2021/5/10

参考书 numerical algorithm AKPETERS

数值中的误差分析

比如
double A = 1. / 3.0
比较 A * 3.0 == 1 实际上是不会相等的，所以我们还是要选择一个计算精度 epsilon，只要小于这个进度，我们就认为是相等的。
“数学上看起来是对的，但是程序实现的时候数值过程就不一定正确。”
$\frac{1}{3} = 0.01010101\ldots_2$，计算机中不会无限表示，所以就会有截断误差。

我们使用 K 位整数，l 为小数来表示固定位数的浮点数。这样可以重用整数运算来表示浮点数的运算。

还有一个问题就是浮点数的乘除法很容易改变小数的位数。

假如我们想表现无限精度的有理数，有理数可以表示为 p/q。我们可以指记录 p 和 q，在运算的时候，直接进行通分 $a/b + c/d = ad + cb/bd$

1. 如果不约分，一个数字可能有多个形式
2. 分子分母量级的指数膨胀

还有一个更通用的方法是，我们在运算的时候同时存储结果和丢失的精度（多存一个误差范围）。最终我们得到的值是 $a \pm \varepsilon$

加法的时候：$(x \pm \varepsilon_1) + (y \pm \varepsilon_2) = (x+y) \pm (\varepsilon_1 + \varepsilon_2 + error(x+y))$

可以通过 c++中操作符的重载来实现它。

在数值计算中产生误差的类型
1. 截断误差（计算机中的精度有限）
2. 离散化误差（本来数字是连续信号，用计算机中的离散值来表示）
3. 建模误差（对应用的问题，模型本身和实际的物理模型不一致）
4. 输入误差（获得的数据可能是观测、可能是数字化得到的，这个过程中也会产生误差）

Eg：一个财务仿真器（描述股市运行），上面的这四个误差都会有。（输入的情报不准确、模型和真实模型不一样、1 和 2 也有）

绝对误差/相对误差
绝对误差 $2cm \pm 0.02cm$
相对误差 $2cm \pm 1\%$

例子：找一个方程 $f(x^*) = 0$ 的根

我们可以得到 x_{est}， $where \left| f(x_{est}) \right| << 1$

逆向误差（backward error）

比如有 $Ax = b$，这样求出来一个 x_{est}，有 $Ax_{est} = b'$，如果 b 改了一点改到 b′，x 也只变了一点到 xest，这就是小的逆向误差。
好的情况：

$$small\ backward\ error \Rightarrow small\ \ forward\ error$$

可以认为是对一个数值稍微扰动一下，对结果的影响不大。这就是一个不敏感的系统。
差的情况（敏感的数值系统）：
稍微扰动一下，值就乱套了。

条件数（condition number）：ration of forward to backward error

Eg:

ax = b，我们可以算到一个解 $x_0 = b/a$

这个式子的正向误差是 $forward\ error = \left| x - x_0 \right|$

$$backward\ error = \left| b - b' \right| = \left| ax - ax_0 \right| = \left| a(x - x_0) \right|$$

$$condition\ number = \frac{foward\ error}{backward\ error} = \frac{1}{\left| a \right|}$$

条件数越小的时候，系统就是 well conditioned

Eg.找根问题，求出使得 $f : R \to R$， $f(x) = 0$ 的 x^*。求此问题的条件数。

我们对原函数泰勒展开， $f(x + \varepsilon) \approx f(x) + \varepsilon f'(x)$

$$forward\ error = \left| (x + \varepsilon) - x \right|$$

$$backward\ error = \left| f(x + \varepsilon) - f(x) \right|$$

$$condition\ number == \frac{|\varepsilon|}{|\varepsilon f'(x)|} = \frac{1}{|f'(x)|}$$

这个问题有什么用处呢？虽然我们并不知道真正的 x 是多少，如果我们检测到了 x 附件的 f'(x)的范围，我们就能够得到这个系统是否稳定、是否敏感。

Eg：计算 $\|\vec{x}\|_2$

如果 xi 数量级差别很大，小的东西容易被丢掉，误差会累计。
提升，所有的 xi 全部用最大值归一化到 0 和 1 之间，最后计算完再取消掉归一化。

Eg：累加问题
n 特别大的时候，最后加的时候数字就被淹没掉了。
实现 1：Kahan 算法，跟踪误差

```
function SIMPLE-SUM(x⃗)
    s ← 0                                   ▷ Current total
    for i ← 1, 2, ..., n : s ← s + x_i
    return s
```

(a)

```
function KAHAN-SUM(x⃗)
    s, c ← 0                                ▷ Current total and compensation
    for i ← 1, 2, ..., n
        v ← x_i + c                         ▷ Try to add x_i and compensation c to the sum
        s_next ← s + v                      ▷ Compute the summation result of this iteration
        c ← v − (s_next − s)                ▷ Compute compensation using the Kahan error estimate
        s ← s_next                          ▷ Update sum
    return s
```

(b)

Figure 2.3 (a) A simplistic method for summing the elements of a vector $\vec{x}$; (b) the Kahan summation algorithm.

线性系统和 LU 分解

$A\vec{x} = \vec{b}$

分成三种情况
1. 完全可解
2. 无解（over determined）
3. 无数解（under determined）

这个线性方程的可解性主要取决于 A，其实也会取决于 b

两个性质:

没有一个宽的矩阵的线性系统会有唯一解。其实很好理解，约束小于变量的个数，要么约束矛盾无解，约束不矛盾必然无穷多解。

每个高的矩阵 A，一定存在一个 $\vec{b}_0$ 使得 $A\vec{x}=\vec{b}_0$ 无解。

矩阵 A 的列空间为 y=Ax

No wide matrix system admits a unique solution.

When A is "tall," that is, when it has more rows than columns ($m > n$), then its n columns cannot possibly span the larger-dimensional $\mathbb{R}^m$. For this reason, there exists some vector $\vec{b}_0 \in \mathbb{R}^m \backslash \operatorname{col} A$. By definition, this $\vec{b}_0$ cannot satisfy $A\vec{x}=\vec{b}_0$ for *any* $\vec{x}$. That is:

For every tall matrix A, there exists a $\vec{b}_0$ such that $A\vec{x}=\vec{b}_0$ is not solvable.

在数学中span是扩张空间的意思。

就是若干个向量通过线性组合得到的一个向量空间（满足向量空间的所有要求）。Span列向量是矩阵中所有的列span成的空间。

S为一向量空间V（附于体F）的子集合。所有S的线性组合构成的集合，称为S所张成的空间，记作span(S)。

$span\ (A)=R(A)$；生成子空间=矩阵A的列空间（非齐次线性方程组$y=Ax$的值域）；

$Ker(A)=N(A)$；矩阵A的核=矩阵A的零空间（其次线性方程组$Ax=0$的解）.

完毕!

