线性代数

huhao

February 6, 2023

基础知识

- 线性空间
- 基
- 线性变换
- 矩阵、基变换

特征值、特征向量、特征多项式

对于矩阵 A，满足 $Av = \lambda v$ 的 λ 是特征值，v 是特征向量。

不难发现，当且仅当 $\det(A - \lambda I) = 0$，λ 是特征向量，不妨称 $f(x) = \det(A - xI)$ 为 A 的特征多项式。

例

求:

$$\begin{bmatrix} & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ a_1 & a_2 & a_3 & \dots & a_n \end{bmatrix}$$

的特征多项式。

Cayley–Hamilton Theorem

若 f 是 A 的特征多项式，则 $f(A)=0$。

所以 $A^n = (x^n \bmod f)|_{x=A}$。

综合此页和上一页的内容可以得到常系数齐次线性递推的做法。

矩阵对角化

如果矩阵 A 有 $n = \dim A$ 个线性无关的特征向量 $\alpha_1 \ldots \alpha_n$，那么它在 α 下的矩阵是对角矩阵。

令 P 为所有线性无关的特征向量（列向量）拼接组成的矩阵，则：

$$D = P^{-1}AP$$

为对角矩阵。

例

对角化:

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

```
In[67]:= A = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}
         DiagonalizableMatrixQ[A]
         对角化矩阵判定
         MatrixForm[P = FullSimplify[Transpose[Eigenvectors[A]]]]
         矩阵格式       完全简化      转置     特征向量
         MatrixForm[d = FullSimplify[Inverse[P].A.P]]
         矩阵格式       完全简化      逆
         A == P.d.Inverse[P]
                  逆

Out[67]= {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}

Out[68]= True
```

Out[69]//MatrixForm=

$$\begin{pmatrix} \frac{1}{22}\left(-11+3\sqrt{33}\right) & \frac{1}{22}\left(-11-3\sqrt{33}\right) & 1 \\ \frac{1}{44}\left(11+3\sqrt{33}\right) & \frac{1}{44}\left(11-3\sqrt{33}\right) & -2 \\ 1 & 1 & 1 \end{pmatrix}$$

Out[70]//MatrixForm=

$$\begin{pmatrix} \frac{3}{2}\left(5+\sqrt{33}\right) & 0 & 0 \\ 0 & -\frac{3}{2}\left(-5+\sqrt{33}\right) & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Out[71]= True

不一定所有矩阵都可以找到 n 个线性无关的特征向量。

```
In[125]:= MatrixForm[A = {{1, 2}, {0, 1}}]
          矩阵格式
          DiagonalizableMatrixQ[A]
          对角化矩阵判定
Out[125]//MatrixForm=
          (1 2)
          (0 1)

Out[126]= False
```

但是，如果将 $(A-\lambda I)^{\dim A}v=0$ 的 v 称为广义特征向量，那么一定会有 $\dim A$ 个线性无关广义特征向量。

考虑若干个 $v_{1,1}\dots v_{1,m_1}\dots v_{k,1}\dots v_{k,m_k}$ 是 n 个线性无关的广义特征向量，且:

$$(A-\lambda_i I)v_{i,j}=v_{i,j+1}$$

那么 A 在这组基下就是:

$$\begin{bmatrix} A_1 & & & \\ & A_2 & & \\ & & \ddots & \\ & & & A_k \end{bmatrix}$$

且 A_i 是大小为 m_i，对角线和对角线上面一格为 1 其它为 0 的矩阵。

这样子的 A 可以很方便的计算幂。

例

计算

$$\begin{bmatrix} & 1 & \\ & & 1 \\ -1 & -3 & -3 \end{bmatrix}^n$$

```
In[278]:= MatrixForm[A = {{0, 1, 0}, {0, 0, 1}, {-1, -3, -3}}]
          矩阵格式
          i = IdentityMatrix[3]
              单位矩阵
          Solve[(A + i).{x, y, z} == {0, 0, 0}, {x, y, z}]
          解方程
          Solve[(A + i).{x, y, z} == {1, -1, 1}, {x, y, z}]
          解方程
          Solve[(A + i).{x, y, z} == {0, 1, -2}, {x, y, z}]
          解方程
          MatrixForm[P = Transpose[{{1, -1, 1}, {0, 1, -2}, {0, 0, 1}}]]
          矩阵格式      转置
          MatrixForm[D = Inverse[P].A.P]
          矩阵格式    ···  逆
```

Out[278]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -3 & -3 \end{pmatrix}$$

Out[279]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

Out[280]= {{y → -x, z → x}}

Out[281]= {{y → 1 - x, z → -2 + x}}

Out[282]= {{y → -x, z → 1 + x}}

Out[283]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & -2 & 1 \end{pmatrix}$$

Out[284]//MatrixForm=

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

当然也可以用 mathematica 内置的。

```
In[291]:= MatrixForm[A = {{0, 1, 0}, {0, 0, 1}, {-1, -3, -3}}]
          矩阵格式
          {P, d} = JordanDecomposition[A]
                   约旦分解
          MatrixForm[P]
          矩阵格式
          MatrixForm[d]
          矩阵格式
```

Out[291]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -3 & -3 \end{pmatrix}$$

Out[292]= {{{1, 2, 3}, {-1, -1, -1}, {1, 0, 0}}, {{-1, 1, 0}, {0, -1, 1}, {0, 0, -1}}}

Out[293]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 \\ -1 & -1 & -1 \\ 1 & 0 & 0 \end{pmatrix}$$

