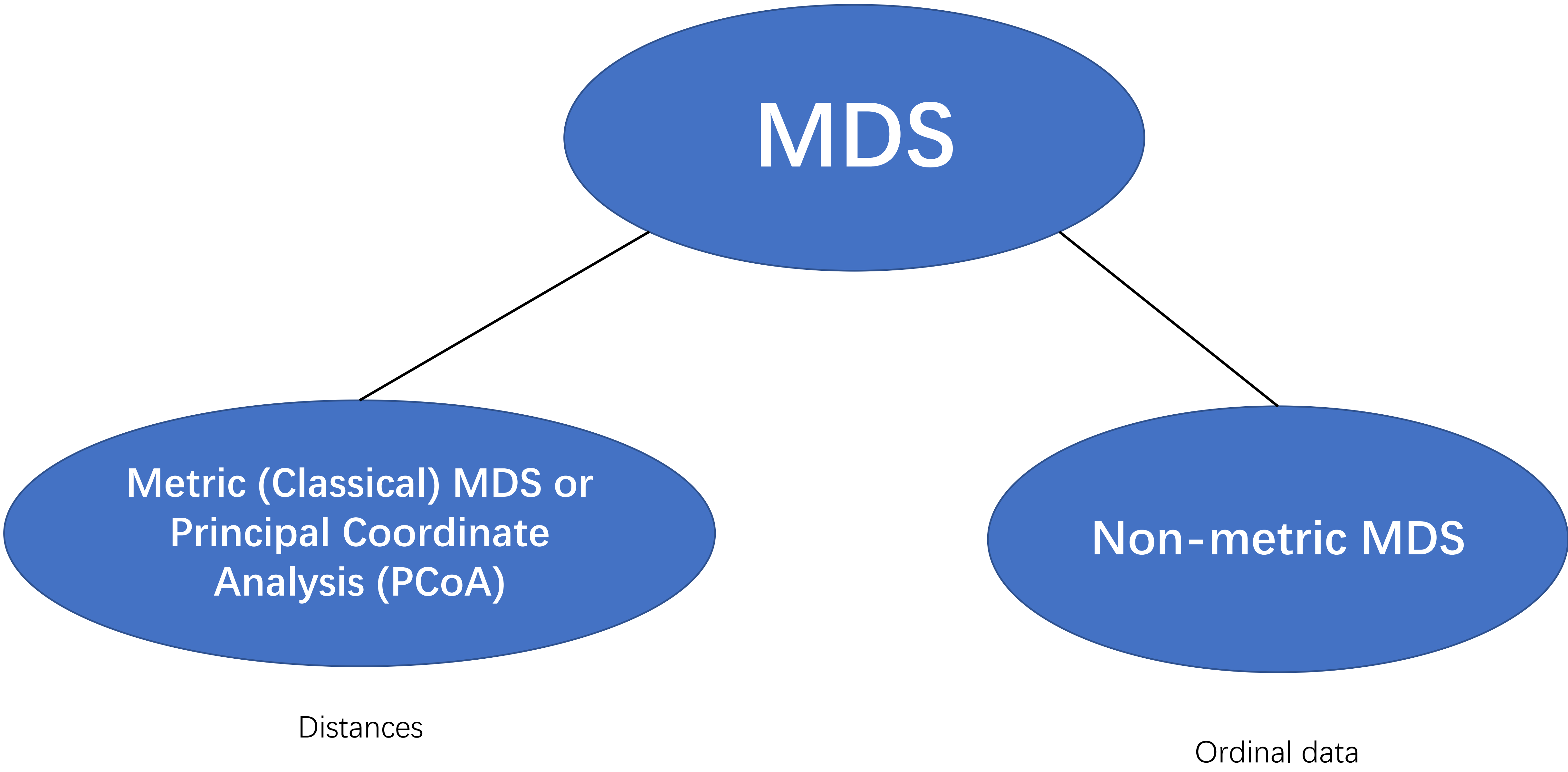
# Multidimensional scaling

Songmei Qin

2021/11/22

# Outline

- Types of Multidimensional Scaling (MDS)
- How does metric MDS actually work?