矩阵论（南航）$P.19$

求逆的坏处：计算速度很慢；可能 poorly conditioned；在很多应用中矩阵是有很好的性质的（比如很稀疏），但是稀疏矩阵的逆不一定是稀疏矩阵，所以一般不建议直接求逆。

线性代数回顾

交大·线性代数·定理 2.3：任何矩阵都可以经过单纯的初等行变换化为（行）阶梯形矩阵。
交大·线性代数·定理 2.4：任何矩阵都可经过单纯的初等行变换化为简化的阶梯形矩阵。
任何矩阵都可经过初等变换化为标准形矩阵。

高斯消元法

高斯消元法包含两步，1.前向替代；2.反向替代。

前向替代步骤：

$$\left(A|\vec{b}\right)=\begin{pmatrix}\times&\times&\times&\times&\times\\\times&\times&\times&\times&\times\\\times&\times&\times&\times&\times\\\times&\times&\times&\times&\times\end{pmatrix}\rightarrow\begin{pmatrix}1&\times&\times&\times&\times\\0&1&\times&\times&\times\\0&0&1&\times&\times\\0&0&0&1&\times\end{pmatrix}$$

反向替代步骤：

$$\left(A|\vec{b}\right)=\begin{pmatrix}1&\times&\times&\times&\times\\0&1&\times&\times&\times\\0&0&1&\times&\times\\0&0&0&1&\times\end{pmatrix}\rightarrow\begin{pmatrix}1&0&0&0&\times\\0&1&0&0&\times\\0&0&1&0&\times\\0&0&0&1&\times\end{pmatrix}$$

数值分析 P53

注意消元的时候，为了归一化，我们不能找一些很小的数作为主元，否则除出来别的数值就非常大了，我们需要找到一个避免除以很小的数字的主元，所以 partial pivoting（列里去找）和 full pivoting（行里去找）

在高斯消元中的每个行变换（乘以一个常数、两行相加减、交换两行）都需要 $O(n)$ 的复杂度。当我们选定一个主元（pivot）后，我们需要做 n 次前向替代和反向替代操作。这代表一个主元的操作都要 $O(n^2)$，所以总体上高斯消元法需要 $O(n^3)$ 的复杂度。

这样计算为 O(N^3)，有很多优化的方法。高斯消元法很多步骤依赖于 A，b 只是在右侧跟着 A 的操作去变化。

很重要的性质，三角系统可以在 O(N^2)的时间中求解。

LU 分解（lower triangle 和 upper triangle）

在很多应用场景下，我们可能要解一系列的方程：$A\vec{x}_i=\vec{b}_i$，因为求解这一系列的方程的过程中 A 是不变的，我们希望能够缓存一些 A 在做前向替代和反向替代的性质，这样就可以不用每次都做一遍重复的操作。

我们发现：求解一个上三角的矩阵方程 $U\vec{x}=\vec{c}$ 是 $O(n^2)$ 的复杂度。换句话说，如果我们能缓存 A 进行前向替代后得到的中间上三角矩阵 U，那么我们每次求解的时候可以直接使用它。我们也可以这么理解，我们需要找到一个 M，使得把复杂的 $A\vec{x}=\vec{b}$ 转化成相对来说简单的 $MA\vec{x}=M\vec{b}$ 的形式。

因为前向替代中只需要进行行变换，我们可以令 $U=M_k\cdots M_1A$，其中 M_i 为初等行变换矩阵。那么我们有 $A=(M_1^{-1}\cdots M_k^{-1})U=LU$，根据初等行变换的性质，很容易得出 L

是一个下三角矩阵。

这样我们可以用 LU 来代替 A 求解，即 $LU\vec{x}=\vec{b}$，已知 A 的 LU 分解的时候，求解一次方程就是 O(N^2)。

我们得到 A 的 LU 分解以后，我们可以使用两次求解来计算线性系统 $A\vec{x}=\vec{b}$，即 $(LU)\vec{x}=\vec{b}$ 或 $\vec{x}=U^{-1}L^{-1}\vec{b}$：

1. 求解 $L\vec{y}=\vec{b}$，即得到 $\vec{y}=L^{-1}\vec{b}$
2. 固定 $\vec{y}$，求解 $U\vec{x}=\vec{y}$

我们可以通过下式，来证明两步法求解出的 $\vec{x}$ 就是线性系统 $A\vec{x}=\vec{b}$ 的解。

$$\begin{aligned}\vec{x} &= U^{-1}\vec{y} \text{ from the second step}\\ &= U^{-1}(L^{-1}\vec{b}) \text{ from the first step}\\ &= (LU)^{-1}\vec{b} \text{ since } (AB)^{-1}=B^{-1}A^{-1}\\ &= A^{-1}\vec{b} \text{ since we factored } A=LU.\end{aligned}$$

所以给定一个 A 的 LU 分解，求解线性系统可以从高斯消去法的 $O(n^3)$ 降到 $O(n^2)$，而 LU 分解需要 $O(\frac{2}{3}n^3)$ 的时间复杂度。如果需要变换主元，我们必须让我们的分解式包含一个置换矩阵 P 来提供行、列的交换，即 $A=PLU$，由于 $P^{-1}=P^T$，所以这个小的变化并不会影响 LU 分解的便捷性。只需要在第一步时添加 $\vec{b}\leftarrow P^T\vec{b}$ 即可。

手算 LU 分解

首先使用高斯消元法的前向替代步骤得到上三角矩阵 U，即 $MA=U$。M 是一系列初等行变换的乘积，故 M 是可逆的，且我们可以计算出 $L=M^{-1}$。

例 1：

计算矩阵 $A=\begin{pmatrix}4&2&1&5\\8&7&2&10\\4&8&3&6\\6&8&4&9\end{pmatrix}$ 的 LU 分解。

方法 1：

$$A=\begin{pmatrix}4&2&1&5\\8&7&2&10\\4&8&3&6\\6&8&4&9\end{pmatrix}\xrightarrow{r_1*0.5}\begin{pmatrix}2&1&0.5&2.5\\8&7&2&10\\4&8&3&6\\6&8&4&9\end{pmatrix}\xrightarrow{r_2-4r_1,r_3-2r_1,r_4-3r_1}\begin{pmatrix}2&1&0.5&2.5\\0&3&0&0\\0&6&2&1\\0&5&2.5&1.5\end{pmatrix}$$

$$\xrightarrow{r_2*1/3}\begin{pmatrix}2&1&0.5&2.5\\0&1&0&0\\0&6&2&1\\0&5&2.5&1.5\end{pmatrix}\xrightarrow{r_3-6r_2,r_4-5r_2}\begin{pmatrix}2&1&0.5&2.5\\0&1&0&0\\0&0&2&1\\0&0&2.5&1.5\end{pmatrix}\xrightarrow{r_4-5/4r_3}\begin{pmatrix}2&1&0.5&2.5\\0&1&0&0\\0&0&2&1\\0&0&0&0.25\end{pmatrix}$$

$$\xrightarrow{r_1*0.5,r_3*0.5,r_4*4}\begin{pmatrix}1&0.5&0.25&1.25\\0&1&0&0\\0&0&1&0.5\\0&0&0&1\end{pmatrix}=U$$

