一、从k近邻到降维

给定测试样本 $\boldsymbol{x}$ ，若其最近邻样本为 $\boldsymbol{z}$,则最近邻分类器出错的概率就是 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 类别标记不同的概率，

$$P(err) = 1 - \sum_{c \in \mathcal{Y}} P(c \mid \boldsymbol{x}) P(c \mid \boldsymbol{z}).$$

$$\begin{aligned} P(err) &= 1 - \sum_{c \in \mathcal{Y}} P(c \mid \boldsymbol{x}) P(c \mid \boldsymbol{z}) \\ &\simeq 1 - \sum_{c \in \mathcal{Y}} P^2(c \mid \boldsymbol{x}) \\ &\leq 1 - P^2(c^* \mid \boldsymbol{x}) \\ &= \left(1 + P\left(c^* \mid \boldsymbol{x}\right)\right)\left(1 - P\left(c^*\right.\right. \\ &\leq 2 \times \left(1 - P\left(c^* \mid \boldsymbol{x}\right)\right). \end{aligned}$$

最近邻分离器的泛化错误率不会超过贝叶斯最优分类器错误率的两倍！

但是在真实的应用中，我们是否能够准确的找到*k*近邻呢？

维数过大时近邻太多--》降维

Q：为什么能进行降维?

数据样本虽是高维的，但与学习任务密切相关的也许仅是某个低维分布，即高维空间中的一个低维“嵌入” (embedding)

MDS

(Multiple Dimensional Scaling) 旨在寻找一个低维子空间，

样本在此子空间内的距离和样本原有距离尽量保持不变

二、PCA

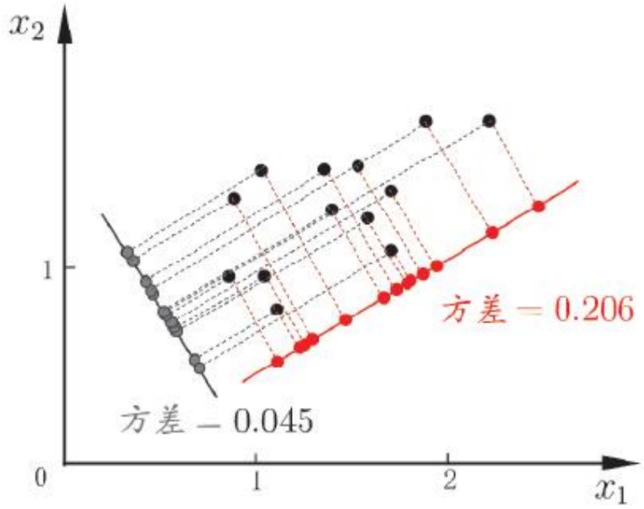
> PCA是无监督方法：只有x没有y，没有标记，没有类别信息先验

搞清楚目的：不同后续任务导致不同降维方法

把高维空间中的低维结构恢复出来

1、两种基础思路

- 最大可分性：样本点在超平面上的投影尽可能分开
- 最近重构性：样本点到超平面距离都足够近



优化目标一致：

$$\min_{\mathbf{W}} \quad -\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)$$
$$\text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}.$$

- 求tr：把矩阵变成标量可优化
- 给W正交的限制，避免无穷大

对式(10.15)或(10.16)使用拉格朗日乘子法可得

$$\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} = \lambda\mathbf{W}\,, \tag{10.17}$$

于是，只需对协方差矩阵 $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ 进行特征值分解，将求得的特征值排序：$\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_d$，再取前 d' 个特征值对应的特征向量构成 $\mathbf{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'})$. 这就是主成分分析的解. PCA 算法描述如图 10.5 所示.

最大可分性

> 投影方差最大化

$$\max_{\mathbf{W}} \quad \operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)$$
$$\text{s.t.} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}\,,$$

最近重构性

> 丢掉维度后变化最小

假定投影变换后得到的新坐标系为$\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_d\}$， 其中 $\boldsymbol{w}_i$是标准正交基向量

$$||\boldsymbol{w}_i||_2 = 1, \boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{w}_j = 0 (i \neq j).$$

若丢弃新坐标系中的部分坐标，即将维度降低到 $d' < d$，则样本点在低维坐标系中的投影是 $\boldsymbol{z}_i = (z_{i1}; z_{i2}; \ldots; z_{id'})$ $z_{ij} = \boldsymbol{w}_j^{\mathrm{T}} \boldsymbol{x}_i$

若基于 $\boldsymbol{z}_i$来重构$\boldsymbol{x}_i$，则会得到 $\hat{\boldsymbol{x}}_i = \sum_{j=1}^{d'} z_{ij} \boldsymbol{w}_j.$

