

PCA 主成分分析

2018.10.02 Louis

PCA 主成分分析

1. 西瓜书的 PCA 解释

1.1 最近重构性

1.2 最大可分性

1.3 算法

2. 其他解释

2.1 推导

2.2 疑问展开

2.3 numpy 代码

2.4 PCA 和 SVD 的关系

2.4.1 结论二 的推导

2.4.2 结论一 的推导

2.5 其他

在高维特征空间中，数据样本稀疏，且大多数点都分布在边界处、实例彼此远离，预测的可靠性更低。训练集维度越高，过拟合风险越大。

降维的方法主要有“投影”和“流形学习”。

PCA 是一种投影的降维方法。

1. 西瓜书的 PCA 解释

Principal Component Analysis 主成分分析是一种常见的降维方法。首先它找到接近数据集分布的超平面，然后将所有的数据都投影到这个超平面上。

这个超平面要具备怎样的性质？

- 最近重构性：样本点到超平面距离足够近
- 最大可分性：样本点到超平面的投影尽可能分开

实际上，超平面的这两种性质是等价的。

[这张图能够直观地解释](#)

1.1 最近重构性

PCA 假定数据集以原点为中心，首先样本数据要进行中心化，即 $\sum_i \mathbf{x}_i = \mathbf{0}$ ，投影变换后得到的坐标系为 $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_d\}$ ，每个 \mathbf{w}_i 都是标准正交基向量， $\|\mathbf{w}_i\|_2 = 1, \mathbf{w}_i^T \mathbf{w}_j = 0 (i \neq j)$ 。

再丢弃新坐标系中的部分坐标，维度从 d 维降低为 d' 维， $d' < d$ ，则原来样本点 x_i 在 d' 维坐标系中的投影是 $z_i = (z_{i1}; z_{i2}; \dots; z_{id'})$ ，其中 $z_{ij} = \mathbf{w}_j^T \mathbf{x}_i$ 是 x_i 在低维坐标系下第 j 维的坐标。

若基于 z_i (即投影后的样本点坐标)，来重新构造出原来坐标系下的样本点 x_i ，则会得到 $\hat{x}_i = \sum_{j=1}^{d'} z_{ij} \mathbf{w}_j$ 。