那么我们可以得到矩阵 M 为

$$M=\begin{pmatrix}0.5&&&\\&1&&\\&&0.5&\\&&&4\end{pmatrix}\begin{pmatrix}1&&&\\&1&&\\&&1&\\-1.25&&&1\end{pmatrix}\begin{pmatrix}1&&&\\&1&&\\&-6&1&\\&-5&&1\end{pmatrix}\begin{pmatrix}1&&&\\&1/3&&\\&&1&\\&&&1\end{pmatrix}$$

$$\begin{pmatrix}1&&&\\-4&1&&\\-2&&1&\\-3&&&1\end{pmatrix}\begin{pmatrix}0.5&&&\\&1&&\\&&1&\\&&&1\end{pmatrix}$$

$L=M^{-1}$

这个方法能算是能算，但是需要借助大量的运算或者卡西欧 991 来辅助计算矩阵乘法。

方法 2：

暴力展开，思路参考“手算 Cholesky 分解”，

设 $A=LU=\begin{pmatrix}L_{11}&&&\\L_{21}&L_{22}&&\\L_{31}&L_{32}&L_{33}&\\L_{41}&L_{42}&L_{43}&L_{44}\end{pmatrix}\begin{pmatrix}U_{11}&U_{12}&U_{13}&U_{14}\\&U_{22}&U_{23}&U_{24}\\&&U_{33}&U_{34}\\&&&U_{44}\end{pmatrix}$

并且因为 LU 分解不唯一，我们需要添加额外条件，即 L 的对角线上的元素恒为 1.

通过（4）~（7）式可逐步进行矩阵L和U中元素的计算，如下所示：

(计算L的对角)

$$L_{00}=L_{11}=L_{22}=L_{33}=1,$$

(U的第一行)

$$U_{00}=a_{00}=4,U_{01}=a_{01}=2,U_{02}=a_{02}=1,U_{03}=a_{03}=5,$$

(L的第一列)

$$L_{10}=\frac{a_{10}}{U_{00}}=\frac{8}{4}=2,L_{20}=\frac{a_{20}}{U_{00}}=\frac{4}{4}=1,L_{30}=\frac{a_{30}}{U_{00}}=\frac{6}{4}=1.5,$$

(U的第二行)

$$U_{11}=a_{11}-L_{10}U_{01}=7-2\times 2=3,$$

$$U_{12}=a_{12}-L_{10}U_{02}=2-2\times 1=0,$$

$$U_{13}=a_{13}-L_{10}U_{03}=10-2\times 5=0,$$

(L的第二列)

$$L_{21}=\frac{1}{U_{11}}(a_{21}-L_{20}U_{01})=\frac{1}{3}\times(8-1\times 2)=2$$

$$L_{31}=\frac{1}{U_{11}}(a_{31}-L_{30}U_{01})=\frac{1}{3}\times(8-1.5\times 2)=\frac{5}{3}$$

（U的第三行）

$$U_{22}=a_{22}-L_{20}U_{02}-L_{21}U_{12}=3-1\times 1-2\times 0=2,$$

$$U_{23}=a_{23}-L_{20}U_{03}-L_{21}U_{13}=6-1\times 5-2\times 0=1,$$

（L的第三列）

$$L_{32}=\frac{1}{U_{22}}(a_{32}-L_{30}U_{02}-L_{31}U_{12})=\frac{1}{2}\times(4-1.5\times 1-\frac{5}{3}\times 0)=1.25,$$

（U的第四行）

$$U_{33}=a_{33}-L_{30}U_{03}-L_{31}U_{13}-L_{32}U_{23}=9-1.5\times 5-\frac{5}{3}\times 0-1.25\times 1=0.25;$$

经迭代计算，最后得到L和U矩阵为：

```
LU分解后L矩阵：
1.000000 0.000000 0.000000 0.000000
2.000000 1.000000 0.000000 0.000000
1.000000 2.000000 1.000000 0.000000
1.500000 1.666667 1.250000 1.000000
LU分解后U矩阵：
4.000000 2.000000 1.000000 5.000000
0.000000 3.000000 0.000000 0.000000
0.000000 0.000000 2.000000 1.000000
0.000000 0.000000 0.000000 0.250000
```

2021/5/13

今天我们讲

AX=b

上节课就讲到了高斯消元法如果直接用的话，时间复杂度为 O(n^3),LU 分解是比较通用的，今天会根据 A 的特点得到一些别的优化方法。尽可能把问题规约到 Ax=b 这个方程，就有很多很多办法可以解决。

回归方法

比如有一些实验数据

$f: R^n \to R$，树高和树的一些参数（阳光、土壤等特性）。我们可以做一些实验来采集很多组数据记录下来，希望能够利用这些数据找出一些函数。

在实际中，$f(\vec{x})$不是完全未知的，我们可以假设 fx 是一个线性函数。这就是线性回归问题。

$f(\vec{x}) = a_1x_1 + a_2x_2 + \cdots + a_nx_n = \vec{a}^T\vec{x}$，转化为了求$a_1, a_2, \cdots, a_n$的问题。

$$\vec{x}^{(k)} \mapsto y^{(k)} \equiv f\left(\vec{x}^{(k)}\right)$$

$$\begin{aligned} y^{(1)} &= f\left(\vec{x}^{(1)}\right) = a_1x_1^{(1)} + a_2x_2^{(1)} + \cdots + a_nx_n^{(1)} \\ y^{(2)} &= f\left(\vec{x}^{(2)}\right) = a_1x_1^{(2)} + a_2x_2^{(2)} + \cdots + a_nx_n^{(2)} \\ &\vdots \end{aligned}$$

如果我们有一百组数据，我们就可以列出一百个方程，列出一个方阵进行求解，（建立出了一个线性系统），比如有十个未知参数，实际上为了求解只需要十个方程就可以了。我们当然可以只选取十组数据，但是怎么把一百组的数据都使用上，之后会讲。

$$\begin{pmatrix} - & \vec{x}^{(1)\top} & - \\ - & \vec{x}^{(2)\top} & - \\ & \vdots & \\ - & \vec{x}^{(n)\top} & - \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{pmatrix}$$

这个回归方法还可以推广到非线性函数基的情况：

$$f(\vec{x}) = a_1f_1(\vec{x}) + a_2f_2(\vec{x}) + \cdots + a_mf_m(\vec{x})$$

$$\begin{pmatrix} f_1\left(\vec{x}^{(1)}\right) & f_2\left(\vec{x}^{(1)}\right) & \cdots & f_m\left(\vec{x}^{(1)}\right) \\ f_1\left(\vec{x}^{(2)}\right) & f_2\left(\vec{x}^{(2)}\right) & \cdots & f_m\left(\vec{x}^{(2)}\right) \\ \vdots & \vdots & \cdots & \vdots \\ f_1\left(\vec{x}^{(m)}\right) & f_2\left(\vec{x}^{(m)}\right) & \cdots & f_m\left(\vec{x}^{(m)}\right) \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix}$$

f can be nonlinear!