MDS
Metric (Classical) MDS or Principal Coordinate Analysis (PCoA)
Non-metric MDS
Distances
Ordinal data

# Non-metric MDS

- A simple example


1
2
3
4
5

# Metric (Classical) MDS

- It maps points residing in a higher-dimensional space to a lower-dimensional space while preserving the distances between those points as much as possible.

- **high dimensional data** ⟶ **lower dimensional data**

$X \in R^{M \times M}$ $Z \in R^{M' \times M}$

- Because of this, the **pairwise distances between points** in the lower-dimensional space are matched closely to their actual distances

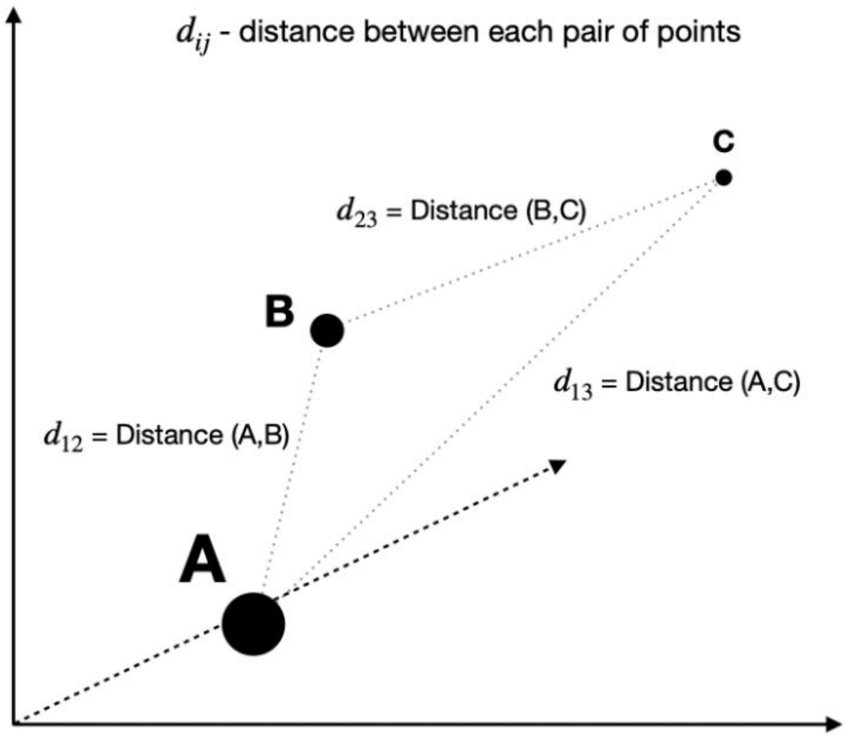
# Steps used by metric MDS algorithm

- **Step 1** — The algorithm calculates distances between each pair of points, as illustrated below.

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

**dissimilarity matrix：**

$$D := \begin{pmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,M} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,M} \\ \vdots & \vdots & & \vdots \\ d_{M,1} & d_{M,2} & \cdots & d_{M,M} \end{pmatrix}$$



# Steps used by metric MDS algorithm

- **Step 2** — With the original distances known, the algorithm attempts to solve the optimization problem by finding a set of coordinates in a lower-dimensional space that minimizes the value of Stress.

$$stress = \sqrt{\Sigma(\left\|Z_i - Z_j\right\| - d_{ij})^2}$$

$Z \in R^{M' \times M}$

$\mathrm{X} \in R^{M \times M}$

# Mathematical deduction

$$\sum_{\min} \left(\left\|z_i - z_j\right\| - d_{ij}\right)^2$$

上式没有唯一解

$$\left\|z_i' - z_j'\right\| = \left\|(z_i - z_0) - (z_j - z_0)\right\| = \left\|z_i - z_j\right\|$$

