

## Lecture 6: 数据降维之主成分分析

2024.3.12

Lecturer: 丁虎

Scribe: 王运韬

对于高维数据集，数据经常是非常稀疏的，一个合理的操作是将这些高维数据降为低维数据。

从几何的角度来说，对于集合  $P = \{p_1, \dots, p_n\} \subset R^d$ ，我们期望找到一个  $k$  维的子空间  $E$ ， $k \ll d$ ，最小化  $\sum_{i=1}^n \|p_i - \pi(p_i)\|_2^2$ ，其中  $\pi$  是  $R^d$  到  $E$  的投影。当然，此处的目标函数也可以替换为其他的度量。

从线性代数的角度讲，对于矩阵  $A \in R^{n \times d}$  找到一个秩为  $k$  的矩阵  $A_k$  使得  $\|A - A_k\|_F$  最小。其中  $\|X\|_F = \sqrt{\sum_{i,j} X_{ij}^2}$ 。

当  $k = 0$ ，我们要寻找的是一个点，由定义可知就是  $P$  的重心  $\nu(P)$ 。当  $k > 0$ ， $\nu(P)$  应当还是在我们要寻找的子空间上：

$$\begin{aligned} \sum_{i=1}^n \|p_i - \mathbf{proj}_E(\nu(P))\|_2^2 &= \sum_{i=1}^n \|\mathbf{proj}_E(p_i) - \mathbf{proj}_E(\nu(P))\|_2^2 + \|\mathbf{proj}_E(p_i) - p_i\|_2^2 \\ &\leq \|\mathbf{proj}_E(p_i) - \mathbf{proj}_E(\nu(P))\|_2^2 + \|p_i - \mathbf{proj}_E(p_i)\|_2^2 + \|\nu(P) - \mathbf{proj}_E(\nu(P))\|_2^2 \\ &= \|p_i - \nu(P)\|_2^2. \end{aligned}$$

由重心的定义，可知不等号只能相等，即  $\nu(P) \in E$ 。

## 1 奇异值分解 (Singular Value Decomposition)

根据上边的表述，我们可以把 PCA 的核心思想表述为：能否找到更小的一组基底，使之能尽可能好地重新代表原数据？为了严格地解决这个问题，我们引入奇异值分解技术。 $X$  为  $n \times m$  矩阵， $X^T X$  秩为  $r$ ，接下来我们定义相关的量：

- $\{\hat{v}_1, \hat{v}_2, \dots, \hat{v}_r\}$  是  $X^T X$  一族正交的  $m \times 1$  特征向量，对应特征值  $\{\lambda_1, \lambda_2, \dots, \lambda_r\}$ 。即

$$(X^T X)\hat{v}_i = \lambda_i \hat{v}_i.$$

- $\sigma_i \equiv \sqrt{\lambda_i}$  正实数特征值。

- $\{\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \dots, \hat{\mathbf{u}}_r\}$  是一族  $n \times 1$  向量, 满足  $\hat{\mathbf{u}}_i \equiv \frac{1}{\sigma_i} X \hat{\mathbf{v}}_i$ .
- $\hat{\mathbf{u}}_i \cdot \hat{\mathbf{u}}_j = \begin{cases} 1 & \text{如果 } i = j \\ 0 & \text{otherwise} \end{cases}$
- $\|X \hat{\mathbf{v}}_i\| = \sigma_i$

我们的目标是构造如下的对角矩阵  $\Sigma$ .

$$\Sigma \equiv \begin{bmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_{\bar{r}} & & & \\ & & & 0 & & \\ & 0 & & & \ddots & \\ & & & & & 0 \end{bmatrix}$$

其中  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\bar{r}}$  是由大到小排列的奇异值. 我们需要构造正交矩阵

$$\begin{aligned} V &= [\hat{\mathbf{v}}_1 \ \hat{\mathbf{v}}_2 \ \dots \ \hat{\mathbf{v}}_m] \\ U &= [\hat{\mathbf{u}}_1 \ \hat{\mathbf{u}}_2 \ \dots \ \hat{\mathbf{u}}_n] \end{aligned}$$

此处增补了  $(m - r)$  和  $(n - r)$  个正交基来使  $V$  和  $U$  为方阵. 满足

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

根据需求, 可以发现如下奇异值分解矩阵  $X$  的算法:

1. 计算  $XX^T$  的特征值和特征向量, 用单位化的特征向量构成  $U$ .
2. 计算  $X^T X$  的特征值和特征向量, 用单位化的特征向量构成  $V$ .
3. 将  $XX^T$  的特征值开方, 得到  $\Sigma$ .