这些函数本身是非线性的。这些函数是一些基函数，可以用来表示很复杂的函数。比如傅里叶变换中的基函数就是 cos 和 sin。这样我们就可以表达更复杂的应用情况。

$$f(\vec{x}) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

"Vandermonde system"

$$f(x) = acos(x + \phi)$$

Mini-Fourier

这里举两个例子：

一个是范德蒙德系统，线性代数里有一个范德蒙德行列式。也就是每个 f_i 对应的是 x 的幂次方。

Given n pairs $x^{(k)} \mapsto y^{(k)}$, we can solve for the parameters $\vec{a}$ via the system

$$\begin{pmatrix} 1 & x^{(1)} & (x^{(1)})^2 & \cdots & (x^{(1)})^{n-1} \\ 1 & x^{(2)} & (x^{(2)})^2 & \cdots & (x^{(2)})^{n-1} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x^{(n)} & (x^{(n)})^2 & \cdots & (x^{(n)})^{n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{pmatrix}.$$

In other words, we take $f_k(x) = x^{k-1}$ in the general form above. Incidentally, the matrix on the left-hand side of this relationship is known as a Vandermonde matrix.

As an example, suppose we wish to find a parabola $y = ax^2 + bx + c$ going through $(-1, 1)$, $(0, -1)$, and $(2, 7)$. We can write the Vandermonde system in two ways:

$$\left\{ \begin{array}{rl} a(-1)^2 + b(-1) + c & = 1 \\ a(0)^2 + b(0) + c & = -1 \\ a(2)^2 + b(2) + c & = 7 \end{array} \right\} \iff \begin{pmatrix} 1 & -1 & (-1)^2 \\ 1 & 0 & 0^2 \\ 1 & 2 & 2^2 \end{pmatrix} \begin{pmatrix} c \\ b \\ a \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \\ 7 \end{pmatrix}.$$

Gaussian elimination on this system shows $(a, b, c) = (2, 0, -1)$, corresponding to the polynomial $y = 2x^2 - 1$.

另一个是傅里叶变换的形式，一堆信号可以用不同频率幅值的正弦信号叠加。即，我们可以得到一个 $f(\vec{x}) = \sum_k a_k \cos(x + \phi_k)$

相当于这实现了一个插值，但是插值也会带来一些问题。选择不同的数据得到不同的结果，并且没有考虑到误差。

1.过拟合问题，会导致把误差也拟合进去。

2.基函数的选取是很重要的，如果基函数找的不合适，那么结果也不会很好。比如用范德蒙德系统来拟合|x|效果可能就如图所示很差。

(a) Overfitting (b) Wrong basis

接近问题（Approximation）

也就是说，我们应该不去追求严格的 $A\vec{x}=\vec{b}$，而是只要接近且满足一定的条件即可，即 $A\vec{x}\approx\vec{b}$，这样的好处就是我们可以使用一百组数据，数据越多效果应该越好。

近似求解，矩阵的高和宽会影响其的可解性。高的矩阵可能会导致过定，出现无解的情况，这时候就要对过定问题进行最小二乘法。

换句话说，我们希望求 $A\vec{x}$ 和 $\vec{b}$ 的残差最小，即 $A\vec{x}\approx\vec{b}\Leftrightarrow\min_x\left\|A\vec{x}-\vec{b}\right\|_2$，可以进一步推导，得到：

$$\begin{aligned}&\min_x\left\|A\vec{x}-\vec{b}\right\|_2\\&=\min_x\left\|A\vec{x}-\vec{b}\right\|_2^2=\min_x\left(\vec{x}^TA^TA\vec{x}-2\vec{b}^TA\vec{x}+\left\|b\right\|_2^2\right)\end{aligned}$$

我们对其求梯度，可以得到

$$\begin{aligned}&2A^TA\vec{x}-2A^T\vec{b}=\vec{0}\\&\Leftrightarrow A^TA\vec{x}=A^T\vec{b}\end{aligned}$$

此时 A^TA 已经变成了方阵，称为格拉姆矩阵(gram matrix)

对于矩阵过宽的情况，解的情况是无数解，我们直接求解 $A\vec{x}=\vec{b}$ 会有各种各样的答案。所以我们可以增加一个约束，即 $\vec{x}$ 的范数尽可能小，即引入的 $\vec{x}$ 的正则项：

$$\min_x\left\|A\vec{x}-\vec{b}\right\|_2+\alpha\left\|\vec{x}\right\|_2^2$$

第二项叫做 Tikhonov 正则项。我们选择其作为惩罚项的理由是奥卡姆剃刀准则，即：在缺少 $\vec{x}$ 信息的时候，我们选择较少 entry 的 $\vec{x}$。为了最小化目标函数，我们进一步有：

$$\begin{aligned}&2A^TA\vec{x}-2A^T\vec{b}+2\alpha=\vec{0}\\&\Leftrightarrow\left(A^TA+2\alpha I_{n\times n}\right)\vec{x}=A^T\vec{b}\end{aligned}$$

- The solution $\vec{x}$ of the Tikhonov-regularized system no longer satisfies $A\vec{x}=\vec{b}$ exactly.
- When α is small, the matrix $A^\top A+\alpha I_{n\times n}$ is invertible but may be poorly conditioned. Increasing α solves this problem at the cost of less accurate solutions to $A\vec{x}=\vec{b}$.

When the columns of A span $\mathbb{R}^m$, an alternative to Tikhonov regularization is to minimize $\|\vec{x}\|_2$ with the "hard" constraint $A\vec{x}=\vec{b}$. Exercise 4.7 shows that this least-norm solution is given by $\vec{x}=A^\top(AA^\top)^{-1}\vec{b}$, a similar formula to the normal equations for least-squares.

Eg:

假如我们有以下线性系统：

$\begin{pmatrix}1 & 1\\1 & 1.00001\end{pmatrix}\vec{x}=\begin{pmatrix}1\\0.99\end{pmatrix}$，其解可以得到为 $\vec{x}=\begin{pmatrix}1001 & -1000\end{pmatrix}$。

这个解的规模是 R^2 的，比起任何原始问题的解都大。我们可以使用 Tikhonov 正则项来促进小的能够近似求解这个问题的解：

$$\left[\begin{pmatrix}1 & 1\\ 1 & 1.00001\end{pmatrix}^T\begin{pmatrix}1 & 1\\ 1 & 1.00001\end{pmatrix}+\alpha I_{2\times 2}\right]\vec{x}=\begin{pmatrix}1 & 1\\ 1 & 1.00001\end{pmatrix}^T\begin{pmatrix}1\\ 0.99\end{pmatrix}$$

$$\begin{pmatrix}2+\alpha & 2.00001\\ 2.00001 & 2.0000200001+\alpha\end{pmatrix}\vec{x}=\begin{pmatrix}1.99\\ 1.9900099\end{pmatrix}$$