再来考虑整个训练集中，原样本点 x_i 与基于投影重构的样本点 \hat{x}_i 直接的距离：

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

我还没想明白为什么正比于那个迹

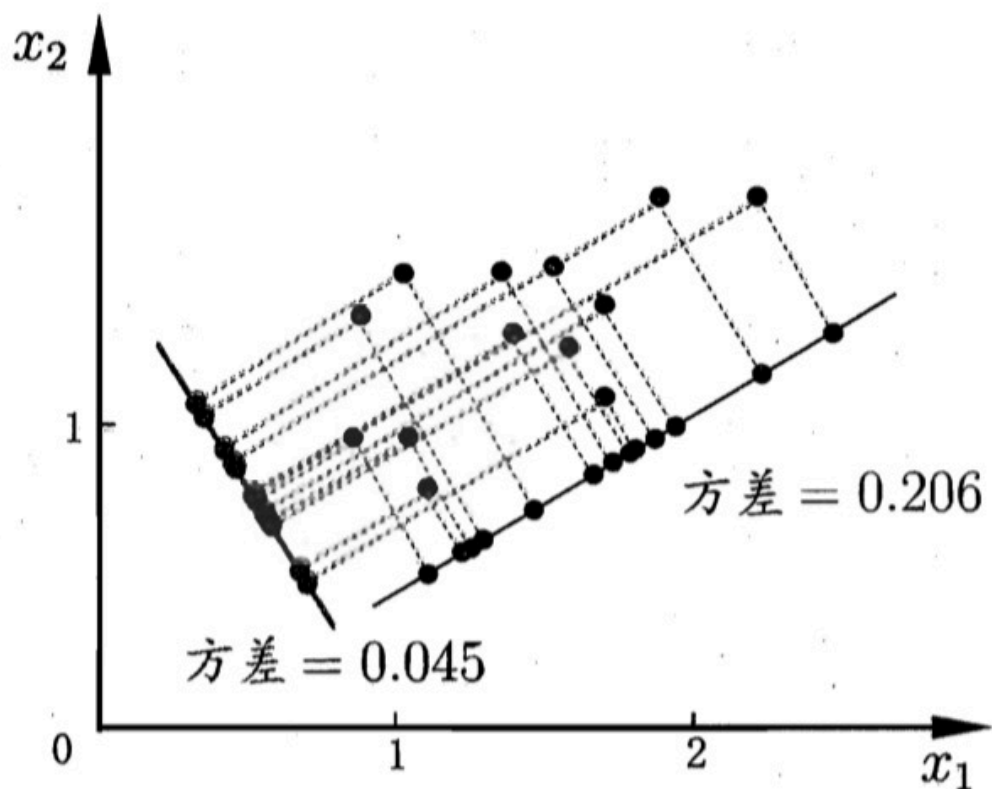
这样，PCA 的优化目标为

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

\mathbf{w}_j 是标准正交基， $\sum_i \mathbf{x}_i \mathbf{x}_i^T$ 是协方差矩阵（实际上乘 $1/(m-1)$ 才是，但常数项不影响）

1.2 最大可分性

样本点 x_i 在新空间中超平面上的投影是 $\mathbf{W}^T \mathbf{x}_i$ ，若所有样本点投影尽可能分开，则应该使得投影后样本点的方差最大化：如图所示



投影后样本点的方差是 $\sum_i \mathbf{W}^T \mathbf{x}_i \mathbf{x}_i^T \mathbf{W}$ ，于是优化目标写为：

$$\begin{aligned} \max_{\mathbf{W}} \operatorname{tr}(\mathbf{W}^T \mathbf{X} \mathbf{X}^T \mathbf{W}) \\ \text{s.t. } \mathbf{W}^T \mathbf{W} = \mathbf{I} \end{aligned}$$

显然，这与最近重构性的优化目标等价。

对上面两式使用拉格朗日乘子法得到

$$\mathbf{X} \mathbf{X}^T \mathbf{W} = \lambda \mathbf{W}$$

只需要对协方差矩阵 $\mathbf{X} \mathbf{X}^T$ 进行特征值分解，将求得的特征值排序： $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$ ，再取前 d' 个特征值对应的特征向量构成 $\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{d'})$ ，这就是主成分分析的解。

1.3 算法

输入: 样本集 $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$;

过程:

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

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

实践中，通常对 X 进行奇异值分解来代替协方差矩阵的特征值分解。

降维后低维空间的维数 d' 通常由用户事先指定，或通过在不同 d' 值的低维空间中对 k 近邻分类器（或其他开销较小的学习器）进行交叉验证来选取较好的 d' 值，对 PCA，还可以从重构的角度设置一个重构阈值，例如 $t = 95\%$ ，然后选取使下式成立的最小 d' 值。

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

降维舍弃部分信息是必要的：

1. 舍弃部分信息后能使采样密度增大，这是降维的重要动机
2. 最小特征值所对应的特征向量往往与噪声有关，舍弃它们一定程度上有去噪的效果

2. 其他解释

2.1 推导

西瓜书上的部分推导较简略，还看到另一个比较详细的推导。

现在只考虑最小化投影造成的损失，即最近重构性。

现在标准正交基设为 $\{u_j\}, (j = 1, \dots, n)$ ，我们要减掉其中一些维度，使得误差足够小。

对 x_i 在方向 u_j 上的投影是 $(x_i^T u_j)u_j$

如果减掉 u_j 这个维度，造成的误差为：

所有样本在 u_j 维度上的投影的平方和取均值。

$$\begin{aligned} J_j &= \frac{1}{m} \sum_{i=1}^m (x_i^T u_j)^2 \\ &= \frac{1}{m} \|x^T u_j\|_2^2 \\ &= \frac{1}{m} (x^T u_j)^T (x^T u_j) \\ &= \frac{1}{m} u_j^T x x^T u_j \end{aligned}$$

注意如何将 $m \times 1$ 维度的 $L2$ 范式转换为 $v^T v$ 的技巧，其中 v^T 维度为 $1 \times m$ ，而 v 维度为 $m \times 1$ ，其实就是每个元素的平方和。