- 重构误差用平方损失

考虑整个训练集, 原样本点 $\boldsymbol{x}_i$ 与基于投影重构的样本点 $\hat{\boldsymbol{x}}_i$ 之间的距离为

$$\begin{aligned}\sum_{i=1}^{m} \left\| \sum_{j=1}^{d'} z_{ij} \boldsymbol{w}_j - \boldsymbol{x}_i \right\|_2^2 &= \sum_{i=1}^{m} \boldsymbol{z}_i^{\mathrm{T}} \boldsymbol{z}_i - 2 \sum_{i=1}^{m} \boldsymbol{z}_i^{\mathrm{T}} \mathbf{W}^{\mathrm{T}} \boldsymbol{x}_i + \text{const} \\ &\propto -\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}} \left(\sum_{i=1}^{m} \boldsymbol{x}_i \boldsymbol{x}_i^{\mathrm{T}} \right) \mathbf{W} \right). \end{aligned} \tag{10.14}$$

$\boldsymbol{w}_j$ 是正交基, $\sum_i \boldsymbol{x}_i \boldsymbol{x}_i^{\mathrm{T}}$ 是协方差矩阵

> 方差构成了对角线上的元素，协方差构成了非对角线上的元素

$$\begin{aligned} \min_{\mathbf{W}} \quad & -\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}} \mathbf{X} \mathbf{X}^{\mathrm{T}} \mathbf{W}\right) \\ \text{s.t.} \quad & \mathbf{W}^{\mathrm{T}} \mathbf{W} = \mathbf{I}. \end{aligned}$$

2、计算方法

基于特征值分解协方差矩阵

输入: 样本集 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
低维空间维数 d'.

过程:

1: 对所有样本进行中心化: $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i - \frac{1}{m} \sum_{i=1}^{m} \boldsymbol{x}_i$;
2: 计算样本的协方差矩阵 $\mathbf{X}\mathbf{X}^{\mathrm{T}}$;
3: 对协方差矩阵 $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ 做特征值分解;
4: 取最大的 d' 个特征值所对应的特征向量 $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'}$.

输出: 投影矩阵 $\mathbf{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{d'})$.

图 10.5 PCA 算法

1、先中心化：样本均值为0，每一位特征减去各自的平均值

> 算方差时不用减去均值

> 对验证集、测试集执行零均值化操作时，均值必须从训练集计算而来

2、从原数据集X直接得到协方差矩阵（均值已消去）

假设我们只有 a 和 b 两个变量，那么我们将它们按行组成矩阵 X：

$$X = \begin{pmatrix} a_1 & a_2 & \cdots & a_m \\ b_1 & b_2 & \cdots & b_m \end{pmatrix}$$

然后：

$$\frac{1}{m}XX^{\mathsf{T}} = \begin{pmatrix} \frac{1}{m}\sum_{i=1}^m a_i^2 & \frac{1}{m}\sum_{i=1}^m a_ib_i \\ \frac{1}{m}\sum_{i=1}^m a_ib_i & \frac{1}{m}\sum_{i=1}^m b_i^2 \end{pmatrix} = \begin{pmatrix} Cov(a,a) & Cov(a,b) \\ Cov(b,a) & Cov(b,b) \end{pmatrix}$$

我们可以看到这个矩阵对角线上的分别是两个变量的方差，而其它元素是 a 和 b 的协方差。两者被统一到了一个矩阵里。

> 这里除或不除样本数量n或n-1,其实对求出的特征向量没有影响

3、特征值分解求协方差矩阵的特征值和特征向量

4、求解W：选最大的d'个特征值对应的特征向量，作为行向量组成W

> 工具包指令了前d'可有快速方法

5、Y=WX得到降维后结果

逐一选取最大方差方向

PCA 也可看作是逐一选取方差最大方向，即先对协方差矩阵 $\sum_i \boldsymbol{x}_i\boldsymbol{x}_i^{\mathrm{T}}$ 做特征值分解，取最大特征值对应的特征向量 $\boldsymbol{w}_1$；再对 $\sum_i \boldsymbol{x}_i\boldsymbol{x}_i^{\mathrm{T}} - \lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^{\mathrm{T}}$ 做特征值分解，取最大特征值对应的特征向量 $\boldsymbol{w}_2$；……由 $\mathbf{W}$ 各分量正交及

$$\sum_{i=1}^m \boldsymbol{x}_i\boldsymbol{x}_i^{\mathrm{T}} = \sum_{j=1}^d \lambda_j\boldsymbol{w}_j\boldsymbol{w}_j^{\mathrm{T}}$$

可知，上述逐一选取方差最大方向的做法与直接选取最大 d' 个特征值等价.

