矩阵特征值的迭代求法

幂方法

- 基本思想：考虑一个有 n 个特征值且特征值的大小满足 $\lambda_1 > \lambda_2 \geq \lambda_3 \cdots \geq \lambda_n$ 的只有唯一最大特征值的矩阵 A 。设与 λ_j 相对应的特征向量为 $\vec{v}^{(j)}$ 。对于任意的初始向量 $\vec{x}$ ，有

  $$\vec{x} = \sum_{j=1}^{n} \beta_j \vec{v}^{(j)}$$

  则因为$A^n \vec{v} = \lambda^n \vec{v}$ 故而

  $$A^n \vec{x} = \sum_{j=1}^{n} \beta_j \lambda_j^n \vec{v}^{(j)} = \lambda_1^n (\sum_{j=1}^{n} \beta_j (\frac{\lambda_j}{\lambda_1})^n \vec{v}^{(j)})$$

  当 n 足够大时，会收敛到 $\beta_1 \lambda_1^n \vec{v}^{(1)}$ ，此时最大的特征值和其对应的特征向量已经凸显出来了，只需要想办法将两者分别求出即可。比如将 n 足够大时的相邻两项相除，就可以得到 λ_1。

- 可能出现的问题
  - 溢出或者舍入的问题 : 考虑每一次迭代时对 $\vec{x}$ 进行某种形式的归一化处理，如统一除以向量中的最大元素，让 $||\vec{x}||_\infty = 1$
  - 矩阵不具备 “存在唯一的最大特征值” 的性质：是幂方法的缺点，没有具体的解决方法。
  - 初始向量选取的很差，使得最大特征向量对应的系数为 0 ,导致后续过程无法正常进行：幂方法的缺点，没有具体的解决方法。

- 主要过程：
  1. 首先调整初始向量，使 $||\vec{x}^{(0)}||_\infty = 1$
  2. 计算 $\vec{y}^{(1)} = A\vec{x}^{(0)}$
  3. 令 $u^{(1)} = ||\vec{y}^{(1)}||_\infty$， $\vec{x}^{(1)} = \vec{y}^{(1)}/u^{(1)}$
  4. 重复 2,3 步，当 $||\vec{x}^{(n)} - \vec{x}^{(n-1)}|| < errorbound$ 时计算停止输出 $u^{(n)}$ 作为预测的特征值，此时的 $\vec{x}^{(n)}$ 作为特征向量。

- 加速收敛： $Aitkan - \Delta^2$ 方法。
  - 基本思路：不是直接将 $\{u^{(n)}\}$ 作为收敛到特征值的序列，而是将

    $$\hat{u}^{(n)} = u^{(n-2)} - \frac{(u^{(n-1)} - u^{(n-2)})^2}{u^{(n)} - 2u^{(n-1)} + u^{(n-2)}}$$

    作为新的序列。将新数列最终收敛到的值作为求出的特征值，特征向量依然是原来求出的特征向量。
    Note: 采取这个方法时，不改变任何原来的计算过程。可以按不考虑 $u^{(n)}$ 时完全一样的过程进行，只是需要对求出的 $\{u^{(n)}\}$ 多进行一步的处理，得出 $\hat{u}^{(n)}$。

对称矩阵的幂方法

- 基本思想：对于对称矩阵来说，可以采取新的确定 $\{u^{(n)}\}$ 的方式来加速幂方法的迭代过程。事实上：

  $$\hat{u}^{(n)} = \vec{x}^{(n-1)t} \vec{y}^{(n)} = \vec{x}^{(n-1)t} A \vec{x}^{(n-1)}$$

  同时在原有的计算过程中的所有无穷范数都用 2-范数来替换。

逆幂方法

- 基本思想：如何借助幂方法取出所有的特征值和特征向量呢？可以考虑构造一个新的矩阵将其他特征值变成最大的特征值，从而利用幂方法求出。

- 具体过程：考虑矩阵 $(A - qI)^{-1}$ ，该矩阵的所有特征值为:

  $$\frac{1}{\lambda_1 - q}, \frac{1}{\lambda_2 - q}, \cdots, \frac{1}{\lambda_n - q}$$

  其中 λ_i 为矩阵 A 的特征值。且对应于 $1/(\lambda_i - q)$ 的特征向量也与 A 的对应于 λ_i 的 特征向量相同。只需要选取 q 尽量的靠近要求的特征值，就可以通过幂方法求出该特征值。

- 加速策略：
  - 选取 q 满足：

$$q = \frac{\vec{x}^{(0)t} A \vec{x}^{(0)}}{\vec{x}^{(0)t} \vec{x}^{(0)}}$$

    这是利用如果 $\vec{x}$ 就是特征向量的话，那么这个公式将求出与之对应的特征值。因此我们预期利用该公式可以求出特征值的较好的近似。
  - $Aitkan - \Delta^2$ 加速。

deflation method

- 基本思想：求其他非最大特征值的另一种方法，即构造新的矩阵将原来矩阵的最大特征值变成 0 。每次求出一个特征值后，就将这个特征值变成 0 ，反复循环，直到求出所有的特征值。