对应一些 α 的解如下所示：

$$\alpha = 0.00001 \longrightarrow \vec{x} \approx (0.499998, 0.494998)$$
$$\alpha = 0.001 \longrightarrow \vec{x} \approx (0.497398, 0.497351)$$
$$\alpha = 0.1 \longrightarrow \vec{x} \approx (0.485364, 0.485366).$$

有小的正则项时，这些解接近对称近似解 $\vec{x}\approx(0.5,0.5)$，幅值很小。如果正则项的权重越来越大，那么正则项会淹没这个系统，使得 $\vec{x}\to(0,0)$。

Eg:

(a) (b) Input images with keypoints (c) Aligned images

Figure 4.3 (a) The image alignment problem attempts to find the parameters A and $\vec{b}$ of a transformation from one image of a scene to another using labeled keypoints $\vec{x}$ on the first image paired with points $\vec{y}$ on the second. As an example, keypoints marked in white on the two images in (b) are used to create (c) the aligned image.

在图像处理中，如果给定了五对特征点，那么像素之间的变换可以通过 A 和 b 来描述。

正定矩阵

在上述中，我们把问题 $A\vec{x}\approx\vec{b}$ 规约到 $\left(A^TA\right)\vec{x}=A^T\vec{b}$。无论 A 长什么样子，矩阵 A^TA 具有一系列特别的性质。

1. A^TA是对称的。

2. A^TA是半正定的矩阵。并且当且仅当 A 的列向量线性无关时，A^TA是正定矩阵。

性质 2 证明如下：

$\vec{x}^T\left(A^TA\right)\vec{x}=\left(A\vec{x}\right)^T\left(A\vec{x}\right)=\left(A\vec{x}\right)\cdot\left(A\vec{x}\right)=\left\|A\vec{x}\right\|_2^2\geq 0$

为了证明性质 2 的第二句，我们先假设 A 的列向量是线性无关的。如果 A 只是半正定的，那么会存在一个非零向量 $\vec{x}\neq\vec{0}$，使得 $\vec{x}^TA^TA\vec{x}=0$，但是这会使得 $\left\|A\vec{x}\right\|_2=0$，即 $A\vec{x}=\vec{0}$，与我们之前 A 的列向量线性无关的假设矛盾。类似的，如果 A 拥有线性相关的列向量，那么存在一个 $\vec{y}\neq\vec{0}$ 使得 $A\vec{y}=\vec{0}$，此时容易验证 A 不是正定的。

作为一个推论，当且仅当 A 的列向量线性无关时，A^TA可逆，这也提供了一种检查最小二乘问题是否达到唯一解的方法。

考虑到最小二乘系统 $A^TA\vec{x}=A^T\vec{b}$ 的盛行，我们需要考察使用特制的线性求解器来加速求解的可能性。加入我们希望求解一个对称正定系统（SPD, symmetric positive definite）$C\vec{x}=\vec{d}$。对应到最小二乘问题中，有 $C=A^TA$ 和 $\vec{d}=A^T\vec{b}$，给定了 C 的结构信息，我们可以得到更好的效果。

Cholesky 分解

我们可以把一个对称正定矩阵 $C\in R^{n\times n}$ 分解为如下所示块状的组成成分：

$C=\begin{pmatrix}c_{11} & \vec{v}^T\\ \vec{v} & \widetilde{C}\end{pmatrix}$，其中 $c_{11}\in R,\vec{v}\in R^{n-1}$，并且 $\widetilde{C}\in R^{(n-1)\times(n-1)}$。根据对称正定矩阵的性质，我们可以进行如下推导

$$\begin{aligned}&\vec{e}_1^TC\vec{e}_1>0\\&=(1\quad 0\quad\cdots\quad 0)\begin{pmatrix}c_{11} & \vec{v}^T\\ \vec{v} & \widetilde{C}\end{pmatrix}\begin{pmatrix}1\\0\\\vdots\\0\end{pmatrix}\\&=(1\quad 0\quad\cdots\quad 0)\begin{pmatrix}c_{11}\\\vec{v}\end{pmatrix}\\&=c_{11}\end{aligned}$$

第一行是大于号，我们不需要进行 pivoting，已经可以保证 $c_{11}\neq 0$ 了。我们继续进行高

斯消元法，我们需要构造一个如下所示的前向替代矩阵（forward-subsitution matrix）E：

$$E=\begin{pmatrix} 1/\sqrt{c_{11}} & \vec{0}^T \\ \vec{r} & I_{(n-1)\times(n-1)} \end{pmatrix}$$

其中的向量 $\vec{r}\in R^{n-1}$ 包括了前向替代放缩因子，满足 $r_{i-1}c_{11}=-c_{i1}$，我们把第一行的主元放缩到了 $\sqrt{c_{11}}$，原因之后再讲，所以说在前向替代之后，乘积 EC 的形式如下：

$$EC=\begin{pmatrix} 1/\sqrt{c_{11}} & \vec{0}^T \\ \vec{r} & I_{(n-1)\times(n-1)} \end{pmatrix}\begin{pmatrix} c_{11} & \vec{v}^T \\ \vec{v} & \tilde{C} \end{pmatrix}=\begin{pmatrix} \sqrt{c_{11}} & v^T/\sqrt{c_{11}} \\ \vec{0} & D \end{pmatrix}$$

其中 $D\in R^{(n-1)\times(n-1)}$。

到此，我们可以摆脱掉高斯消元法了。为了保持对称的性质，我们在右侧乘以一个 E^T，即：

$$\begin{aligned} ECE^T &= (EC)E^T \\ &= \begin{pmatrix} \sqrt{c_{11}} & v^T/\sqrt{c_{11}} \\ \vec{0} & D \end{pmatrix}\begin{pmatrix} 1/\sqrt{c_{11}} & \vec{r}^T \\ \vec{0} & I_{(n-1)\times(n-1)} \end{pmatrix} \\ &= \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & D \end{pmatrix} \end{aligned}$$

在右上角的 $\vec{0}^T$ 是因为 E 是个消去矩阵（elimination matrix）。我们消去了 C 的第一行和第一列，并且遗留下来的子矩阵 D 会在之后证明它也是对称正定的。

我们可以对 C 的所有行和列对称地重复这个消去过程。这个方法只对对称正定矩阵有用，因为对称性允许我们在两边乘以一个相同的 E；正定性保证了 $c_{11}>0$ 也就是 $1/\sqrt{c_{11}}$ 存在。