将 $\frac{1}{m} x x^T$ 记作 S ，接下来要考虑我们要减去哪 t 个维度，使得损失最小化：

$$\begin{aligned} J &= \sum_{j=n-t}^n u_j^T S u_j \\ s.t. \quad &u_j^T u_j = 1 \end{aligned}$$

此时使用拉格朗日乘子法使得：

$$\tilde{J} = \sum_{j=n-t}^n u_j^T S u_j + \lambda_j (1 - u_j^T u_j)$$

最小化上式子，求导有：

$$\frac{\delta \tilde{J}}{\delta u_j} = S u_j - \lambda_j u_j$$

上面是标量对矢量求导，注意求导后的维度与被求导的维度(u_j)要是一致的。

使其为 0 则得到：

$$S u_j = \lambda_j u_j$$

这是标准的特征值的定义， λ_j 是特征值， u_j 是对应的特征向量，所以对 S 进行特征值分解即可得到解，将上式代入原始的 J 中，得到：

$$\begin{aligned}
 J &= \sum_{j=n-t}^n u_j^T S u_j \\
 &= \sum_{j=n-t}^n u_j^T \lambda_j u_j \\
 &= \sum_{j=n-t}^n \lambda_j
 \end{aligned}$$

所以要使 J 最小，就去掉变换后维度中最小的 t 个特征值对应的维度就好了。

2.2 疑问展开

一、为什么 $\frac{1}{m-1}xx^T$ 就是协方差矩阵呢？

对于两个变量 X, Y ，其协方差 $Cov(X, Y)$ 为

$$Cov(X, Y) = E[X - E(X)][Y - E(Y)]$$

现在是对向量而言，协方差矩阵的计算公式为：

$$\Sigma = E[(\mathbf{x} - E[\mathbf{x}])(\mathbf{x} - E[\mathbf{x}])^T]$$

由于我们第一步进行了去中心化 $x - E[x]$ ，所以 $s = \frac{1}{m}xx^T$ 其实就是协方差矩阵（虽然标准的协方差矩阵上应该是 $1/(m-1)$ ，但系数对特征值、特征向量无影响）

2.3 numpy 代码

下面代码中的矩阵 X 其实是上面推导中的 X^T ，每一行是一个样本

```
def pca(X):
    # Step1: 去中心化
    mean_ = np.mean(X, axis=0)
    X = X - mean_

    # Step2: 得到协方差矩阵
    covMat = np.cov(X, rowvar = 0) # rowvar=False 每列代表一个变量
    # 实际上是否去中心化对于得到协方差矩阵无影响， 只是为了方便后续降维

    # Step3: 得到特征值和特征向量 eigenvalue, eigenvector
    eigVal, eigVec = sp.linalg.eig(covMat)

# 不用 cov，只用矩阵乘法
def pca_dot(X):
    mean_ = np.mean(X, axis=0)
    X = X - mean_
    M, N = X.shape
    Sigma = np.dot(X.transpose(), X) / (M - 1)
    eigVal, eigVec = sp.linalg.eig(Sigma)
```

2.4 PCA 和 SVD 的关系

简而言之，就是 **SVD** 奇(qí)异值分解，在 PCA 的应用中常用来代替特征值分解。

用特征值分解，矩阵中一些较小的数容易在平方中丢失。而 **SVD** 分解不直接计算 $X^T X$ ，所以不会丢失较小的数，而且速度比特征值分解快很多，充分利用了协方差矩阵的性质。

[奇异值分解 \(Wikipedia\)](#)

$$X = U\Sigma V^*$$

U 是 $m \times m$ 阶 **酉矩阵**

Σ (sigma) 是 $m \times n$ 阶非负实数对角矩阵

V^* ，即 V 的共轭转置，是 $n \times n$ 阶 **酉矩阵**

这样的分解就称作 X 的**奇异值分解**

1. U 的列组成一套对 X 的正交“输出”的基向量，这些向量是 XX^T 的特征向量。
2. Σ 对角线上的元素是**奇异值**，可视为是在输入与输出间进行的标量的“膨胀控制”。这些是 $X^T X$ 和 XX^T 特征值的非零平方根，并与 U 和 V 的行向量相对应。
3. V 的列组成一套对 X 的正交“输入”或“分析”的基向量。这些向量是 $X^T X$ 的特征向量。