- 理论细节：设 A 有特征值

$$\lambda_1, \lambda_2, \ldots, \lambda_n$$

  与之对应的特征向量是

$$\vec{v}^{(1)}, \vec{v}^{(2)}, \vec{v}^{(3)}, \ldots, \vec{v}^{(n)}$$

  并设 $\vec{x}$ 满足 $\vec{x}^t \vec{v}^{(1)} = 1$ ，$\vec{x}$ 具体取为

$$\vec{x} = \frac{1}{\lambda_1 \vec{v}_i^{(1)}} (a_{i1}, a_{i2}, \ldots, a_{in})^t$$

  其中 i 是 1 到 n 中的任意一个值，a_{ij} 是矩阵 A 的 (i,j) 元。考虑矩阵 $B = A - \lambda_1 \vec{v}^{(1)} \vec{x}^t$ ，则该矩阵有特征值

$$0, \lambda_2, \lambda_3, \ldots, \lambda_n$$

  对应的特征向量是

$$\vec{v}^{(1)}, \vec{w}^{(2)}, \vec{w}^{(3)}, \ldots, \vec{w}^{(n)}$$

  并且 $\vec{v}^{(i)}$ 和 $\vec{w}^{(i)}$ 有关系

$$\vec{v}^{(i)} = (\lambda_i - \lambda_1) \vec{w}^{(i)} + \lambda_1 (\vec{x}^t \vec{w}^{(i)}) \vec{v}^{(1)}$$

- 具体过程：
  1. 求出最大的特征值和特征向量
  2. 按照公式求出新的矩阵 B ，其中 B 的第 i 行应该为 0（由 $\vec{x}$ 的选取方式导致）；
  3. 去掉 B 中的第 i 行和第 i 列后，再次利用幂方法求出此时的最大特征值，和（在第 i 位补零后）对应的特征向量 $\vec{w}^{(2)}$ 。利用 $\vec{w}$ 和 $\vec{v}$ 的关系可以求出实际的特征向量。

Householder's method

- 基本思想：复数域上，对称矩阵一定正交相似于一个对角矩阵。但直接求出与对称矩阵相似的对角矩阵很困难，于是考虑求出与对称矩阵正交相似的三对角矩阵。

- 理论细节：选取变换矩阵为

$$P = I - 2\vec{w}\vec{w}^t$$

  其中 $\vec{w}$ 满足

$$\vec{w}^t \vec{w} = 1$$

  然后只需要找出适当的 $\vec{w}$ 使得 $P^{-1}AP$ 为下三角矩阵即可。注意到，在要求 $\vec{w}$ 满足 $\vec{w}^t \vec{w} = 1$ 时，矩阵 P 是正交且对称的矩阵。先考虑求出一个向量 $\vec{w}^{(1)}$ 使得所得矩阵的第一列满足条件

$$a_{11}^{(1)} = a_{11}^{(0)},\ a_{21}^{(1)} = \alpha,\ a_{j1}^{(1)} = 0\ j > 2$$

  事实上现在有 $\vec{w}_i,\ i = 1, 2, \ldots, n$ 和 α 这 $n+1$ 个变量，而有 $a_{i1}^{(1)},\ i = 1, 2, \ldots, n$ ，$||\vec{w}||_2 = 1$ 和 P 正交这 $n+1$ 个条件。于是我们可以求出：

$$\alpha = -sgn(a_{21}^{(0)})(\sum_{j=2}^{n} a_{j1}^{(0)2})^{\frac{1}{2}}$$

$$\vec{w}_1^{(1)} = 0,\ \vec{w}_2^{(1)} = \frac{a_{21}^{(0)} - \alpha}{2r},\ \vec{w}_j^{(1)} = \frac{a_{j1}^{(0)}}{2r}\ j = 3, 4, \dots, n$$

其中

$$r = (\frac{1}{2}\alpha^2 - \frac{1}{2}a_{21}^{(0)}\alpha)^{\frac{1}{2}}$$

此时我们求出了 $\vec{w}^{(1)}$ 也就求出了 $P^{(1)}$ 然后可以得到 $A^{(1)} = P^{(1)}AP^{(1)}$ 为一个第一列只有第一二行有元素的新矩阵。重复这个过程，我们可以得到:

$$\alpha = -sgn(a_{k+1,k}^{(k-1)})(\sum_{j=k+1}^{n} a_{jk}^{(k-1)2})^{\frac{1}{2}}$$

$$\vec{w}_i^{(k)} = 0\ i = 1, 2, \dots, k,\ \vec{w}_{k+1}^{(k)} = \frac{a_{k+1,k}^{(k-1)} - \alpha}{2r},\ \vec{w}_j^{(k)} = \frac{a_{jk}^{(k-1)}}{2r}\ j = k+2, k+3, \dots, n$$