和矩阵的 LU 分解类似，我们现在可以得到一个 $C=LL^T$ 的分解，其中 L 是下三角矩阵。这个分解具体的构造方式如下，我们按照上文所述过程，多次对称地使用消去矩阵，直到最终我们得到一个单位矩阵 I，即：

$$E_k\cdots E_2E_1CE_1^TE_2^T\cdots E_k^T=I_{n\times n}$$

根据 LU 分解中所推导的那样，L 是一系列下三角矩阵的乘积：

$$L\equiv E_1^{-1}E_2^{-1}\cdots E_k^{-1}$$

我们就把 $C=LL^T$ 叫做 C 的 cholesky 分解。如果求平方根会引起数学问题，有一个相关的 LDL^T 的分解形式，其中 D 是一个对角矩阵。

Example 4.6 (Cholesky factorization, initial step)**.** As a concrete example, consider the following symmetric, positive definite matrix

$$C = \begin{pmatrix} 4 & -2 & 4 \\ -2 & 5 & -4 \\ 4 & -4 & 14 \end{pmatrix}.$$

We can eliminate the first column of C using the elimination matrix E_1 defined as:

$$E_1 = \begin{pmatrix} 1/2 & 0 & 0 \\ 1/2 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \longrightarrow E_1 C = \begin{pmatrix} 2 & -1 & 2 \\ 0 & 4 & -2 \\ 0 & -2 & 10 \end{pmatrix}.$$

We chose the upper left element of E_1 to be $1/2 = 1/\sqrt{4} = 1/\sqrt{c_{11}}$. Following the construction above, we can post-multiply by $E_1^\top$ to obtain:

$$E_1 C E_1^\top = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 4 & -2 \\ 0 & -2 & 10 \end{pmatrix}.$$

The first row *and* column of this product equal the standard basis vector $\vec{e}_1 = (1, 0, 0)$.

手算 Cholesky 分解

我们写出公式

$$A = LL^T = \begin{pmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{pmatrix} \begin{pmatrix} L_{11} & L_{21} & L_{31} \\ 0 & L_{22} & L_{32} \\ 0 & 0 & L_{33} \end{pmatrix} = \begin{pmatrix} L_{11}^2 & & (此处对称) \\ L_{21}L_{11} & L_{21}^2 + L_{22}^2 & \\ L_{31}L_{11} & L_{31}L_{21} + L_{32}L_{22} & L_{31}^2 + L_{32}^2 + L_{33}^2 \end{pmatrix}$$

我们可以得到方程：$\begin{aligned} A_{11} &= L_{11}^2 \\ A_{21} &= L_{21}L_{11} \\ A_{22} &= L_{21}^2 L_{22}^2 \\ &\vdots \end{aligned}$，即 $L = \begin{pmatrix} \sqrt{A_{11}} & 0 & 0 \\ A_{21}/L_{11} & \sqrt{A_{22} - L_{21}^2} & 0 \\ A_{31}/L_{11} & (A_{32} - L_{31}L_{21})/L_{22} & \sqrt{A_{33} - L_{31}^2 - L_{32}^2} \end{pmatrix}$

我们可以得到迭代的公式如下图所示：

$$L_{j,j} = (\pm)\sqrt{A_{j,j} - \sum_{k=1}^{j-1} L_{j,k}^2},$$

$$L_{i,j} = \frac{1}{L_{j,j}} \left(A_{i,j} - \sum_{k=1}^{j-1} L_{i,k} L_{j,k} \right) \quad \text{for } i > j.$$

在实际做题中，直接设出 L 的对应元素，乘出来然后一个个元素求解即可，根本不用背上述的迭代公式，也不需要理解 Cholesky 分解的原理，这个还是比较简单的。

2021/5/20

上节课讲到最小二乘法、LU 分解和 Cholesky 分解。

今天先要讲一个矩阵中的条件数怎么计算。对问题 $A\vec{x}=\vec{b}$ 进行少量的扰动，得到问题 $(A+\delta A)\vec{x}=\vec{b}+\delta\vec{b}$ 。我们想求解前者的问题，但是实际上可能因为各种各样的误差求解的是后者的 $\vec{x}_0$。

范数

Definition 4.2 (Vector norm)**.** A vector norm is a function $\|\cdot\|:\mathbb{R}^n\to[0,\infty)$ satisfying the following conditions:

- $\|\vec{x}\|=0$ if and only if $\vec{x}=\vec{0}$ ("$\|\cdot\|$ separates points").
- $\|c\vec{x}\|=|c|\|\vec{x}\|$ for all scalars $c\in\mathbb{R}$ and vectors $\vec{x}\in\mathbb{R}^n$ ("absolute scalability").
- $\|\vec{x}+\vec{y}\|\le\|\vec{x}\|+\|\vec{y}\|$ for all $\vec{x},\vec{y}\in\mathbb{R}^n$ ("triangle inequality").

Other than $\|\cdot\|_2$, there are many examples of norms:

- The p-norm $\|\vec{x}\|_p$, for $p\ge1$, is given by

$$\|\vec{x}\|_p\equiv\left(|x_1|^p+|x_2|^p+\cdots+|x_n|^p\right)^{1/p}.$$

  Of particular importance is the 1-norm, also known as the "Manhattan" or "taxicab" norm:

$$\|\vec{x}\|_1\equiv\sum_{k=1}^{n}|x_k|.$$

  This norm receives its nickname because it represents the distance a taxicab drives between two points in a city where the roads only run north/south and east/west.

- The ∞-norm $\|\vec{x}\|_\infty$ is given by

$$\|\vec{x}\|_\infty\equiv\max(|x_1|,|x_2|,\cdots,|x_n|).$$

These norms are illustrated in Figure 4.7 by showing the "unit circle" $\{\vec{x}\in\mathbb{R}^2:\|\vec{x}\|=1\}$ for different choices of norm $\|\cdot\|$; this visualization shows that $\|\vec{v}\|_p\le\|\vec{v}\|_q$ when $p>q$.

Theorem 4.2 (Equivalence of norms on $\mathbb{R}^n$)**.** All norms on $\mathbb{R}^n$ are equivalent.

This somewhat surprising result implies that all vector norms have the same *rough* behavior, but the choice of a norm for analyzing or stating a particular problem still can make a huge difference. For instance, on $\mathbb{R}^3$ the ∞-norm considers the vector $(1000,1000,1000)$ to have the same norm as $(1000,0,0)$, whereas the 2-norm certainly is affected by the additional nonzero values.

为了求矩阵的范数，我们可以把 $A^{m\times n}$ 拉直变成 $mn\times1$ 的向量。

Since we perturb not only vectors but also matrices, we must also be able to take the norm of a matrix. The definition of a matrix norm is nothing more than Definition 4.2 with matrices in place of vectors. For this reason, we can "unroll" any matrix in $\mathbb{R}^{m\times n}$ to a vector in $\mathbb{R}^{nm}$ to adapt any vector norm to matrices. One such norm is the *Frobenius norm*

$$\|A\|_{\text{Fro}} \equiv \sqrt{\sum_{i,j} a_{ij}^2}.$$

Such adaptations of vector norms, however, are not always meaningful. In particular, norms on matrices A constructed this way may not have a clear connection to the *action* of A on vectors. Since we usually use matrices to encode linear transformations, we would prefer a norm that helps us understand what happens when A is multiplied by different vectors $\vec{x}$. With this motivation, we can define the matrix norm *induced* by a vector norm as follows:

更有意思的是由它引起的一个变换。

导出范数

矩阵 $A \in R^{m\times n}$ 由向量范数 $\|\cdot\|$ 所导出的范数（induced norm）定义如下：

$$\|A\| \equiv \max\left\{\|A\vec{x}\| : \|\vec{x}\| = 1\right\}$$

即，矩阵的导出范数是单位向量乘以 A 之后的像的最大长度。

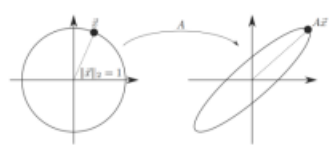


Figure 4.8 The norm $\|\cdot\|_2$ induces a matrix norm measuring the largest distortion of any point on the unit circle after applying A.

