

1.5. 矩阵的秩

在解线性方程组 $\mathbf{Ax} = \mathbf{b}$ 时:

$$\mathbf{A} \xrightarrow{\text{行}} \text{rref}(\mathbf{A})$$

$$(\mathbf{A}, \mathbf{b}) \xrightarrow{\text{行}} \text{rref}(\mathbf{A}, \mathbf{b})$$

系数矩阵 $\mathbf{A}$ 和增广矩阵 $(\mathbf{A}, \mathbf{b})$ 的最简行阶梯型 $\text{rref}(\mathbf{A})$ 和 $\text{rref}(\mathbf{A}, \mathbf{b})$ 的非零行数相等, 则方程组有解, 否则无解!

问题

给定矩阵 $\mathbf{A}$, $\text{rref}(\mathbf{A})$ 的非零行数是否由 $\mathbf{A}$ 唯一确定?

即: 与将 $\mathbf{A}$ 变为 $\text{rref}(\mathbf{A})$ 所采用的初等行变换的选取无关?

注: 实际上, 甚至 $\text{rref}(\mathbf{A})$ 也是由 $\mathbf{A}$ 唯一确定的!

假设 $\mathbf{A}$ 的最简行阶梯型 $\text{rref}(\mathbf{A})$ 有 r 个非零行, 比如:

例

$$\text{假设 } \text{rref}(\mathbf{A}) = \begin{pmatrix} 0 & 1 & 2 & 0 & 3 & 0 & 4 \\ 0 & 0 & 0 & 1 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

可对 $\text{rref}(\mathbf{A})$ 进一步用列初等变换, 将其化为

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

通过上面的例子可以发现，如果 $\mathbf{A}$ 的最简行阶梯型 $\text{rref}(\mathbf{A})$ 有 r 行非零，可进一步作列初等变换，将其化为

$$\begin{pmatrix} \mathbf{I}_r & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{pmatrix}$$

形式。

引理

上面矩阵中的 r 是由 $\mathbf{A}$ 唯一确定的，叫作矩阵 $\mathbf{A}$ 的秩(rank)，记作 $\text{rank}(\mathbf{A})$ 。

矩阵的秩有下面几条基本的性质。

命题

假设 $\mathbf{A}$ 为 m 行 n 列矩阵，则下列结论成立.

- ① 如果 $\mathbf{A}$ 只有 l 行 (列) 非零，则 $\text{rank}(\mathbf{A}) \leq l$;
- ② $\text{rank}(\mathbf{A}) \leq \min\{m, n\}$;
- ③ 初等变换不改变矩阵的秩;
- ④ 若 $\mathbf{P}, \mathbf{Q}$ 可逆，则 $\text{rank}(\mathbf{PAQ}) = \text{rank}(\mathbf{A})$;
- ⑤ $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}^T)$

从定义还容易得到如下结论。

命题

n 阶方阵 $\mathbf{A}$ 可逆当且仅当 $\text{rank}(\mathbf{A}) = n$.

利用矩阵秩的语言，还可以重新总结线性方程组解的条件。

定理

对于线性方程组 $\mathbf{Ax} = \mathbf{b}$, 其中 $\mathbf{A} \in M_{m,n}(F)$, 下列结论成立。

- ① $\mathbf{Ax} = \mathbf{b}$ 有解，当且仅当 $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}, \mathbf{b})$;
- ② $\mathbf{Ax} = \mathbf{b}$ 有唯一解，当且仅当 $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}, \mathbf{b}) = n$.
- ③ $\mathbf{Ax} = \mathbf{b}$ 有无穷多解，当且仅当 $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}, \mathbf{b}) < n$.

一个重要但不显然的结论是

命题

两个矩阵乘积的秩不超过其中任何一个矩阵的秩，即

$$\text{rank}(\mathbf{AB}) \leq \min\{\text{rank}(\mathbf{A}), \text{rank}(\mathbf{B})\}$$

证明: 假设 $\text{rank}(\mathbf{B}) = r$ ，则存在可逆矩阵 $\mathbf{P}, \mathbf{Q}$ 使得

$$\mathbf{PBQ} = \begin{pmatrix} \mathbf{I}_r & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{pmatrix}.$$

因此 $\mathbf{AB} = \mathbf{AP}^{-1}\mathbf{PBQQ}^{-1}$. 左右两边同时乘可逆矩阵不改变矩阵的秩，因此

$$\text{rank}(\mathbf{AB}) = \text{rank}(\mathbf{AP}^{-1}(\mathbf{PBQ})).$$

因为 $\mathbf{PBQ}$ 只有前 r 列非零, 所以 $\mathbf{AP}^{-1}(\mathbf{PBQ})$ 至多前 r 列非零, 即

$$\text{rank}(\mathbf{AB}) = \text{rank}(\mathbf{AP}^{-1}(\mathbf{PBQ})) \leq r,$$

即 $\text{rank}(\mathbf{AB}) \leq \text{rank}(\mathbf{B})$.