基于SVD分解协方差矩阵

> 特征值和特征向量是针对方阵才有的，任意形状矩阵都可以做奇异值分解

数据标准化

> 协方差矩阵易受特征尺度影响（取值范围）

超参数d'确定

- 用户指定
- 交叉验证确定，重构误差可能是U型曲线
- 设置重构阈值：占用95%的特征值大小，选取下式成立最小的d'

$$\frac{\sum_{i=1}^{d'} \lambda_i}{\sum_{i=1}^{d} \lambda_i} \geqslant t \ .$$

3、例题&分析

算法示例

$$X=\begin{pmatrix}-1 & -1 & 0 & 2 & 0\\ -2 & 0 & 0 & 1 & 1\end{pmatrix}$$

以X为例，我们用PCA方法将这两行数据降到一行。

1)因为X矩阵的每行已经是零均值，所以不需要去平均值。

2)求协方差矩阵：

$$C=\frac{1}{5}\begin{pmatrix}-1 & -1 & 0 & 2 & 0\\ -2 & 0 & 0 & 1 & 1\end{pmatrix}\begin{pmatrix}-1 & -2\\ -1 & 0\\ 0 & 0\\ 2 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}\frac{6}{5} & \frac{4}{5}\\ \frac{4}{5} & \frac{6}{5}\end{pmatrix}$$

3)求协方差矩阵的特征值与特征向量。

求解后的特征值为：

$$\lambda_1=2，\lambda_2=\frac{2}{5}$$

对应的特征向量为：

$$c_1\begin{pmatrix}1\\ 1\end{pmatrix}，c_2\begin{pmatrix}-1\\ 1\end{pmatrix}$$

其中对应的特征向量分别是一个通解， c_1 和 c_2 可以取任意实数。那么标准化后的特征向量为：

$$\begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix}，\begin{pmatrix}-\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix}$$

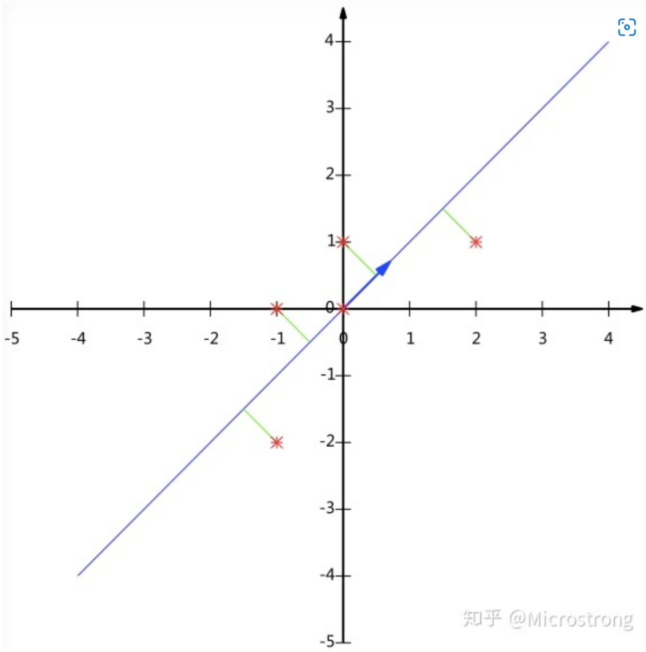
4)矩阵P为：

$$P=\begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\end{pmatrix}$$

5)最后我们用P的第一行乘以数据矩阵X，就得到了降维后的表示：

$$Y=\begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}-1 & -1 & 0 & 2 & 0\\ -2 & 0 & 0 & 1 & 1\end{pmatrix}=\begin{pmatrix}-\frac{3}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & \frac{3}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{pmatrix}$$

- 示意图：



数据矩阵X降维投影结果

注意：如果我们通过特征值分解协方差矩阵，那么我们只能得到一个方向的PCA降维。这个方向就是对数据矩阵X从行(或列)方向上压缩降维。

有关协方差

设原始数据矩阵 X 对应的协方差矩阵为 C，而 P 是一组基按行组成的矩阵，设 Y=PX，则 Y 为 X 对 P 做基变换后的数据。设 Y 的协方差矩阵为 D，我们推导一下 D 与 C 的关系：

$$
\begin{aligned}
D &= \frac{1}{m}YY^T \\
&= \frac{1}{m}(PX)(PX)^T \\
&= \frac{1}{m}PXX^TP^T \\
&= P(\frac{1}{m}XX^T)P^T \\
&= PCP^T
\end{aligned}
$$

