第15章 奇异值分解

VCG

2025-11-07

引言

- SVD是线性代数中最重要、最优美的概念之一
- 不仅是一个数学工具，更是一种看待数据和变换的哲学
- 其应用贯穿机器学习、信号处理、推荐系统等几乎所有数据科学领域

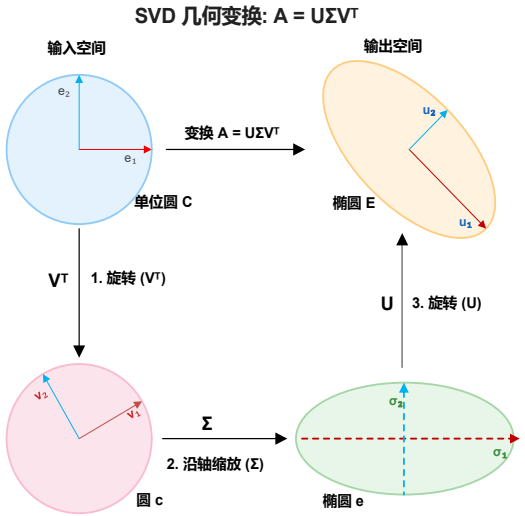
线性变换的本质？

- 一个矩阵 $\mathbf{A}$ 可以将一个向量 $\mathbf{x}$ 变换为另一个向量 $\mathbf{y} = \mathbf{Ax}$
  - 变换过程可能看起来非常复杂，它可能既有旋转，又有拉伸，还有切变...

- 我们能否找到一种“通用语言”来描述任何矩阵所代表的线性变换？
- 是否存在一种方法，能将这个复杂的变换分解为一系列最简单的、最基础的操作？

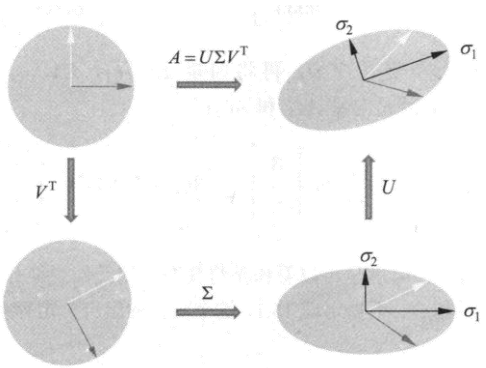
线性变换的几何含义：变换一个单位圆

- 线性变换（矩阵A）作用于单位圆C（内部均匀点云）
  - 当矩阵 **A** 作用于整个圆时，圆C变换成椭圆E
    - C的协方差矩阵为I，变换后的协方差矩阵为$S = AA^T$
      - S对称矩阵，正交对角化 => 输出空间椭圆E（长短轴为S特征向量方向）
      - 椭圆E，输出空间（长短轴为XY轴的）椭圆e旋转（矩阵U ）而来
      - 椭圆e，输入空间圆c对角阵（Σ，对角线元素为S特征值）变换而来
  - $A = U\Sigma V^T$ => $A^T = V\Sigma^T U^T$，类似的
    - C的协方差矩阵为I，变换后的协方差矩阵为$S = A^TA$
      - S对称矩阵，正交对角化 => 输入空间椭圆E（长短轴为S特征向量方向）
      - 椭圆E，由输入空间（长短轴为XY轴的）椭圆e旋转（矩阵V ）而来
      - 椭圆e，由输出空间圆c对角阵（Σ，对角线为S的特征值）而来
        - AA^T和A^TA的非零特征值是一样的！



矩阵变换分解变换“三部曲”

- SVD表明，任何复杂的线性变换 $\mathbf{A}$，都可以分解为三个纯粹的步骤
  - 一次旋转 (或反射) - $\mathbf{V}^T$
    - 在输入空间，将标准坐标系旋转，对齐到我们找到的那组特殊正交基 $\{\mathbf{v}_i\}$ 上
  - 一次纯粹的沿轴缩放 - $\mathbf{\Sigma}$
    - 在新的坐标系下，沿着每个轴进行独立的拉伸或压缩。缩放的比例为奇异值 σ_i。并转换到输出空间
  - 另一次旋转 (或反射) - $\mathbf{U}$
    - 在输出空间，将缩放后的坐标系旋转，对齐到输出椭圆的主轴基 $\{\mathbf{u}_i\}$ 上
- SVD：$\mathbf{A} = \mathbf{U\Sigma V}^T$
  - $\mathbf{V}^T$: 输入空间的旋转
  - $\mathbf{\Sigma}$: 沿轴缩放，并转换到输出空间
  - $\mathbf{U}$: 输出空间的旋转



