here we define the ratio of missing value as

$$\frac{\text{the number of missing values}}{N(N-1)/2}$$

We count one missing value when d_{ij} and d_{ji} are both missing. In this process, the convergence velocity is intolerable when ratio of missing values is higher than 0.03, because it is easy to fall into a swapping cycle. That is also why we select the swapping columns randomly to prevent this condition from occurring.

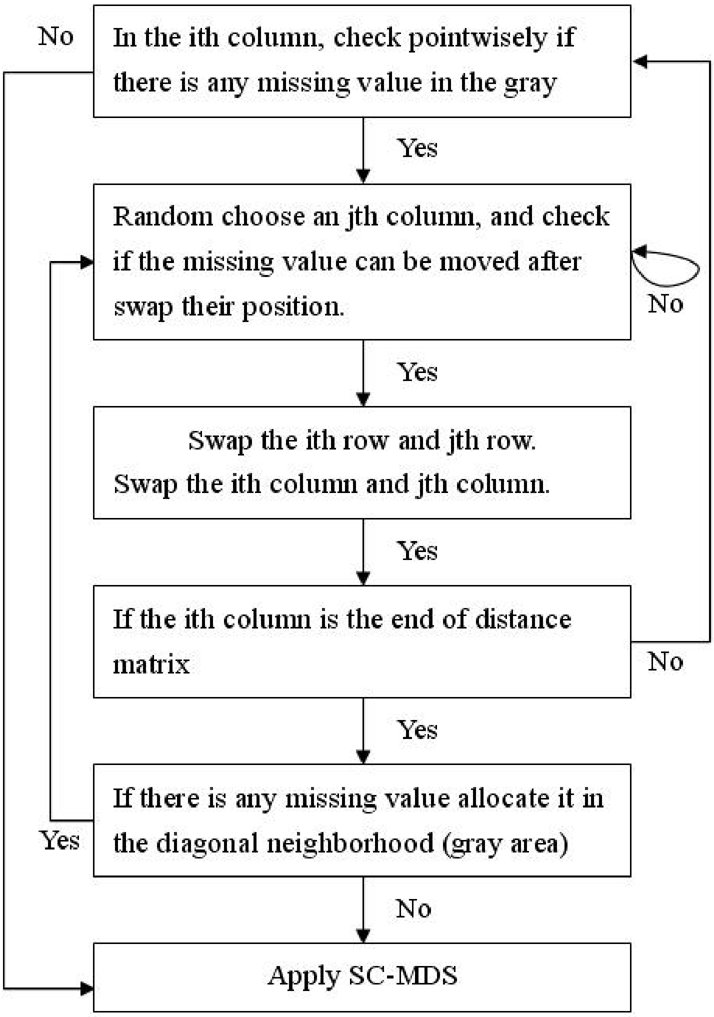

In the ith column, check pointwisely if there is any missing value in the gray
No
Yes
Random choose an jth column, and check if the missing value can be moved after swap their position.
No
Yes
Swap the ith row and jth row.
Swap the ith column and jth column.
Yes
If the ith column is the end of distance matrix
No
Yes
If there is any missing value allocate it in the diagonal neighborhood (gray area)
Yes
No
Apply SC-MDS

Fig. 12: Flow chart for dealing with missing value problem by intuitive method.

Another idea is to fill up the missing part when we sort the index to speed up the convergence velocity. In the process of permutation, we apply SC-MDS on a partial of distance matrix at the same time as long as SC-MDS available to be used. Hence, we can decrease the ratio of missing values and speed up the converge velocity. In Figure 11, we assume the second and the third group has no missing value in the gray area, and there are some missing values in the white area. Thus, we can apply SC-MDS to find the point configuration and then calculate its distance matrix for these points. The missing part in the white area can be filled up. In this case, once we find some part of the distance matrix that has this feature, which has missing values in the upper right corner and lower left corner and complete data in the diagonal region, we can fill up the missing part by SC-MDS. With this method to decrease the ratio of missing value, convergence velocity has increase. But the convergence velocity is still intolerable when ratio of missing value is higher than 0.06.