其中

$$r = (\frac{1}{2}\alpha^2 - \frac{1}{2}a_{k+1,k}^{(k-1)}\alpha)^{\frac{1}{2}}$$

最终我们会得到三对角的对称矩阵 $A^{(n-2)} = P^{(n-2)}P^{(n-3)}\cdots P^{(1)}AP^{(1)}\cdots P^{(n-3)}P^{(n-2)}$

QR algorithm

- 基本思想，利用 QR 分解，将三对角矩阵分解为正交矩阵和上三角矩阵的乘积

$$A^{(1)} = Q^{(1)}R^{(1)}$$

再令

$$A^{(2)} = R^{(1)}Q^{(1)}$$

则有

$$A^{(2)} = (Q^{(1)})^{-1}A^{(1)}Q^{(1)}$$

所以 $A^{(2)}$ 与 $A^{(1)}$ 有相同的特征值。而 $A^{(n)}$ 随着迭代的进行会逐渐趋向于对角矩阵（需要理论证明，此处略），于是特征值可求。

- 具体实现中，采取一系列旋转矩阵的乘积作为变换矩阵将三对角矩阵 A 转化为上三角矩阵。旋转矩阵的定义为:

$$p_{ii} = p_{jj} = cos\theta\ \ and\ \ p_{ij} = -p_{ji} = sin\theta$$

除了第 i,j 行和 i,j 列以外的地方都和单位矩阵相同，i 与 j 不相同。

- 实现细节：首先取

$$p_{11} = p_{22} = cos\theta_2\ \ and\ \ p_{12} = -p_{21} = sin\theta_2$$

使得 A 的第一列的第二个元素（设为b_2，同时将对角线上元素设为 a_1）变成0。则直接暴力带入计算可得

$$sin\theta_2 = \frac{b_2}{\sqrt{b_2^2 + a_1^2}}\ \text{ and }\ cos\theta_2 = \frac{a_1}{\sqrt{b_2^2 + a_1^2}}$$

重复该过程，即可得到

$$sin\theta_{k+1} = \frac{b_{k+1}}{\sqrt{b_{k+1}^2 + x_k^2}}\ \text{ and }\ cos\theta_{k+1} = \frac{x_k}{\sqrt{b_{k+1}^2 + x_k^2}}$$

其中 x_k 是第 k 次迭代时的第 k 个对角元。因为其值可能在之前的计算过程中被改变，所以不用 a_k 来表示。此时我们得到了一系列旋转矩阵

$$P^{(2)}, P^{(3)}, \dots, P^{(n)}$$

它们满足

$$P^{(n)}P^{(n-1)}\cdots P^{(2)}A = R$$

所求的 Q 即为

$$Q = (P^{(2)})^t (P^{(3)})^t \cdots (P^{(n)})^t$$

- 加速方法，类似于逆幂方法的想法：令

$$A^{(i)} - \sigma I = Q^{(i)} R^{(i)}$$

而

$$A^{(i+1)} = R^{(i)} Q^{(i)} + \sigma I$$

只要 σ 足够靠近特征值，就可以加快收敛速度。实际计算中，通常从第 n 行开始，取对角线上的元素和上下对角线上的元素组成的 2×2 矩阵直接计算特征值作为 σ 进行计算，当 $A(n, n-1)$ 已经足够小后，$A(n, n)$ 作为合适的特征值估计值，然后去掉第 n 行，重复计算。

Singular value Decomposation

- 基本思想：对任意一个 $n \times n$ 的矩阵 A ，都能找出对应的正交矩阵 $U_{m\times m}$ 和 $V_{n\times n}$ 以及一个只在主对角线上存在元素的矩阵 $D_{m\times n}$ st $A = UDV^T$。

- 如何求 U，D，V ?
  - 求 D: 求解 $A^t A$ 的特征值放在 D 的主对角线上即可。
  - 求 V: 求解 $A^t A$ 的特征向量并正交化作为列向量即可。
  - 求 U: 两种方法
    - 求解 AA^t 的特征向量并正交化作为列向量即可。
    - 令

$$\vec{u}_i = \frac{1}{s_i} A \vec{v}_i \quad i = 1, 2, \ldots, k$$

      其中 s_i 是 D 的主对角线上的元素，k 为 $A^t A$ 的非 0 特征值的个数。然后再随便选取与这 k 个向量线性无关的正交化的 $n-k$ 个向量作为剩下的列向量即可。

- 应用:
  - 离散数据点最小二乘法拟合的另一种计算途径。
  - 数据压缩: 只保留 U 中的前 k 列，D 的 $k \times k$ 的主子式，和 V 的前 k 行即可。k 是 $A^t A$ 的不为 0 的特征值的数目。
  - 数据去噪声