奇异值分解

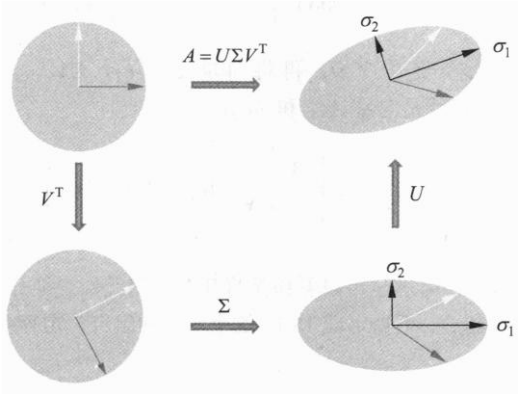
任何复杂的线性变换（矩阵A），无论它看起来多么扭曲和复杂，其本质都可以被SVD分解为这三个步骤：一个输入空间的旋转 (VT)，一次纯粹的沿轴缩放 (Σ)，以及一个输出空间的旋转 (U)。SVD就是找到了完成这个优雅过程的精确“蓝图”

奇异值分解定义

- 对于任何一个 $m \times n$ 的实矩阵 $\mathbf{A}$，其奇异值分解 (SVD) 形式为：

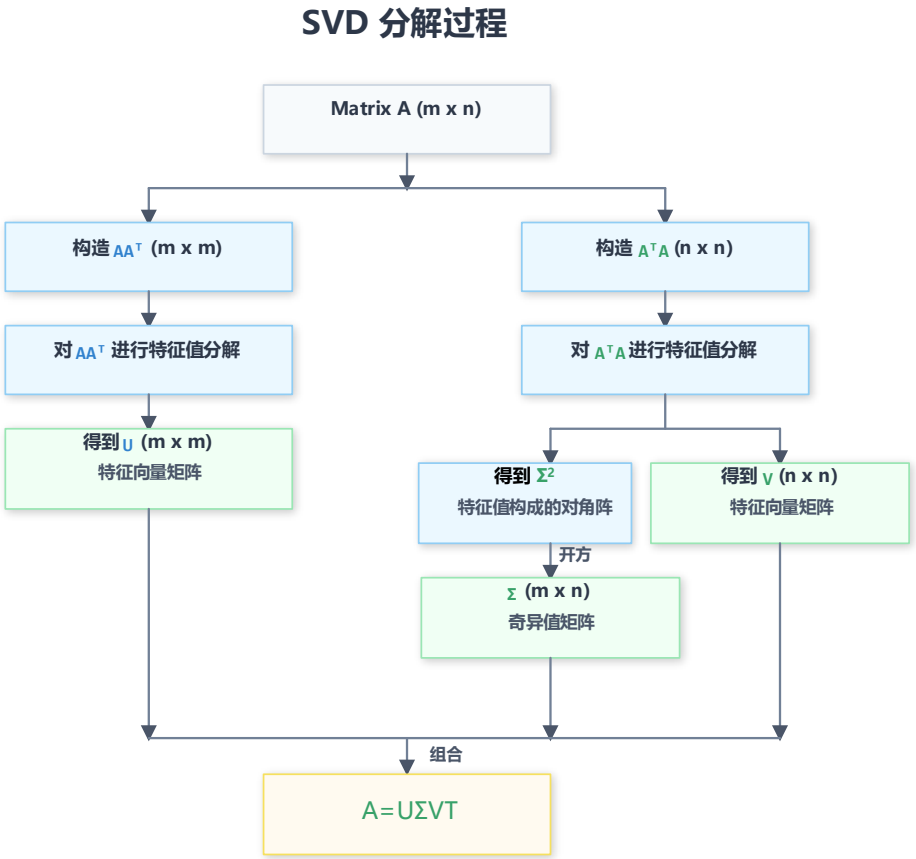
$$\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m}\mathbf{\Sigma}_{m\times n}(\mathbf{V}_{n\times n})^T$$