4.2 A SC-MDS basic concept based method

In the third method, we don't focus on how to permute these objects such that all these missing part can be removed from the neighborhood around the main diagonal part. Instead, we utilize the basic concept of SC-MDS to combine only each complete part in the data set to compute the coordinate of objects. Let's see an extreme example. Suppose there is a distance matrix with dimensionality r and size N (meaning there are N objects). All elements in this distance matrix are missing values except the elements from the i-th column to the (i+r)-th column and from the i-th row to the (i+r)-th row. This matrix can be shown as in Figure 10. Gray represents the missing part and white represents the complete part. As shown in the figure, this distance matrix has information in the white cross region. On the other hand, we have complete information about the r+1 objects. Obviously, we can not find a permutation of index (shown in Figure 11) such that the chain of any two neighbor groups have a partial overlap at the same time. In this case, the only way to solve this question is to align all the other points individually with the r+1 points as the center group. The split step will show in figure 13 on the right side. We split the whole data set into N-(r+1) groups, except the (r+1) points. The remainder will be allocated to a different group, and each group will include those (r+1) points with complete information and one point of the remainder. Each groups will have size r+2. Therefore, we can combine each group in the r dimension coordinate space through the (r+1) overlapping part after applying MDS on each part.

We have two remarks here.

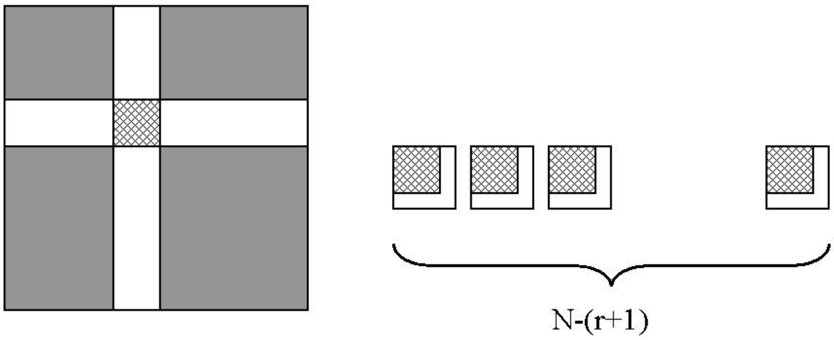


Fig. 13: An extreme example for missing value problem. This square represent a distance matrix and there is information only on the white cross region.

1. Do not persist in permuting objects. In the condition that the data has no missing value (complete dataset), each chain group has different overlapping part (see in Figure 11). But with the effect of missing data, we don't have sufficient information to do this. Hence, to make good use of the information in existence, we allow the overlapping part to occur repeatedly in different groups.

2. When we know the actual dimension of samples and the overlapping region is greater than the dimension, the random permutation plays no role in improving accuracy of SC-MDS. In the case of missing value problem, we should focus on how to find the max group that the pairwise distance has no missing value. Then using this group as the center to combine other points, then we can process SC-MDS to get the coordinates of the other points.

How do we utilize the concept of SC-MDS to deal with the missing value problem practically?

- Let a set $G = \{1, 2, \cdots, N\}$ records that the whole distance matrix is composed of which objects.

- At fist, we find the largest complete data groups $G_1 = \{g_{11}, g_{12}, \cdots, g_{1k_1}\}$, g_{1i} is the index of samples such that the distance matrix composed of the set of object in G_1 have no missing value.

- In the second step, we want to find a set G_2 which satisfies $length(G_1 \cap G_2) \geq r + 1$, $length(G_2 \cap (G \setminus G_1)) \geq 1$ and the distance matrix is composed of objects in G_2 have no missing value. Then, we apply the SC-MDS process on two groups to find the point configuration.

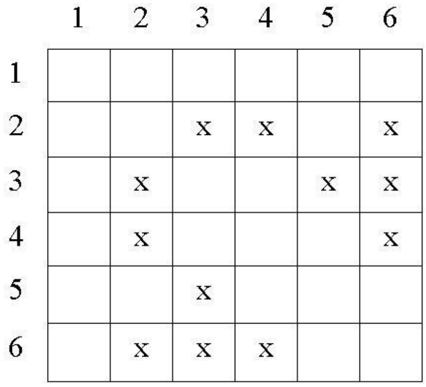
- The next step is to find a set G_3 which satisfies G_2 which satisfies $length(G_1 \cap (G_1 \cup G_2)) \geq r+1$, $length(G_2 \cap (G \setminus (G_1 \cup G_2))) \geq 1$, and the distance matrix composed of objects in G_3 has no missing value. Then, we apply an MDS process on group 3 to find the point configuration and use the combine step to align with group 1 and group 2.

$$\vdots$$

- The next step is to find a set G_i which satisfies $length(G_1 \cap (\bigcap_{k=1}^{i-1} G_k) \geq r+1)$, $length(G_2 \cap (G \setminus \bigcup_{k=1}^{i-1} G_k)) > 1$, and the distance matrix composed of objects in G_i have no missing value. Then, we apply the MDS process on group i to find the point configuration and use the combine step to align with $\bigcup_{k=1}^{i-1} G_k$. We consider $\bigcup_{k=1}^{i-1} G_k$ as the center group of G_i.