当 $\|\cdot\| = \|\cdot\|_2$ 时，如上图所示。考虑到向量范数满足性质 $\|c\vec{x}\| = |c|\|\vec{x}\|$，上述定义等价于：

$$\|A\| \equiv \max_{x\in R^n\setminus\{0\}} \frac{\|A\vec{x}\|}{\|\vec{x}\|}$$

从这个角度来说，由 $\|\cdot\|$ 导出的 A 的范数是 $A\vec{x}$ 相对于输入 $\vec{x}$ 的最大可达比例。（the largest achievable ration of the norm of $A\vec{x}$ relative to that of the input $\vec{x}$）.

使用这个最大化问题来定义这个导出范数，使得计算由 $\|\cdot\|$ 导出的 A 的范数比较困难。幸运的是，矩阵由很多著名的向量范数导出的范数可以被简化。一些已知的矩阵导出范数如下：

1-范数的导出范数：$\|A\|_1 = \max_{1\le j\le n} \sum_{i=1}^{m} |a_{ij}|$

无穷范数的导出范数：$\|A\|_\infty = \max_{1\le i\le m} \sum_{j=1}^{n} |a_{ij}|$

矩阵中的条件数

考虑到我们现在拥有工具来测量一个矩阵了，我们可以推广条件数的概念，来定义一个线性系统的条件数。

加入我们给了矩阵 A 一个扰动δA，给了 b 一个扰动$\delta\vec{b}$。对于小的ε，忽略掉可逆的技术，我们可以出一个解的函数$(A+\delta A)\vec{x}(\varepsilon)=\vec{b}+\delta\vec{b}$.

对两边同时对ε求导，并且应用求导乘法规则，我们有：

$$\delta A\cdot\vec{x}(\varepsilon)+(A+\varepsilon\cdot\delta A)\frac{d\vec{x}(\varepsilon)}{d\varepsilon}=\delta\vec{b}$$

特别地，当$\varepsilon=0$时，有$\delta A\cdot\vec{x}(0)+A\left.\frac{d\vec{x}(\varepsilon)}{d\varepsilon}\right|_{\varepsilon=0}=\delta\vec{b}$

等价地，有：$\left.\frac{d\vec{x}(\varepsilon)}{d\varepsilon}\right|_{\varepsilon=0}=A^{-1}\left(\delta\vec{b}-\delta A\cdot\vec{x}(0)\right)$

我们使用泰勒展开后，我们可以把$\vec{x}(\varepsilon)$写成$\vec{x}(\varepsilon)=\vec{x}(0)+\varepsilon\vec{x}'(0)+O(\varepsilon^2)$，此时根据定义，我们又有$\vec{x}'(0)=\left.\frac{d\vec{x}}{d\varepsilon}\right|_{\varepsilon=0}$。因此，我们可以通过求解这个扰动的系统来展开相对误差。

$$\frac{\|\vec{x}(\varepsilon)-\vec{x}(0)\|}{\|\vec{x}(0)\|}=\frac{\|\varepsilon\vec{x}'(0)+O(\varepsilon^2)\|}{\|\vec{x}(0)\|}\text{ by the Taylor expansion above}$$

$$=\frac{\|\varepsilon A^{-1}(\delta\vec{b}-\delta A\cdot\vec{x}(0))+O(\varepsilon^2)\|}{\|\vec{x}(0)\|}\text{ by the derivative we computed}$$

$$\leq\frac{|\varepsilon|}{\|\vec{x}(0)\|}(\|A^{-1}\delta\vec{b}\|+\|A^{-1}\delta A\cdot\vec{x}(0))\|)+O(\varepsilon^2)$$

$$\text{by the triangle inequality }\|A+B\|\leq\|A\|+\|B\|$$

$$\leq|\varepsilon|\|A^{-1}\|\left(\frac{\|\delta\vec{b}\|}{\|\vec{x}(0)\|}+\|\delta A\|\right)+O(\varepsilon^2)\text{ by the identity }\|AB\|\leq\|A\|\|B\|$$

$$=|\varepsilon|\|A^{-1}\|\|A\|\left(\frac{\|\delta\vec{b}\|}{\|A\|\|\vec{x}(0)\|}+\frac{\|\delta A\|}{\|A\|}\right)+O(\varepsilon^2)$$

$$\leq|\varepsilon|\|A^{-1}\|\|A\|\left(\frac{\|\delta\vec{b}\|}{\|A\vec{x}(0)\|}+\frac{\|\delta A\|}{\|A\|}\right)+O(\varepsilon^2)\text{ since }\|A\vec{x}(0)\|\leq\|A\|\|\vec{x}(0)\|$$

$$=|\varepsilon|\|A^{-1}\|\|A\|\left(\frac{\|\delta\vec{b}\|}{\|\vec{b}\|}+\frac{\|\delta A\|}{\|A\|}\right)+O(\varepsilon^2)\text{ since by definition }A\vec{x}(0)=\vec{b}.$$

到此为止，我们已经应用了一些矩阵导出范数的一些特性。

我们定义最后一个等式中的$D\equiv\frac{\|\delta\vec{b}\|}{\|\vec{b}\|}+\frac{\|\delta\vec{a}\|}{\|\vec{a}\|}$为扰动$\delta A$和$\delta\vec{b}$相对于 A 和$\vec{b}$的幅值。从

这个角度来说，我们已经把扰动这个系统 ε 的相对误差约束到了下式这个上界内，其中 $\kappa = \|A\|\|A^{-1}\|$：

$$\frac{\|\vec{x}(\varepsilon) - \vec{x}(0)\|}{\|\vec{x}(0)\|} \le |\varepsilon| \cdot D \cdot \kappa + O(\varepsilon^2)$$

因此，$\kappa = \|A\|\|A^{-1}\|$的数值约束了 A 这个线性系统的条件数。

矩阵条件数：

矩阵 $A \in R^{m \times n}$ 对于给定的矩阵范数 $\|\cdot\|$ 的条件数为 $\text{cond}A \equiv \|A\|\|A^{-1}\|$，如果 A 不可逆，那么 $\text{cond}A \equiv \infty$ 。

For nearly any matrix norm, cond $A \geq 1$ for all A. Scaling A has no effect on its condition number. Large condition numbers indicate that solutions to $A\vec{x} = \vec{b}$ are unstable under perturbations of A or $\vec{b}$.