  - $\mathbf{U}$ $(m \times m)$: 左奇异向量 (Left Singular Vectors)
    - 正交矩阵 $(\mathbf{U}^T\mathbf{U} = \mathbf{I})$
    - 列向量 $\{\mathbf{u}_i\}$ 构成了输出空间的一组标准正交基
  - $\mathbf{V}$ $(n \times n)$: 右奇异向量 (Right Singular Vectors)
    - 正交矩阵 $(\mathbf{V}^T\mathbf{V} = \mathbf{I})$
    - 列向量 $\{\mathbf{v}_i\}$ 构成了输入空间的一组标准正交基
  - $\mathbf{\Sigma}$ $(m \times n)$: 奇异值矩阵 (Singular Values)
    - 对角矩阵（非方阵时为“伪对角”）
    - 对角线上元素 $\sigma_1 \geq \sigma_2 \geq \ldots \geq 0$ 称为奇异值，其大小代表了数据在对应方向上的“重要性”/“能量”



如何计算SVD？

➢算法


SVD 分解过程
Matrix A (m x n)
构造 AA^T (m x m)
构造 A^TA (n x n)
对 AA^T 进行特征值分解
对 A^TA 进行特征值分解
得到 U (m x m)
特征向量矩阵
得到 Σ^2
特征值构成的对角阵
得到 V (n x n)
特征向量矩阵
开方
Σ (m x n)
奇异值矩阵
组合
A=UΣVT

如何计算SVD？

- 计算 $\mathbf{V}$ 和 $\mathbf{\Sigma}$
  - 构造矩阵 $\mathbf{A}^T\mathbf{A}$ (一个 $n \times n$ 的对称矩阵)
  - $\mathbf{A}^T\mathbf{A} = (\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T)^T(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T) = \mathbf{V}\mathbf{\Sigma}^T\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{V}(\mathbf{\Sigma}^T\mathbf{\Sigma})\mathbf{V}^T$
    - $\mathbf{V}$ 的列向量就是 $\mathbf{A}^T\mathbf{A}$ 的特征向量
    - 奇异值的平方 σ_i^2 就是 $\mathbf{A}^T\mathbf{A}$ 的特征值
- 计算 $\mathbf{U}$
  - $\mathbf{A}\mathbf{v}_i = \sigma_i\mathbf{u}_i$
  - 或者
    - 类似地，构造矩阵 $\mathbf{A}\mathbf{A}^T$ (一个 $m \times m$ 的对称矩阵)
    - $\mathbf{A}\mathbf{A}^T = \mathbf{U}(\mathbf{\Sigma}\mathbf{\Sigma}^T)\mathbf{U}^T$
- 结论
  - $\mathbf{U}$ 的列向量就是 $\mathbf{A}\mathbf{A}^T$ 的特征向量
  - 计算流程：对 $\mathbf{A}^T\mathbf{A}$ 进行特征值分解得到 $\mathbf{V}$ 和 $\mathbf{\Sigma}$，然后可以通过关系 $\mathbf{A}\mathbf{v}_i = \sigma_i\mathbf{u}_i$ 求出 $\mathbf{U}$

SVD与PCA

- PCA的目标是找到数据协方差矩阵 $\mathbf{S} = \frac{1}{N-1}\mathbf{X}^T\mathbf{X}$ 的特征向量
- SVD的视角
  - 对中心化数据 $\mathbf{X}$ 进行SVD ($\mathbf{X} = \mathbf{U\Sigma V}^T$)，那么 $\mathbf{X}^T\mathbf{X} = \mathbf{V}(\mathbf{\Sigma}^T\mathbf{\Sigma})\mathbf{V}^T$
  - 意味着
    - PCA的主成分就是SVD的右奇异向量 $\mathbf{V}$
    - 降维后的数据（主成分得分）可以通过 $\mathbf{Z} = \mathbf{XV} = \mathbf{U\Sigma}$ 直接得到
- 为什么有时用SVD实现PCA?
  - 直接对 $\mathbf{X}$ 做SVD比先计算 $\mathbf{X}^T\mathbf{X}$ 再做特征值分解更稳定
  - 对于“维度 > 样本数”的宽表数据，SVD更高效，且有其它的优化方法