Then we could get the spatial configuration of all objects.

The following is an easy example to help you to understand the process more clearly. Assume there are six objects and their distance matrix is expressed as the following. A cross symbols a missing value.



	1	2	3	4	5	6
1						
2			x	x		x
3		x			x	x
4		x				x
5			x			
6		x	x	x		

To get the largest complete data groups, we delete an object that has the most cross marks (missing value).

	1	3	4	5	6
1					
3				x	x
4					x
5		x			
6		x	x		

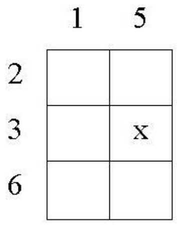
	1	4	5
1			
4			
5			

$G_1 = \{1, 4, 5\}$

To get G_2 which satisfies $length(G_1 \cap G_2) \geq r + 1$, $length(G_2 \cap (G \setminus G_1)) > 1$ and a distance matrix composed of objects in G_2 that has no missing value. Assume we only need two overlapping objects, and find the object which has no missing value with at least two object in $G_1 = \{1, 4, 5\}$.

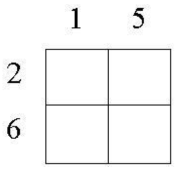
	1	4	5
2		x	
3			x
6		x	

Choose two overlapping objects which have the fewest missing values.



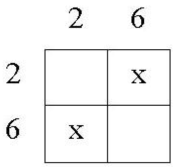
	1	5
2		
3		x
6		

Delete the object which has the most cross marks



	1	5
2		
6		

Check if there is any missing value in relation of object 2 and object 6.



	2	6
2		x
6	x	

$G_2 = \{1, 5, 2\}$ To get G_3 which satisfies $length(G_1 \cap (G_1 \cup G_2)) \geq r + 1$, $length(G_2 \cap (G \setminus (G_1 \cup G_2))) > 1$, and the distance matrix composes by objects in G_3 have no missing value.

Choose two overlapping objects which have fewest missing value.

	1	2	4	5
3		x		x
6		x	x	

	1	5
6		

$G_3 = \{1, 5, 6\}$

Repeat the process consist with former.

	1	2	4	5	6
3		x		x	x

	1	4
3		

$G_4 = \{1, 3, 4\}$

Apply SC-MDS on G_1, G_2, G_3, G_4 in sequence. The tolerance of missing data depends on the number of the overlapping numbers. We perform the simulation with $N = 1000$ and $r = 3$.The tolerance of ratio of missing value is around 0.3 based on the simulation results. However, SC-MDS is also possible to operate successfully when ratio of missing value is more than 0.3. The missing value should spread well enough. What does ”spread well” means? The first sufficient condition is that each column should have missing value less than $(N - r - 2)$, because each object needs at least the information about the relation of itself and the overlapping part. Moreover, each group should have at least r+1 overlapping points connecting with its center group. For example, if the missing value is located as Figure 15, there are no overlapping region between the two groups (or the overlapping region is smaller than $r + 1$), then SC-MDS falt to reconstruct the coordinate from the given distance matrix.

The following is the time cost variation in different ratios of missing values. By intuition, the time cost should increase as the ratio of missing values increases. However, as shown in Figure 15, the time cost increases sharply when the ratio of missing values goes up, then decreases mildly when the ratio of missing values exceeds 0.13. The main reason for this is

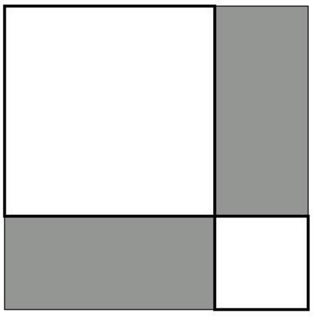


Fig. 14: Missing values do not spread well to employ SC-MDS.

in the sorting process. When we try to find the largest complete data group, we will choose certain objects as our overlapping points. Then, we will collect all the objects that have no missing values with these overlapping objects and delete one object from the collection at a time according to which has the most missing values until the distance between pairs of these objects have no missing values. In this process, the number of missing values in each column of the distance matrix will be sorted over and over. We compare the sorting process when the ratio of missing values is 0.13 and 0.30. The number of objects that have no missing values and that have overlapping objects will be larger when the ratio of missing values is 0.13 than when the ratio of missing values is 0.30. It will cost more time when the ratio of missing values is 0.13 than when the ratio is 0.30.