当然, 因为计算  $XX^T$  的开销可能很大, 这个算法不是实践中使用的算法. 此外, 在固定了  $\Sigma$  对角元的排序后, 奇异值分解是唯一的.

## 2 主成分分析

回顾我们的设定，原始数据可以写成  $X \in \mathbb{R}^{n \times d}$ ， $n$  是样本个数， $d$  是特征维数。假设我们之前假设的  $k$  维子空间  $E$  有一组标准正交基  $v_1, v_2, \dots, v_k$ ，并将它扩展到  $d$  维全空间，那么对于任意的向量  $a$ ，

$$\|a - \mathbf{proj}_E(a)\|^2 = \left\| \sum_{j=1}^d v_j \langle a, v_j \rangle - \sum_{j=1}^k v_j \langle a, v_j \rangle \right\|^2 = \sum_{j=k+1}^d \langle a, v_j \rangle^2$$

我们希望子空间  $E$  对全部数据  $X$  带来最小的均方损失，由上式可知：损失函数为  $\sum_{i=1}^n \|p_i - \pi(p_i)\|_2^2 = \|XV - XV_k\|_F^2 = \|XW\|_F^2$ ，其中  $V_k$  是一个后  $d - k$  列均为 0 的列正交矩阵， $W$  是一个  $d \times k$  的列正交矩阵。实际上， $(V_k[1:k], W) = (v_1, \dots, v_d) =: V$ 。所以问题转化为：

$$\begin{aligned} \min_{V_k \in \mathbb{R}^{d \times k}} \quad & - \|XV_k\|_F^2 \\ \text{s.t.} \quad & V_k^\top V_k = \mathbf{I}_k \end{aligned}$$

直接使用拉格朗日乘子法，可得

$$X^\top X v_i = \sigma_i^2 v_i.$$

即  $v_1, \dots, v_k$  对应奇异值分解的右乘方阵  $V$  的前  $k$  列。

主成分分析的缺点包括：

1. 复杂度高。基于 QR 分解的奇异值分解有复杂度  $O(nd^2)$ 。
2. 作用于矩阵，数值稳定性在维度高时难以保障。
3. 对数据多次读取，不适合分布式计算。

## 3 相关研究

### 3.1 低秩近似

**Theorem 3.1** (Eckart–Young–Mirsky). 对  $X \in \mathbb{R}^{n \times d}$ ， $\min_{\text{rank}(X_k) \leq k} \|X - X_k\|$  仅在  $X_k = \sum_{i=1}^k \sigma_i \hat{u}_i \hat{v}_i^\top$  时取到。此处的范数可以为谱范数或 Frobenius 范数， $\sigma_1, \dots, \sigma_d$  是  $X$  从大到小排列的奇异值， $\hat{u}_i, \hat{v}_j$  分别代表  $U, V$  的第  $i$  和第  $j$  列。

此外，一些基于采样的随机算法也被开发出来，参考 [1]。

## 3.2 非负矩阵分解

对非负矩阵  $M \in \mathbb{R}^{n \times m}$ , 求解

$$\begin{aligned} \min_{A, W \geq 0} \quad & \|AW - M\|_F \\ \text{s.t.} \quad & A \in \mathbb{R}^{n \times r}, W \in \mathbb{R}^{r \times m}. \end{aligned}$$

**Example 3.2.**  $M$  可以表示  $m$  个  $n$  维数据,  $A$  表示  $r$  个基向量,  $W$  表示这些基向量的凸组合。

推荐系统曾经广泛采用这种算法。求精确解的复杂度为  $O(mn^{O(r^2)})$ , 同样是  $NP$  完全问题。

## References

- [1] D. P. Woodruff. Sketching as a tool for numerical linear algebra. *Foundations and Trends in Theoretical Computer Science*, 10(1–2):1–157, 2014.