

SVD（奇异值分解）

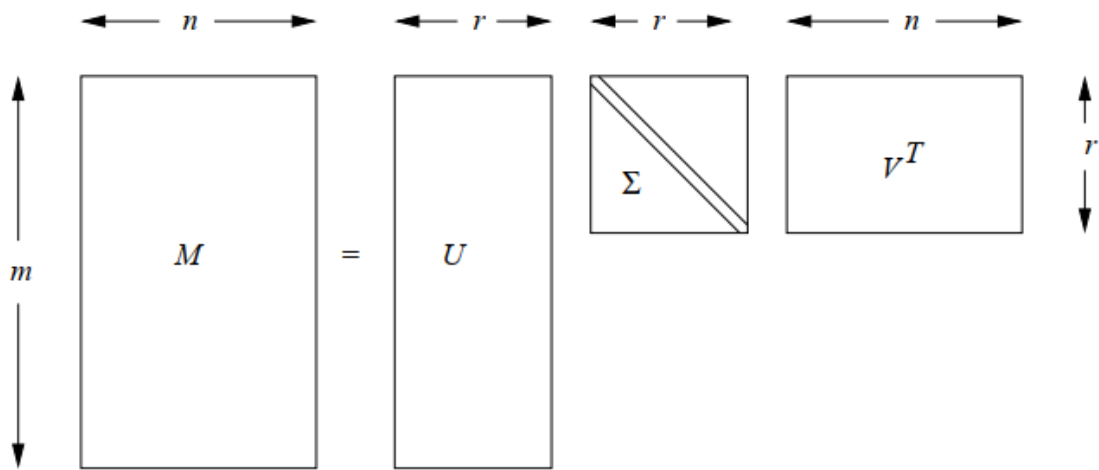
讨论矩阵降维的第二种形式，奇异值分解。该方法可以找到任何矩阵的精确表示，并且在给定维数大小的情况下消除其中不重要的维数。当然选择的维数越少，近似值越不准确。

首先介绍SVD定义的连接矩阵的行和列的概念，然后介绍了消除最不重要的内容的方法来实现更接近原始矩阵的低维表示。然后介绍如何使用这些概念来更有效地查询原始矩阵，最后提供了一种SVD执行算法。

SVD的定义

设矩阵 $M_{m \times n}$ ，定义矩阵 M 的秩为 r ，矩阵的行（列）秩指的是所有行（列）中线性独立的行（列）的最大个数。然后我们可以找到满足如下条件的矩阵 U, Σ, V ,

- 1. U 是大小为 $m \times r$ 的列正交矩阵：矩阵的每一列都是单位向量，任何列向量之间的点积为0。
- 2. V 是大小为 $n \times r$ 的行正交矩阵：通常我们使用 V^T ，因此 V^T 的行向量是相互正交的。
- 3. Σ 是大小为 $r \times r$ 的对角矩阵：非主对角线的元素都是0， Σ 的元素称为 M 的奇异值。



例如考虑观众和电影评分之间的矩阵，这里矩阵的行秩和列秩都为2.

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	0	0	4	4
Jenny	0	0	0	5	5
Jane	0	0	0	2	2

由于 $r = 2$ ，可以将矩阵 M 分解为下图，

$$\begin{matrix}
 \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 0 & 0 & 2 & 2 \end{bmatrix} & = & \begin{bmatrix} .14 & 0 \\ .42 & 0 \\ .56 & 0 \\ .70 & 0 \\ 0 & .60 \\ 0 & .75 \\ 0 & .30 \end{bmatrix} & \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} & \begin{bmatrix} .58 & .58 & .58 & 0 & 0 \\ 0 & 0 & 0 & .71 & .71 \end{bmatrix} \\
 M & & U & \Sigma & V^T
 \end{matrix}$$

其中， Σ 的大小为 2×2 。

SVD的解释

理解SVD的关键是理解 U , Σ , V 中 r 列作为隐藏在矩阵 M 中的代表概念。在上图的例子中，这些概念很清楚，其中一个“science fiction”另一个是“romance”。我们把矩阵 M 的行看做people，列看做movies。那么矩阵 U 将people和概念连接起来。例如，Joe对应矩阵 M 的第一行，他只喜欢“science fiction”这个概念。0.14这个值在 U 的第一行和第一列，小于在该列的其他值，因为Joe的评分并不高。第一行的第二列的值都是0，因为Joe并没有给romance电影打分。