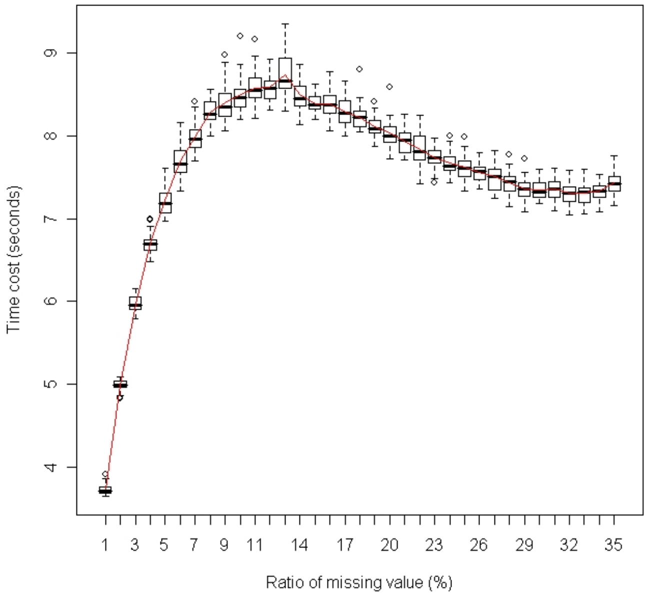


Fig. 15: Average time cost for SC-MDS with missing values with different ratio of missing value