这样我们就看清楚了，我们要找的 P 是能让原始协方差矩阵对角化的 P。换句话说，优化目标变成了**寻找一个矩阵 P，满足 PCP^T 是一个对角矩阵，并且对角元素按从大到小依次排列，那么 P 的前 K 行就是要寻找的基，用 P 的前 K 行组成的矩阵乘以 X 就使得 X 从 N 维降到了 K 维并满足上述优化条件。**

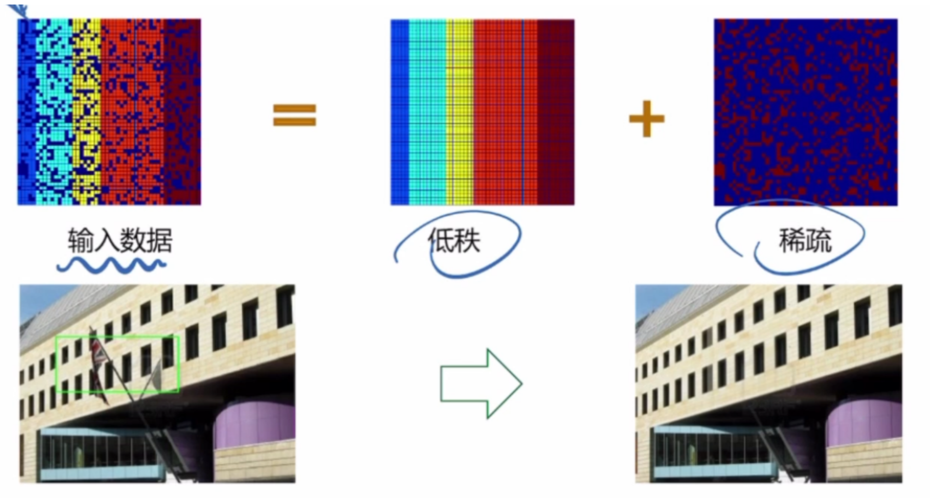
4、PCA应用

- 人脸识别中的特征脸，是降维后的特征向量
- 识别图片：使用均值和几个特征的线性组合即可表示所有可能的

Robust PCA：

不但要找到低秩重构，还要重构误差是稀疏的（少量非0）

用于图片去噪，人脸去墨镜



函数推广

找到函数空间的基

傅里叶变换：轻量级处理大规模核函数

5、非线性降维

线性降维方法假设从高维空间到低维空间的函数映射是线性的

然而在许多现实任务中，可能需要非线性映射才能找到恰当的低维嵌入

非线性降维的常用方法：

核化线性降维：如KPCA，KLDA，...

流形学习 (manifold learning)

三、度量学习

给度量函数加参数，学习度量矩阵M

> 给平方项和交叉项都加参数，就相当于乘上一个半正定矩阵M

马氏距离：

$$\mathrm{dist}^2_{\mathrm{mah}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}\mathbf{M}(\boldsymbol{x}_i - \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2_{\mathbf{M}}$$

Q：为什么度量矩阵M必须半正定？

- 为了保持距离非负且对称
- 也就是必须有正交基P使M=PP^T(ppt笑死)

可以把错误率这样的监督学习目标作为度量学习的优化目标

还可以引入邻域先验知识，相似样本加入必连集合，不相似加入勿连集合

距离度量学习 – NCA：Neighborhood Component Analysis

近邻分类器在进行判别时通常使用多数投票法，替换为概率投票法. 对于任意样本x_j，它对x_i分类结果影响的概率为

$$p_{ij} = \frac{\exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_{\mathbf{M}}^2\right)}{\sum_l \exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_l\|_{\mathbf{M}}^2\right)} ,$$

x_i样本的LOO正确率：

$$p_i = \sum_{j \in \Omega_i} p_{ij} ,$$

训练集上的LOO正确率：

$$\sum_{i=1}^{m} p_i = \sum_{i=1}^{m} \sum_{j \in \Omega_i} p_{ij} .$$

NCA的优化目标：

$$\min_{\mathbf{P}} \ 1 - \sum_{i=1}^{m} \sum_{j \in \Omega_i} \frac{\exp\left(-\left\|\mathbf{P}^{\mathrm{T}}\boldsymbol{x}_i - \mathbf{P}^{\mathrm{T}}\boldsymbol{x}_j\right\|_2^2\right)}{\sum_l \exp\left(-\|\mathbf{P}^{\mathrm{T}}\boldsymbol{x}_i - \mathbf{P}^{\mathrm{T}}\boldsymbol{x}_l\|_2^2\right)} .$$