此外, $\text{rank}(\mathbf{AB}) = \text{rank}(\mathbf{B}^T \mathbf{A}^T) \leq \text{rank}(\mathbf{A}^T) = \text{rank}(\mathbf{A})$.

上述命题有个重要的推论。

推论

如果 n 阶方阵 A, B 满足 $AB = I_n$, 那么 A, B 可逆且互为逆矩阵。

应用: 矩阵分解与推荐系统

假设 $\mathbf{A}$ 是 $m \times n$ 矩阵, 且 $\text{rank}(\mathbf{A}) = r$, 则存在可逆矩阵 $\mathbf{P} \in M_{m,m}(F)$, $\mathbf{Q} \in M_{n,n}(F)$ 使得

$$\mathbf{A} = \mathbf{P} \begin{pmatrix} \mathbf{I}_r & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{pmatrix} \mathbf{Q} = \mathbf{P} \begin{pmatrix} \mathbf{I}_r \\ \mathbf{O} \end{pmatrix} (\mathbf{I}_r \quad \mathbf{O}) \mathbf{Q}.$$

令

$$\mathbf{U} = \mathbf{P} \begin{pmatrix} \mathbf{I}_r \\ \mathbf{O} \end{pmatrix}, \quad \mathbf{V} = (\mathbf{I}_r \quad \mathbf{O}) \mathbf{Q}.$$

则 $\mathbf{U}$ 是 $m \times r$ 矩阵, $\mathbf{V}$ 是 $r \times n$ 矩阵, 且 $\mathbf{A} = \mathbf{UV}$.

当 r 比较小, 而 m, n 比较大时, 这种分解可以大大减少存储空间。因为此时 $rm + rn$ 都远小于 mn , 存储 $\mathbf{U}$ 和 $\mathbf{V}$ 所需的空间远小于存储 $\mathbf{A}$ 所需的空间。

假设有 4 个用户、5 部电影，我们把评分排成矩阵

$$R = \begin{pmatrix} 5 & 4 & ? & 1 & ? \\ 4 & 5 & ? & 1 & 2 \\ ? & ? & 5 & 4 & 5 \\ 1 & 1 & 4 & 5 & 4 \end{pmatrix}.$$

例如：

	复仇者联盟	钢铁侠	爱乐之城	泰坦尼克号	恋恋笔记本
A	5	4	?	1	?
B	4	5	?	1	2
C	?	?	5	4	5
D	1	1	4	5	4

假如有 10 万部电影，一个用户就是一个 10 万维向量：

$$(5, 4, 0, 1, \dots).$$

但这其实很浪费，因为用户的兴趣往往可以用少数几个“潜在因素”描述。例如可以假设：

$$\text{兴趣因素} = \left\{ \begin{array}{l} \text{动作片偏好,} \\ \text{爱情片偏好,} \\ \text{喜剧片偏好,} \\ \text{科幻片偏好,} \\ \text{文艺片偏好.} \end{array} \right.$$

于是一个用户 A 不必用 10 万个数字表示，只需要例如

$$u_A = (0.9, 0.1, 0.3, 0.8, 0.2),$$

假设用户兴趣只有 k 个潜在因素。设用户矩阵为

$$U = \begin{pmatrix} - & - & - & u_1 & - & - & - \\ - & - & - & u_2 & - & - & - \\ & & & \vdots & & & \\ - & - & - & u_m & - & - & - \end{pmatrix} \in \mathbb{R}^{m \times k},$$

电影矩阵为

$$V = \begin{pmatrix} - & - & - & v_1 & - & - & - \\ - & - & - & v_2 & - & - & - \\ & & & \vdots & & & \\ - & - & - & v_n & - & - & - \end{pmatrix} \in \mathbb{R}^{n \times k}.$$

评分矩阵就可以近似写成

$$R \approx UV^T.$$

我们真正想知道的是矩阵中的“?”。例如

$$R = \begin{pmatrix} 5 & 4 & ? & 1 \\ 4 & 5 & ? & 1 \\ ? & ? & 5 & 4 \end{pmatrix}.$$

我们希望找到两个低维矩阵 U, V , 使得在已有评分的位置,

$$R_{ij} \approx u_i^T v_j.$$

因此可以解一个优化问题:

$$\min_{U, V} \sum_{(i,j) \in \Omega} (R_{ij} - u_i^T v_j)^2,$$

其中 Ω 表示“已经有评分的位置”。

训练完成以后，对于没有评分的位置，直接计算

$$\hat{R}_{ij} = u_i v_j^T.$$

比如预测

$$\hat{R}_{A, \text{泰坦尼克号}} = 1.7,$$

那系统就认为 A 大概不会喜欢；如果

$$\hat{R}_{A, \text{星球大战}} = 4.8,$$

系统就很可能向 A 推荐《星球大战》。