5 Empirical Study

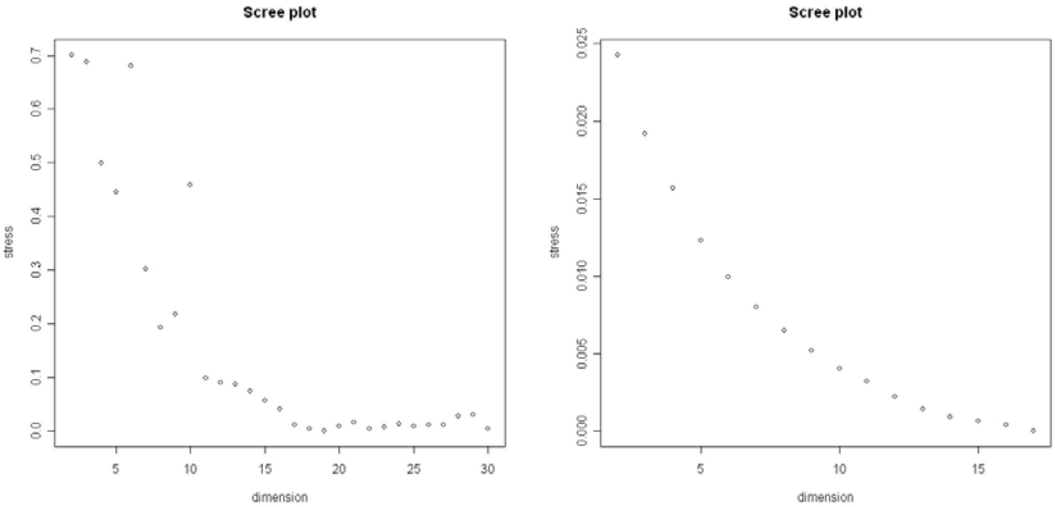


Fig. 16: Scree plot of SC-MDS and CMDS

Yeast data obtained from Cho et al., 1998. It records 6457 genes whose expression changes during 17 hours. We keep 4000 genes which changes significantly by evaluating the ratio of standard error to mean for each gene, and remove the remainder 2457 genes. We apply SC-MDS on this data with 4000 genes. On the other hand, we can remove some values from distance matrix of this data randomly. Then, we use SC-MDS to reconstruct the distance matrix and evaluate the error by calculating stress. Then, we compare the performance of SC-MDS in both conditions. A scree plot is shown in Figure 16. The left panel is the scree plot of SC-MDS, and the right panel is the scree plot of CMDS. It can help us to estimate the hidden dimensionality of data. Here, we choose $r = 19$, $Ni = 20$, $Ng = 1.5 * 20 = 30$, and the ratio of missing value is 0.2. Stress of SC-MDS without missing values is $2.09 * 10^-9$, and time cost is 1.29 seconds. Stress of SC-MDS with missing values is 0.54, and time cost is 445.21 seconds. As we mentioned above, the tolerance of ratio of missing value is strongly related to the sample size and hidden dimensions. Especially when the missing value is randomly remove from the distance matrix, it is hard to achieve the "well spread" as we mentioned before. Consequently, the tolerance of ratio of missing value will decade. Figure 17 is the result of SC-MDS of data with missing values.

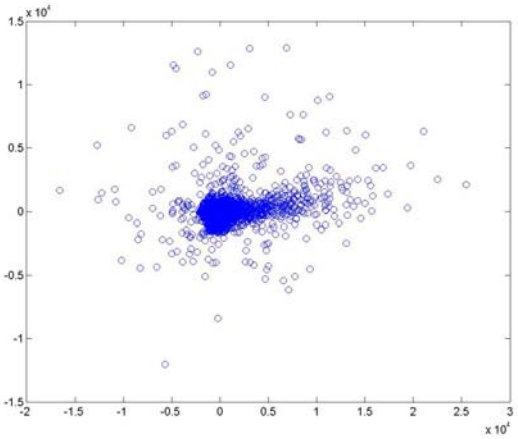


Fig. 17: Spatial configuration of yeast data with missing values

6 Conclusion

In this article, we try to complete the SC-MDS process. Parameters in SC-MDS have suggestions. SC-MDS will have the optimal performance when the number of overlapping points, N_I, is at least the dimensionality of samples plus one, and the size of group, N_g, is about 1.51 times N_I. We can also estimate the hidden dimensionality from the variation of error by changing the number of overlapping points. Besides, the combine step should have slight revision. When we process the combine step, we should take into account of the dimensionality of two groups. Consider the group with larger dimensionality as center groups to align two groups together. At last, we prove that the result of SCMDS is the same as CMDS in the sense of rotation effect, if there is at least one dimensionality of groups is equal to the dimensionality of the total data set.

Another result is using SC-MDS to solve the missing value problem. We apply the property of SC-MDS on dealing with incomplete data. The tolerance of missing values have improvement, ratio of missing value raises from 20% (Troyanskaya et al., 2001) to more than 30%.

7 Reference

1. A. Mead (1992). Review of the Development of Multidimensional Scaling Methods. *The Statistician*, **41**, 27-39

2. Buja A, Swayne DF, Littman M, Dean N, Hofmann H (2001). XGvis: Interactive data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics*

3. Chalmers M (1996). A linear iteration time layout algorithm for visualizing high-dimensional data. Proceedings of the 7th conference on Visualization, 127-ff

4. Donald A. Jackson (1993). Stopping Rules in Principal Components Analysis: A Comparison of Heuristical and Statistical Approaches. *Ecology*, Vol. 74, No. 8., 2204-2214

5. Florian Wickelmaier (2003). An Introduction to MDS. Sound Quality Research Unit, Aalborg University, Denmark.

6. G. Golub and C. Van Loan. *Matrix Computations*, Johns Hopkins University Press, Baltimore, (1996).

7. Gale Young and A. S. Householder (1938). DISCUSSION OF A SET OF POINTS IN TERMS OF THEIR MUTUAL DISTANCES. *PSYCHOMETRIKA*, Vol. 3, No. 1, 19-22

8. Jengnan Tzeng, Henry Horng-Shing Lu, and Wen-Hsiung Li (2008). Multidimensional Scaling for Large Genomic Data Sets.*BMC Bioinformatics* **9:179**

9. Mark Steyvers (2001). Multidimensional Scaling. *Encyclopedia of Cognitive Science*, Macmillan Reference Ltd

10. Morrison A, Ross G, Chalmers M (2003). Fast multidimensional scaling through sampling, springs and interpolation. *Information Visualization*, 2:68-77

11. Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein and Russ B. Altman (2001). Missing value estimation methods for DNA microarrays. *BIOINFORMATICS*, Vol.17, no.6, 520-525

12. Trevor F. Cox and Michael A. A. Cox (2001). *Multidimensional Scaling*, Second Edition, CHAPMAN & HALL/CRC