总结

- SVD的本质
  - 将任何线性变换分解为旋转-缩放-旋转三部曲
- 几何意义
  - 找到了输入空间和输出空间中的两组特殊正交基，使得变换在这两组基之间只有缩放
- 计算核心
  - 通过对 $\mathbf{A}^T\mathbf{A}$ 和 $\mathbf{A}\mathbf{A}^T$ 进行特征值分解来找到SVD的各个组成部分
- 最强大的应用
  - 低秩近似。通过保留最大的奇异值，实现数据压缩、去噪和核心模式提取
- 与PCA的关系
  - SVD为PCA提供了一种更通用、更稳健的计算途径

Thanks

奇异值分解的另一种解释：外积展开形式

$A=U\Sigma V$^T=σ_1 u_1 v_1^T+σ_2 u_2 v_2^T+⋯+σ_n u_n v_n^T

变换视角，把A看作一系列变换组合而成

数据视角，把A看作一系列（投影）分量组合而成

SVD最强大的应用：低秩近似

- SVD的外积展开形式：

$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \, \mathbf{u}_i \mathbf{v}_i^T$$

其中 r 是矩阵的秩

  - 直观理解：任何矩阵都可以看作是多个“秩为1”的简单矩阵 $(\mathbf{u}_i \mathbf{v}_i^T)$ 的加权和
  - 权重就是奇异值 σ_i

- 低秩近似 (Low-Rank Approximation)：
  - 由于奇异值是降序排列的 $(\sigma_1 \geq \sigma_2 \geq \ldots)$，最大的几个奇异值包含了矩阵最主要的信息
  - 我们可以通过只保留前 k 个最大的奇异值来近似原始矩阵：

$$\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i \, \mathbf{u}_i \mathbf{v}_i^T \quad (k < r)$$

- Eckart-Young定理：$\mathbf{A}_k$ 是在所有秩为 k 的矩阵中，与原始矩阵 $\mathbf{A}$ 最接近的矩阵（在Frobenius范数意义下）

低秩近似的威力：图像压缩实例

- 一张灰度图像可以看作一个矩阵，每个像素值是矩阵的一个元素
- 原始图像 (秩 r)
  - 存储需要 $m \times n$ 个数值
- SVD近似 (秩 k)
  - 我们只需要存储 k 个奇异值、 k 个 $\mathbf{u}$ 向量 (每个 m 维)、 k 个 $\mathbf{v}$ 向量 (每个 n 维)
  - 总存储量为 $k + k \times m + k \times n = k(1 + m + n)$
  - 当 $k \ll r$ 时，可以实现巨大的数据压缩

- 结论：SVD能够抓住数据中最核心的“模式”，并用极少的成分来重构它

SVD的两种实用形式

- 完全SVD (Full SVD)
  - $\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m}\mathbf{\Sigma}_{m\times n}\mathbf{V}^T_{n\times n}$
  - $\mathbf{U}$ 和 $\mathbf{V}$ 都是方阵，$\mathbf{\Sigma}$ 的尺寸与 $\mathbf{A}$ 相同，可能包含很多零。在数学上完备，但在计算上冗余
- 紧SVD (Compact SVD / Thin SVD)
  - $\mathbf{A}_{m\times n} = \mathbf{U}_{m\times r}\mathbf{\Sigma}_{r\times r}\mathbf{V}^T_{n\times r}$
  - 只保留与非零奇异值对应的 r 个奇异向量
  - $\mathbf{\Sigma}$ 变成了 $r \times r$ 的方阵。这是无损压缩
- 截断SVD (Truncated SVD)
  - $\mathbf{A}_{m\times n} \approx \mathbf{U}_{m\times k}\mathbf{\Sigma}_{k\times k}\mathbf{V}^T_{n\times k} \quad (k < r)$
  - 只保留前 k 个最大的奇异值和对应的奇异向量
  - 这就是我们前面讲的低秩近似，是有损压缩，也是SVD在机器学习中最常用的形式