因此 **SVD** 的结果得到的特征向量，可以直接用于 PCA 降维。

```
# 求出协方差矩阵
def pca_svd_cov(X):
    mean_ = np.mean(X, axis=0)
    X = X - mean_
    M,N=X.shape
    Sigma=np.dot(X.transpose(),X) #这里直接去掉/(M-1)方便和pca_svd比较，对求得特
征向量无影响
    U,S,V = sp.linalg.svd(Sigma); # 把 eig 改成 svd
    eigVal,eigVec = S,U
```

结论一：协方差矩阵（或 $X^T X$ ）的奇异值分解结果和特征值分解结果一致

```
# 不求协方差矩阵，通过 SVD 也可以直接得出  $X^T X$  的特征向量
def pca_svd(X):
    mean_ = np.mean(X, axis=0)
    X = X - mean_
    U, S, V = sp.linalg.svd(X)
    # S 对角线就是特征值非零平方根，V 列向量就是特征值
    eigVal, eigVec = S, V
```

结论二：V 的列组成一套对 X 的正交“输入”或“分析”的基向量。这些向量是 $X^T X$ 的特征向量。

2.4.1 结论二的推导

根据奇异值分解的定义：

$$X = U\Sigma V^T$$

则有

$$\begin{aligned} X^T X &= V\Sigma U^T U\Sigma V^T \\ &= V\Sigma^2 V^T \\ &= V\Sigma^2 V^{-1} \end{aligned}$$

Σ 是对角矩阵，U 是标准正交基（酉矩阵），V 是标准正交基（ $VV^T = I; V = V^{-1}$ ）

$X^T X$ 是一个对称的半正定矩阵，它可以通过特征值分解为

$$X^T X = Q\Lambda Q^{-1}$$

（ Λ 是对角化特征值，Q 是特征向量）

可以看到 $X^T X = V\Sigma^2 V^{-1}$ 和 $X^T X = Q\Lambda Q^{-1}$ 形式一致，当限定了特征值顺序后，这样的组合是唯一的，所以结论二成立：V 是 $X^T X$ 的特征向量，奇异值和特征值是平方关系：

$$\begin{aligned} V &= Q \\ \Lambda &= \Sigma^2 \end{aligned}$$

所以 `u, s, v` 得到的奇异值 `s` 的平方才是特征值 `eigval`，可以通过运行代码得到验证

2.4.2 结论一的推导

结论一：协方差矩阵（或 $X^T X$ ）的奇异值分解结果和特征值分解结果一致

我们对 $X^T X$ 进行 SVD 分解，为了区分，U 取下标 2

$$X^T X = U_2 \Sigma_2 V_2^T$$

注意是：

$$X = U\Sigma V^T$$

SVD 分解性质的第二条：

U 的列组成一套对 X 的正交“输出”的基向量，这些向量是 XX^T 的特征向量

注意这里 X 是 $X^T X$ ，所以 U_2 的列是矩阵 $X^T X X^T X = (X^T X) * (X^T X)^T$ 的特征向量

$$\begin{aligned} X^T X X^T X &= U_2 \Sigma_2 V_2^T (U_2 \Sigma_2 V_2^T)^T \\ &= U_2 \Sigma_2^2 U_2^T \end{aligned}$$

$$\begin{aligned} X^T X &= Q_2 \Lambda_2 Q_2^{-1} \\ X^T X X^T X &= Q_2 \Lambda_2^2 Q_2^{-1} \end{aligned}$$

所以有：

$$\begin{aligned} U_2 &= Q_2 \\ \Sigma^2 &= \Lambda^2 \end{aligned}$$

能得到这样的结果是因为 $X^T X$ 本身是对称的半正定矩阵。

2.5 其他

SVD 还常用来计算伪逆，这在最小二乘中有应用：

$$X^{-1} = V \Sigma^{-1} U^T$$