Out[294]//MatrixForm=

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

这个幂是很容易算的

```
In[295]:= MatrixForm[MatrixPower[d, n]]
          矩阵格式    矩阵的幂

Out[295]//MatrixForm=
```

$$\begin{pmatrix} (-1)^n & (-1)^{1+n}\,n & \frac{1}{2}(-1)^n(-1+n)\,n \\ 0 & (-1)^n & (-1)^{1+n}\,n \\ 0 & 0 & (-1)^n \end{pmatrix}$$

例

A 是上三角矩阵，求 A^k，其中 n 远小于 $\log k$。

QR 分解

矩阵的 QR 分解是一个很重要的算法，对于矩阵 A，要求出正交矩阵 Q 和上三角矩阵 R 使得：

$$A = QR$$

分解方法为对 A 的每一个列向量 a_i 使用 Gram–Schmidt 过程得到 e_i，然后：

$$\begin{aligned} Q =& [e_i] \\ R =& [\langle e_i, a_j \rangle] \end{aligned}$$

可以参考：*https://pi.math.cornell.edu/~web6140/TopTenAlgorithms/QRalgorithm.html*

若要求 A 的特征值，可以令 $A_0 = A$，且：$A_i = Q_i R_i, A_{i+1} = R_i Q_i$，则：

$$A_{i+1} = R_i Q_i = Q_i^{-1} Q_i R_i Q_i = Q_i^{-1} A_i Q_i$$

不难发现，A_{i+1} 和 A_i 相似，有着相同的特征值。

不断这样计算，直到收敛至上三角矩阵。

上海森伯格化

海森伯格矩阵是只有对角线上方，以及对角线下一个元素非 0 的矩阵。

考虑

$$A = \begin{bmatrix} a & A_1 \\ A_2 & A_3 \end{bmatrix}$$

那么如果 T_i 是满足 $TA_2 = e_1$，那么：

$$\begin{bmatrix} 1 & \\ & T_i \end{bmatrix} A \begin{bmatrix} 1 & \\ & T_i^{-1} \end{bmatrix}$$

可以将第 1 列 $3 \sim$ 变为 0，然后一次做第 $2 \ldots$ 列即可。

令 $\hat{T}_i = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & T_i \end{bmatrix}$ 这样可以得到 $\hat{T} = \hat{T}_{n-2} \ldots \hat{T}_1$ 满足 $\hat{T} A \hat{T}^{-1}$ 是海森伯格矩阵。

上海森伯矩阵的 QR 分解

考虑相邻两行，对这两行应用旋转变换：

$$G_1^T H = \begin{array}{rcccccc} \begin{pmatrix} c_1 & s_1 \\ -s_1 & c_1 \end{pmatrix} * & * & * & * & * & * & * \\ & * & * & * & * & * & * \\ & & * & * & * & * & * \\ & & & * & * & * & * \\ & & & & * & * & * \\ & & & & & * & * \end{array}$$

$$G_2^T G_1^T H = \begin{array}{rcccccc} & * & * & * & * & * & * \\ \begin{pmatrix} c_2 & s_2 \\ -s_2 & c_2 \end{pmatrix} * & & * & * & * & * & * \\ & & * & * & * & * & * \\ & & & * & * & * & * \\ & & & & * & * & * \\ & & & & & * & * \end{array}$$

$$G_3^T G_2^T G_1^T H = \begin{array}{rcccccc} & * & * & * & * & * & * \\ & & * & * & * & * & * \\ \begin{pmatrix} c_3 & s_3 \\ -s_3 & c_3 \end{pmatrix} * & & & * & * & * & * \\ & & & * & * & * & * \\ & & & & * & * & * \\ & & & & & * & * \end{array}$$

$$G_4^T G_3^T G_2^T G_1^T H = \begin{array}{rcccccc} & * & * & * & * & * & * \\ & & * & * & * & * & * \\ & & & * & * & * & * \\ \begin{pmatrix} c_4 & s_4 \\ -s_4 & c_4 \end{pmatrix} * & & & & * & * & * \\ & & & & * & * & * \end{array}$$

可以发现，逆回来也是上海森伯矩阵：

$$G_5^T \ldots G_1^T H G_1 = \begin{matrix} * & * & * & * & * & * \\ & * & * & * & * & * \\ & & * & * & * & * \\ & & & * & * & * \\ & & & & * & * \\ & & & & & * \end{matrix} \quad * \begin{pmatrix} c_1 & -s_1 \\ s_1 & c_1 \end{pmatrix}$$

$$G_5^T \ldots G_1^T H G_1 G_2 = \begin{matrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ & & * & * & * & * \\ & & & * & * & * \\ & & & & * & * \\ & & & & & * \end{matrix} \quad * \begin{pmatrix} c_2 & -s_2 \\ s_2 & c_2 \end{pmatrix}$$

$$G_5^T \ldots G_1^T H G_1 G_2 G_3 = \begin{matrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ & * & * & * & * & * \\ & & & * & * & * \\ & & & & * & * \\ & & & & & * \end{matrix} \quad * \begin{pmatrix} c_3 & -s_3 \\ s_3 & c_3 \end{pmatrix}$$

$$G_5^T \ldots G_1^T H G_1 G_2 G_3 G_4 = \begin{matrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ & * & * & * & * & * \\ & & * & * & * & * \\ & & & & * & * \\ & & & & & * \end{matrix} \quad * \begin{pmatrix} c_4 & -s_4 \\ s_4 & c_4 \end{pmatrix}$$

平移加速

一般来说，A 的特征值间差异越大，QR 算法收敛越快。

不妨令：

$$A_i - \delta I = QR, A_{i+1} = RQ + \delta I$$

那么 A_{i+1} 的特征值和 A_i 的特征值依然是一样的。

一般来说，δ 可以取 A 对角线上的 2×2 的方阵的特征值。