特征工程技术洞察 02
多维缩放

Jiangsheng Yu

05/06/2024

内积矩阵和距离矩阵的定义

向量 $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathbb{R}^d$ 或者矩阵 $\boldsymbol{A} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n)$ 的格拉姆矩阵或内积矩阵定义为

$$\boldsymbol{G}_{n\times n} = \boldsymbol{A}^\top \boldsymbol{A}，其中\ \boldsymbol{A} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n)$$

显然，格拉姆矩阵 $\boldsymbol{G}$ 的 (i, j) 元素 g_{ij} 即是 $\boldsymbol{x}_i, \boldsymbol{x}_j$ 的内积 $\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle = \boldsymbol{x}_i^\top \boldsymbol{x}_j$。

用 $d(\boldsymbol{x}, \boldsymbol{y})$ 表示 $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ 之间的距离 (distance)。譬如，$d(\boldsymbol{x}, \boldsymbol{0})$ 是向量 $\boldsymbol{x}$ 到原点的距离，即 $\boldsymbol{x}$ 的长度。对于一组给定的向量 $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathbb{R}^d$，其距离矩阵 (distance matrix) $\boldsymbol{\Delta}_{n\times n} = (d_{ij})$ 定义为

$$d_{ij} = d(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

内积矩阵和距离矩阵的关系

(距离矩阵与内积矩阵的关系) 设向量 $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathbb{R}^d$ 是中心化的，即满足 $\boldsymbol{x}_1 + \cdots + \boldsymbol{x}_n = \boldsymbol{0}$，则其内积矩阵 $\boldsymbol{G} = (g_{ij})$ 可由欧氏距离矩阵 $\boldsymbol{\Delta}$ 构造如下。

$$\boldsymbol{G} = -\frac{1}{2}\boldsymbol{H}\boldsymbol{\Delta}^2\boldsymbol{H}$$

其中，

$$\boldsymbol{\Delta}^2 = (d_{ij}^2)_{n\times n} = (\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2)_{n\times n}$$

$$\boldsymbol{H} = \boldsymbol{I}_n - \frac{1}{n}\boldsymbol{E}_n$$

多维缩放（MDS）问题

在几何上，如何把样本的维数降下来，同时又尽量不破坏它们之间的距离关系？多维缩放 (multidimensional scaling, MDS) 就是实现这类数据表示的方法。

已知样本 $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n$ 的距离矩阵 $\boldsymbol{\Delta} = (d_{ij})_{n\times n}$，譬如，欧氏距离矩阵。我们要寻找的数据表示 $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_n \in \mathbb{R}^k$ 要尽可能地保证

$$\|\boldsymbol{y}_i - \boldsymbol{y}_j\|_2 = d_{ij}, \forall i, j \in \{1, \cdots, n\}$$

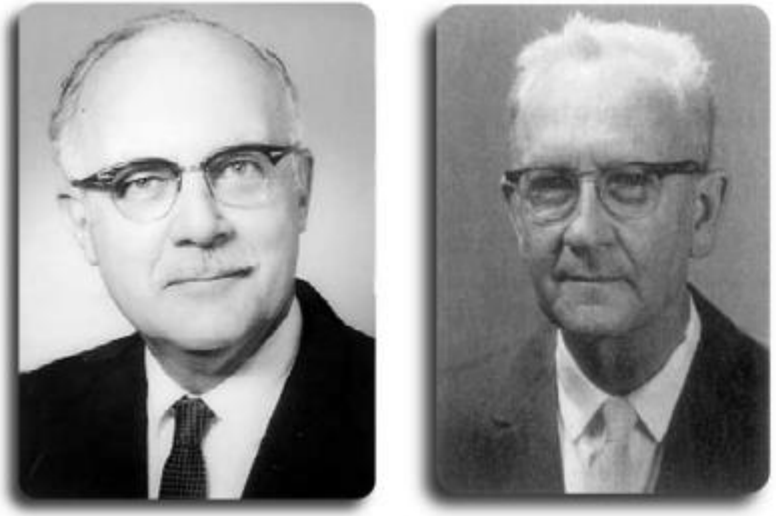
换句话说，就是求解下面的全局最优化问题。

$$\underset{y_1,\cdots,y_n \in \mathbb{R}^k}{\operatorname{argmin}} \sum_{i<j} (\|\boldsymbol{y}_i - \boldsymbol{y}_j\|_2 - d_{ij})^2$$

历史背景

实际上，该问题探讨的是如何仅仅通过给定的距离矩阵 Δ 重新表示样本，使其距离矩阵就是 Δ。重构样本的维数 k 是由 Δ 决定的。

1938 年，美国数学家盖尔·杨 (Gail Young, 1915—1999) 和阿尔斯通·斯科特·豪 斯霍尔德 (Alston Scott Householder, 1904—1993) 利用由欧氏距离矩阵 Δ 构造的内积矩阵的谱分解给出了一个答案。