矩阵 V 将movies和概念相连接， V^T 的第一行前三列的0.58表明前三个movies (matrix, alien, star wars)，每一个都属于“science fiction”，而后两列的0表明这些电影不属于romance概念，同时， V^T 第二行表明电影casablanca和titanic是属于romances。

矩阵 Σ 给出了每个概念的强度，在我们的例子中，“science fiction”的强度是12.4，“romance”的强度是9.5，直觉上讲，“science fiction”概念要比romance更强，因为数据提供了更多的该类电影和更多的人。

一般情况下，这些概念并不会清晰地描述，在 U , V 中有更少的0，即使 Σ 总是一个对角矩阵，对角线之外的值都是0。代表矩阵 M 的行和列的实例例如people和movies会在不同程度影响不同的概念。由于矩阵的秩和期望的 U , Σ , V 的列的个数相等，因此上图例子的压缩较为简单。我们可以得到一个矩阵 M 的精确压缩，在每个矩阵 U , Σ , V 只有两列的条件下，乘积 $U\Sigma V^T$ 在无限的精确条件下，就是准确的矩阵 M 。在实际应用中，例子不会很简单，如果 M 的秩比我们想要的 U , Σ , V 的列数要多的话，压缩是不精确的。我们需要从精确的压缩中消除匹配最小的奇异值的 U 和 V 的列，来达到最好的近似值，下面的例子是简单修改上图例子后的一个近似的例子。

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	2	0	4	4
Jenny	0	0	0	5	5
Jane	0	1	0	2	2

新矩阵 M' 中在alien中增加了两个打分。因此上图的矩阵 M' 的秩为3，例如第一、六、七行是线性无关的，下图则是对矩阵 M' 的压缩结果。

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = M'$$

$$\begin{bmatrix} .13 & .02 & -.01 \\ .41 & .07 & -.03 \\ .55 & .09 & -.04 \\ .68 & .11 & -.05 \\ .15 & -.59 & .65 \\ .07 & -.73 & -.67 \\ .07 & -.29 & .32 \end{bmatrix} \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \\ .40 & -.80 & .40 & .09 & .09 \end{bmatrix}$$

$$U \qquad \qquad \qquad \Sigma \qquad \qquad \qquad V^T$$

在 U, Σ, V 中我们使用了三列因为被压缩矩阵的秩为3, U 和 V 的列仍然和概念相关联, 第一个仍然是"science fiction", 第二个仍然是"romance", 第三列的概念很难解释, 但是它不是那么重要, 因为它在 Σ 中给出的权重是远低于其他两个权重的。

下一部分, 将会考虑移除不太重要的概念, 在本例子中, 可能会移除第三个列。

使用SVD降维

假设我们期望将一个大规模矩阵 M 通过SVD表达成矩阵 U, Σ, V 的形式, 但是这三个矩阵仍然较大, 影响存储, 最好的三个矩阵降维的方式是将最小的奇异值设为0, 如果将 s 个最小的奇异值设为0, 那么对应的可以将 U, V 中对应的 s 个行去掉。

例如:

上例中压缩结果有三个奇异值, 假设我们想要将维数降到2, 于是我们把最小的奇异值1.3设为0, 因此矩阵 U 的第三列和 V^T 的第三行乘以0, 因此该行和该列可以去掉。这意味着, 矩阵 M' 的近似可以由两个大的特征值得到。如图,

$$\begin{bmatrix} .13 & .02 \\ .41 & .07 \\ .55 & .09 \\ .68 & .11 \\ .15 & -.59 \\ .07 & -.73 \\ .07 & -.29 \end{bmatrix} \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \end{bmatrix}$$

$$= \begin{bmatrix} 0.93 & 0.95 & 0.93 & .014 & .014 \\ 2.93 & 2.99 & 2.93 & .000 & .000 \\ 3.92 & 4.01 & 3.92 & .026 & .026 \\ 4.84 & 4.96 & 4.84 & .040 & .040 \\ 0.37 & 1.21 & 0.37 & 4.04 & 4.04 \\ 0.35 & 0.65 & 0.35 & 4.87 & 4.87 \\ 0.16 & 0.57 & 0.16 & 1.98 & 1.98 \end{bmatrix}$$