例如:

$$X = [x_1, x_2] = \begin{bmatrix} 0,1,0,0,0 \\ 1,0,0,0,0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & \sqrt{2} \\ \sqrt{2} & 0 \end{bmatrix}$$

$$z_1 = \left[\frac{\sqrt{2}}{2}\right], z_2 = \left[-\frac{\sqrt{2}}{2}\right]$$

$$z_1 = [0], z_2 = \left[\sqrt{2}\right]$$

# Mathematical deduction

$$\sum_{\min} \left( \|z_i - z_j\| - d_{ij} \right)^2$$

直接求 $Z$ 比较困难，转而求

$$B = Z^T Z \qquad Z \in R^{d' \times m} \qquad B \in R^{m \times m}$$

B是一个实对称矩阵，如果能够求得B，那么：

$$B = U \Lambda U^T = \left( \Lambda^{\frac{1}{2}} U^T \right)^T \left( \Lambda^{\frac{1}{2}} U^T \right) = Z^T Z$$

特征向量（$U$） 特征值（$\Lambda$）

$$d_{ij}^2 = \|z_i - z_j\|^2 = \|z_i\|^2 + \|z_j\|^2 - 2z_i^T z_j$$
$$= z_i^T z_i + z_j^T z_j - 2z_i^T z_j$$
$$= b_{ii} + b_{jj} - 2b_{ij}$$

$$\sum_{i=1}^{m} d_{ij}^2 = \sum_{i=1}^{m} b_{ii} + mb_{jj} + \sum_{i=1}^{m} 2z_i^T z_j = \sum_{i=1}^{m} b_{ii} + mb_{jj} + 2\left( \sum_{i=1}^{m} z_i^T \right) z_j$$
$$= \text{track(B)} + mb_{jj}$$

可通过调整偏移量使该项为0

$$\sum_{j=1}^{m} d_{ij}^2 = mb_{ii} + \sum_{j=1}^{m} b_{jj} + \sum_{j=1}^{m} 2z_i^T z_j = mb_{ii} + \text{track(B)}$$

$$\sum_{i=1}^{m} \sum_{j=1}^{m} d_{ij}^2 = 2m * \text{track(B)}$$

# Mathematical deduction

$$\begin{cases} d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij} \\ \sum_{i=1}^{m} d_{ij}^2 = \text{track(B)} + \text{m}b_{jj} \\ \sum_{j=1}^{m} d_{ij} = mb_{ii} + \text{track(B)} \\ \sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij} = 2\text{m*track(B)} \end{cases}$$

解方程可得:

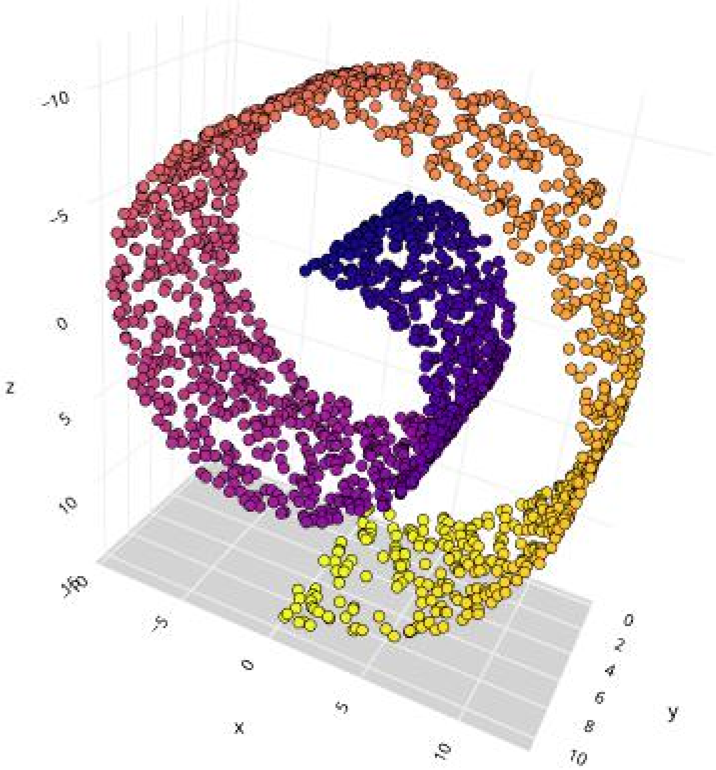
$$b_{ij} = -\frac{1}{2}\left(d_{ij}^2 - \frac{1}{\text{m}}\sum_{i=1}^{m} d_{ij}^2 - \frac{1}{\text{m}}\sum_{j=1}^{m} d_{ij}^2 + \frac{1}{\text{m}^2}\sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij}^2\right)$$

