# MDS降维原理

【参考资料】

周志华 《机器学习》

MDS全称为多维缩放（Multiple Dimensional Scaling），是一种经典的线性降维技术。其基本思想是**使原始空间中样本点之间的距离在低维空间中得以保持。**

假设$m$个$d$维的样本在原始空间的距离矩阵为$\mathbf{D} \in \mathbb{R}^{m \times m}$，其第$i$行第$j$列的元素$dist_{ij}$为样本$x_i$到样本$x_j$的距离。经过降维后，样本在$d'$维空间的表示$\mathbf{Z} \in \mathbb{R}^{d' \times m}$，并且$d' \leqslant d$。根据MDS的思想，任意两个样本在$d'$维空间的欧式距离要等于原始空间中的距离，即$\|z_i - z_j\| = dist_{ij}$。

令$\mathbf{B} = \mathbf{Z}^{\mathrm{T}}\mathbf{Z} \in \mathbb{R}^{m \times m}$，其中$\mathbf{B}$为降维后样本的内积矩阵，$b_{ij} = \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{z}_j$，有

$$
\begin{aligned}
\operatorname{dist}_{ij}^2 &= \|z_i\|^2 + \|z_j\|^2 - 2z_i^{\mathrm{T}}z_j \\
&= b_{ii} + b_{jj} - 2b_{ij}
\end{aligned}
$$

为便于讨论，令降维后的样本$\mathbf{Z}$被中心化，即$\sum_{i=1}^m z_i = 0$。显然，矩阵$\mathbf{B}$的行与列之和均为零，即$\sum_{i=1}^m b_{ij} = \sum_{j=1}^m b_{ij} = 0$。易知

$$
\begin{aligned}
\textstyle\sum_{i=1}^m \operatorname{dist}_{ij}^2 &= \operatorname{tr}(\mathbf{B}) + m b_{jj} \\
\textstyle\sum_{j=1}^m \operatorname{dist}_{ij}^2 &= \operatorname{tr}(\mathbf{B}) + m b_{ii} \\
\textstyle\sum_{i=1}^m \sum_{j=1}^m \operatorname{dist}_{ij}^2 &= 2m \operatorname{tr}(\mathbf{B})
\end{aligned}
$$

其中$\operatorname{tr}(\mathbf{B}) = \sum_{i=1}^m \|\boldsymbol{z}_i\|^2$。令

$$
\begin{aligned}
\operatorname{dist}_{i.}^2 &= \frac{1}{m}\sum_{j=1}^m dist_{ij}^2 \\
\operatorname{dist}_{.j}^2 &= \frac{1}{m}\sum_{i=1}^m dist_{ij}^2 \\
dist_{..}^2 &= \frac{1}{m^2}\sum_{i=1}^m\sum_{j=1}^m dist_{ij}^2
\end{aligned}
$$

可得

$$
b_{ij} = -\frac{1}{2}\left(dist_{ij}^2 - dist_{i.}^2 - dist_{;j}^2 + dist_{..)}^2\right.
$$

由此，即可通过降维前后保持不变的距离矩阵$\mathbf{D}$求取内积矩阵$\mathbf{B}$。

对矩阵$\mathbf{B}$做特征值分解，$\mathbf{B} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathbf{T}}$，其中$\mathbf{\Lambda} = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d)$为特征值构成的对角矩阵，$\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_d$，$\mathbf{V}$为特征向量矩阵。假定其中有$d^*$个非零特征值，它们构成对角矩阵$\mathbf{\Lambda}_* = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{d^*})$，令$\mathbf{V}_*$表示相应的特征向量矩阵，则$\mathbf{Z}$可表达为

$$\mathbf{Z} = \mathbf{\Lambda}_*^{1/2} \mathbf{V}_*^{\mathrm{T}} \in \mathbb{R}^{d^* \times m}$$

在现实应用中为了有效降维，往往仅需降维后的距离与原始空间中的距离尽可能接近，而不必严格相等。此时，可取$d' \ll d$个最大特征值构成对角矩阵$\tilde{\mathbf{\Lambda}} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{d'})$，令$\tilde{\mathbf{V}}$表示相应的特征向量矩阵，则$\mathbf{Z}$可表达为

$$\mathbf{Z} = \tilde{\mathbf{\Lambda}}^{1/2} \tilde{\mathbf{V}}^{\mathrm{T}} \in \mathbb{R}^{d' \times m}$$

MDS算法的完整描述如下：

**输入:** 距离矩阵 $\mathbf{D} \in \mathbb{R}^{m \times m}$, 其元素 $dist_{ij}$ 为样本 $\boldsymbol{x}_i$ 到 $\boldsymbol{x}_j$ 的距离;
低维空间维数 $d'$.

**过程:**

1: 根据式(10.7)~(10.9)计算 $dist_{i\cdot}^2$, $dist_{\cdot j}^2$, $dist_{\cdot\cdot}^2$;

2: 根据式(10.10)计算矩阵 $\mathbf{B}$;

3: 对矩阵 $\mathbf{B}$ 做特征值分解;

4: 取 $\tilde{\mathbf{\Lambda}}$ 为 $d'$ 个最大特征值所构成的对角矩阵, $\tilde{\mathbf{V}}$ 为相应的特征向量矩阵.

**输出:** 矩阵 $\tilde{\mathbf{V}}\tilde{\mathbf{\Lambda}}^{1/2} \in \mathbb{R}^{m \times d'}$, 每行是一个样本的低维坐标

MDS和PCA非常相似，都是要计算数据样本点在某个空间中的表示，差别仅在于PCA的输入是样本点在原空间的坐标，而MDS则需要输入各个点的距离度量，事实上已经有证明如果MDS输入的是样本点在原空间的欧式距离，则与PCA等价。