结果矩阵和矩阵 M' 十分接近，具体的不同原因是将最后一个特征值设为0，然而，在这个简单例子中，大部分不同原因是由计算结果精确到两位有效数字的舍入误差引起的。

将较小的奇异值归0的原因

后续补充

使用概念查询

这部分讲解SVD如何帮助某个查询变得高效和精确。我们假定将要压缩原始的电影评分数据到SVD中，现在有一个人Quincy不是其中的一个人，但是他想通过该系统了解他可能喜欢的电影。她看了一个电影matrix并且打了4分，因此可以将Quincy表示为 $q = [4, 0, 0, 0, 0]$ ，作为原始矩阵其中一行。

如果使用协同过滤的办法，将要挨个对比原矩阵中的每个用户。我们将Quincy映射到概念空间，通过乘以矩阵 V ，则 $qV = [2.32, 0]$ ，也就是说，Quincy很有可能会对"science fiction"感兴趣，对romance不感兴趣。

现在我们在概念空间有了一个Quincy的代表，它是从原始电影空间派生的，但是不同于它的原始表达。一个有用的事是映射它的表达到电影空间通过 $[2.32, 0]$ 乘以 V^T ，结果是 $[1.35, 1.35, 1.35, 0, 0]$ ，表明Quincy可能会喜欢alien和star wars，但是不会喜欢casablanca或者titanic。

另一种可以在概念空间展示的查询顺序是找到和Quincy类似的用户。首先可以将所有的用户数据通过 V 映射到概念空间，例如Joe映射为 $[1.74, 0]$ ，Jill映射为 $[0, 5.68]$ 。注意在这个简单例子中，所有用户都是100%的science fiction fans或者100% romance fans，因此每个向量在其中一个值都是0。实际上，喜好是复杂的，因此他们的概念会不同但非零，通常情况下，我们可以通过概念空间的cosine距离来计算用户之间的相似性。

计算矩阵的SVD

矩阵 M 的SVD和对称矩阵 $M^T M$ 和 $M M^T$ 的特征值紧密联系，假设矩阵 M 的奇异值分解为 $M = U \Sigma V^T$ ，那么我们有

$$M^T = (U \Sigma V^T)^T = (V^T)^T \Sigma^T U^T = V \Sigma^T U^T \quad (1)$$

因为 Σ 是对角矩阵，因此 $M^T = V \Sigma U^T$ 。

然后有

$$M^T M = V \Sigma U^T U \Sigma V^T \quad (2)$$

又因为 U 是正交矩阵，则 $U^T U = I$ ，因此有

$$M^T M = V \Sigma^2 V^T \quad (3)$$

由于矩阵 V 是正交矩阵，因此 $V^T = V^{-1}$ ，则有

$$M^T M V = V \Sigma^2 \quad (4)$$

此处 Σ 是对角矩阵， Σ^2 也是一个对角矩阵，它的第 i 行和第 i 列都是 Σ 原位置值的平方，等式（4）表明矩阵 V 是矩阵 $M^T M$ 的特征向量组成的矩阵， Σ^2 是对角矩阵，它的非零值是对应的特征值。

因此，计算 $M^T M$ 的特征值和特征向量的算法给出了计算矩阵 M 的SVD中的 V 的算法。该算法也给出了SVD的特征值，即将 $M^T M$ 的特征值开方即可。

然后计算 U ，通过同样的计算 V 的算法计算，如

$$M M^T = U \Sigma V^T (U \Sigma V^T)^T = U \Sigma V^T V \Sigma U^T = U \Sigma^2 U^T \quad (5)$$

同理可得

$$M M^T U = U \Sigma^2 \quad (6)$$

意味着 U 是矩阵 $M M^T$ 的特征向量矩阵。

注意

关于 U 和 V 的细节需要解释，两个矩阵都有 r 列， $M^T M$ 是 $n \times n$ 大小的矩阵， $M M^T$ 是大小为 $m \times m$ 的矩阵。 n 和 m 至少和 r 是一样大的，因此 $M^T M$ 和 $M M^T$ 应该分别有 $n - r$ 和 $m - r$ 个额外的特征值，显然这些值在 U, V, Σ 中都不会出现，由于矩阵 M 的秩为 r ，其他的特征值为0，没有实际作用。