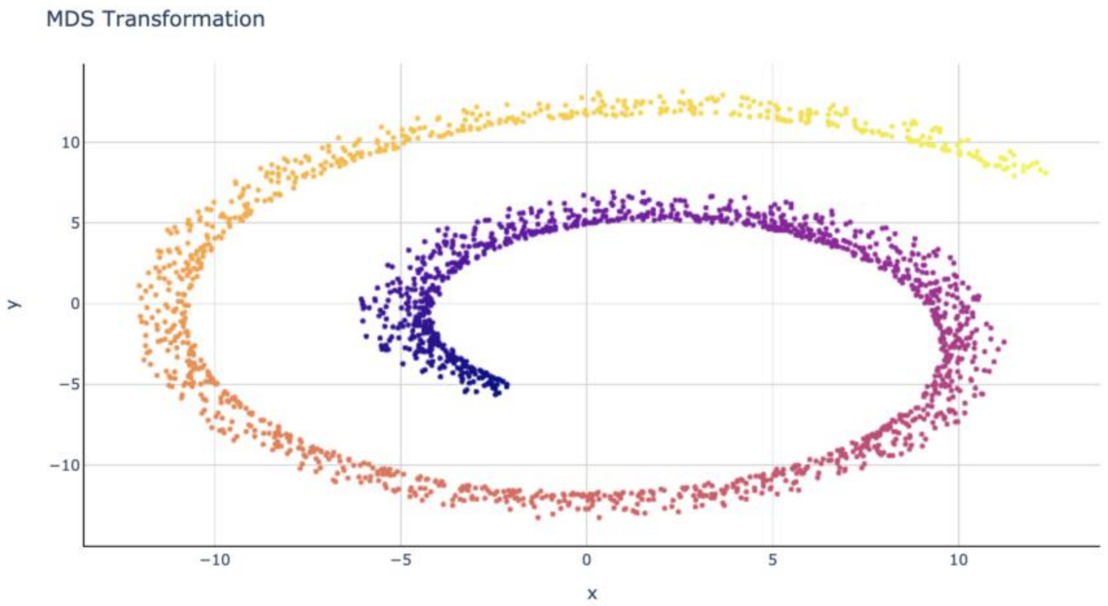
MDS 一般步骤:

（1） 利用给定数据计算距离矩阵（不相似矩阵）D

（2）计算降维后矢量z的互相关矩阵B

（3）对B进行特征值分解，选取较大的若干特征值与特征矢量获取Z


z
x
y


MDS Transformation
y
x
−10
−5
0
5
10

# reference


- https://en.wikipedia.org/wiki/Multidimensional_scaling
- https://towardsdatascience.com/mds-multidimensional-scaling-smart-way-to-reduce-dimensionality-in-python-7c126984e60b
- Tutorial:Multi-Dimension Scaling (MDS) (paperspace.com)
- Book: Modern Multidimensional Scaling, Ingwer Borg et al. 2005

# Thanks!