杨和豪斯霍尔德

多维缩放（MDS）算法

给定欧氏距离矩阵 Δ，下述算法给出MDS问题 的一个解。

(1) 由 Δ 构造内积矩阵 G。

(2) 计算 G 的谱分解

$$\begin{aligned} \boldsymbol{G} &= \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^\top \\ &= \boldsymbol{V}\mathrm{diag}(\lambda_1,\cdots,\lambda_r)\boldsymbol{V}^\top \end{aligned}$$

(3) 我们得到MDS问题的一个解

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{V}^\top \\ &= \mathrm{diag}(\sqrt{\lambda_1},\cdots,\sqrt{\lambda_r})\boldsymbol{V}^\top \end{aligned}$$

显然，MDS问题的解并不唯一，以下也是它的一个解。

$$\boldsymbol{Y}_{n\times n} = \boldsymbol{V}\boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{V}^\top$$

MDS的例子

从普林斯顿大学 WordNet 词汇语义知识库或者类似的知识图谱中，可以抽取词汇距离（例如，两 个概念节点之间按照某种关系的最短路径的长度）。多维缩放算法使得这些词汇或概念在欧氏空间里得以向量化，为自然语言处理提供了数据表示。

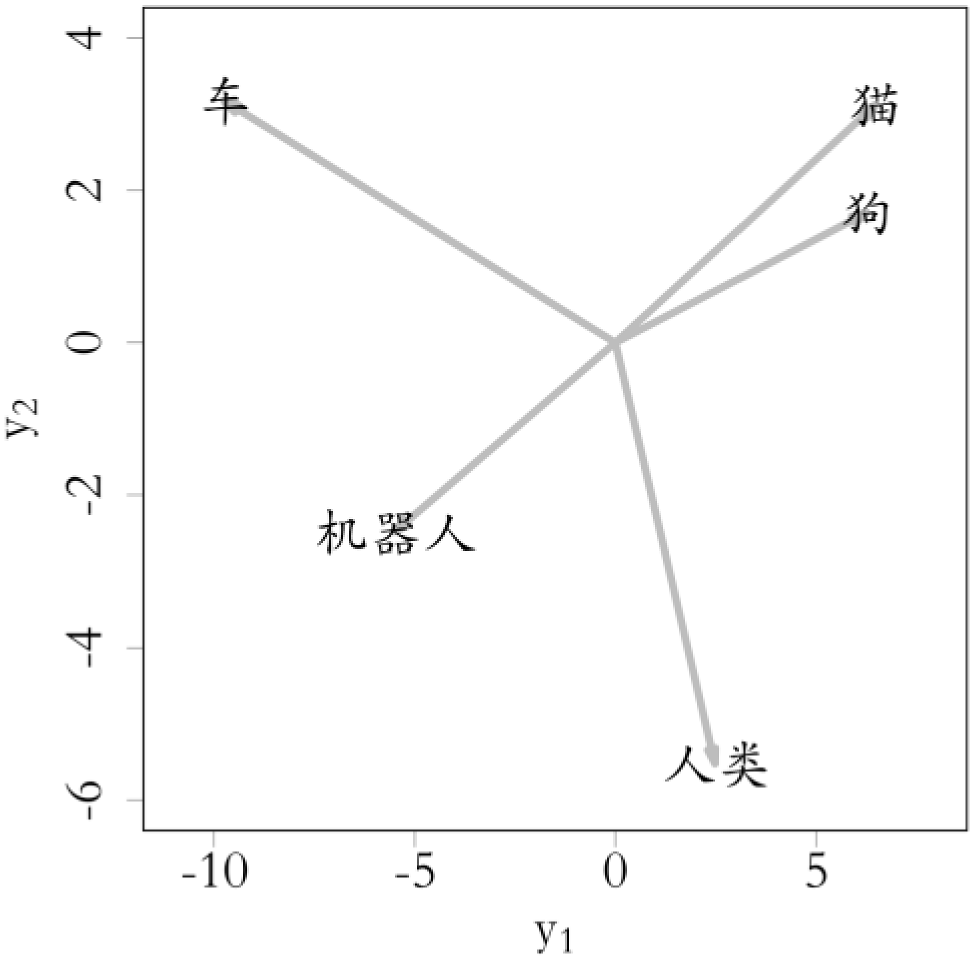
虚构的词汇距离数据

Δ	狗	猫	人类	机器人	车
狗	0	3	8	12	16
猫	3	0	9	13	16
人类	8	9	0	6	15
机器人	12	13	6	0	4
车	16	16	15	4	0

MDS的例子

二维词汇向量

	狗	猫	人类	机器人	车
y_1	6.26	6.49	2.49	−5.50	−9.74
y_2	1.70	3.13	−5.52	−2.48	3.17



在平面内的二维词汇向量

以上的讨论都是基于 Δ 是欧氏距离矩阵。在多维缩放中，采用欧氏距离的作法是值得商榷的。譬如，数据若分布在一个球面上，用欧氏距离来实现多维缩放就是错误的。一般地，数据在空间的分布是未知的，甚至连数据所在的空间我们也是知之甚少。若要考虑比球面更一般的几何对象，就需要流形 (manifold) 的概念。