If $\|\cdot\|$ is induced by a vector norm and A is invertible, then we have

$$\begin{aligned}
\|A^{-1}\| &= \max_{\vec{x} \neq \vec{0}} \frac{\|A^{-1}\vec{x}\|}{\|\vec{x}\|} \text{ by definition} \\
&= \max_{\vec{y} \neq \vec{0}} \frac{\|\vec{y}\|}{\|A\vec{y}\|} \text{ by substituting } \vec{y} = A^{-1}\vec{x} \\
&= \left(\min_{\vec{y} \neq \vec{0}} \frac{\|A\vec{y}\|}{\|\vec{y}\|} \right)^{-1} \text{ by taking the reciprocal.}
\end{aligned}$$

P86

列空间和 QR 分解

如之前所示，$\vec{x}$ 是 $A\vec{x} \approx \vec{b}$ 的最小二乘解的充分必要条件是 $\vec{x}$ 满足规范等式 $(A^T A)\vec{x} = A^T \vec{b}$，这个等式说明了可以用矩阵 $A^T A$ 的线性方程组来求解最小二乘问题。

限制规范等式使用的有一大问题，我们假设 A 是方阵：

$$\begin{aligned}
&\text{cond } A^T A = \|A^T A\|\|(A^T A)^{-1}\| \\
&\approx \|A^T\|\|A\|\|A^{-1}\|\|(A^T)^{-1}\| \text{ 对于大部分} \|\cdot\| \text{都成立} \\
&= \|A\|^2 \|A^{-1}\|^2 \\
&= (\text{cond } A)^2
\end{aligned}$$

也就是说，$A^T A$ 的条件数近似等于 A 的条件数的平方。因此，虽然这个最小二乘问题一般情况下是可以通过求解规范等式获得的，但是当 A 的各列是线性相关的时候，这个策略可能就会导致相当大的误差，因为它并没有直接使用 A。

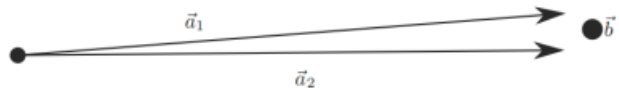


Figure 5.1 The vectors $\vec{a}_1$ and $\vec{a}_2$ nearly coincide; hence, writing $\vec{b}$ in the span of these vectors is difficult since $\vec{v}_1$ can be replaced with $\vec{v}_2$ or vice versa in a linear combination without incurring much error.

直觉上来说，A^TA 的条件数比较大的一个主要的原因是 A 的列可能比较“接近”，正如上图所示。我们把矩阵 A 的每一列考虑成一个 R^m 中的向量，如果两列满足 $\vec{a}_i \approx \vec{a}_j$，那么如果我们用 $\vec{a}_j$ 来替换掉 $\vec{a}_i$ 的话，最小二乘的残差 $\left\|\vec{b}-A\vec{x}\right\|_2$ 不会变化多少。这种相似但不完全相同的等价解会导致 poor condition（病态条件）。

为了解决这种病态条件的情况，我们使用一个替代方法，主要考虑 A 的列空间而不是高斯消元中的行空间。这个策略就是显式地处理这些列向量的方向接近的情况，这会让我们的数值计算更加稳定。

正交性

我们已经找到了为什么最小二乘法可能比较复杂的问题，我们就想解决这么避免最小二乘法成为病态条件问题的情况。如果我们可以把一个系统规约到简单的问题中，这就等于我们找到了一个解决病态问题的方法。

最简单的线性系统是 $I_{n\times n}\vec{x}=\vec{b}$，其中 I 是单位矩阵，很明显答案就是 $\vec{x}=\vec{b}$。我们甚至都用不到线性求解器来求 I 的转置，但是这种情况太稀有了，A 可能并不释放的。我们可能考虑让格拉姆矩阵 A^TA 等于单位矩阵，这会使得最小二乘问题变成一个平凡问题（trivial），为了避免混淆。我们使用 $Q^TQ=I_{n\times n}$ 来表示这样一个矩阵。

我们尝试写下 Q 的列向量 $\vec{q}_1,\cdots,\vec{q}_n\in R^m$，这样的话 Q^TQ 就有如下的结构：

$$
\begin{aligned}
Q^\top Q &= \begin{pmatrix} - & \vec{q}_1^\top & - \\ - & \vec{q}_2 & - \\ & \vdots & \\ - & \vec{q}_n & \end{pmatrix} \begin{pmatrix} | & | & & | \\ \vec{q}_1 & \vec{q}_2 & \cdots & \vec{q}_n \\ | & | & & | \end{pmatrix} \\
&= \begin{pmatrix} \vec{q}_1\cdot\vec{q}_1 & \vec{q}_1\cdot\vec{q}_2 & \cdots & \vec{q}_1\cdot\vec{q}_n \\ \vec{q}_2\cdot\vec{q}_1 & \vec{q}_2\cdot\vec{q}_2 & \cdots & \vec{q}_2\cdot\vec{q}_n \\ \vdots & \vdots & \cdots & \vdots \\ \vec{q}_n\cdot\vec{q}_1 & \vec{q}_n\cdot\vec{q}_2 & \cdots & \vec{q}_n\cdot\vec{q}_n \end{pmatrix}
\end{aligned}
$$

我们发现如果我们让 $Q^TQ=I_{n\times n}$，那么每一项有如下的形式：

$$\vec{q}_i\cdot\vec{q}_j=\begin{cases}1 & i=j\\ 0 & i\neq j\end{cases}$$

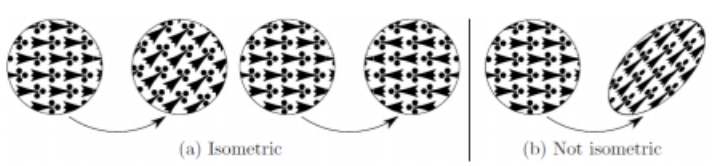
换句话说，Q 的列向量之间是正交的，并且都是单位长度的。我们把它们称为 Q 的列空间的一组正交基。

正交和正交矩阵的定义：
一个向量集合是正交的当且仅当这些向量都是单位长度，并且不同向量之间点积为 0.一个列向量全部正交的矩阵叫做一个正交矩阵。

标准正交基 $\{\vec{e}_1,\vec{e}_2,\ldots,\vec{e}_n\}$ 是一组正交基的例子，因为单位矩阵 I 的列向量就是这组正交基，所以 I 是一个正交矩阵。

现在我们知道了 $Q^TQ=I_{n\times n}$ 等价于 Q 是一个正交矩阵。而且，如果 Q 是可逆方阵且 Q 是正交的，那么有 $Q^{-1}=Q^T$，即 $Q\vec{x}=\vec{b}$ 等价于 $\vec{x}=Q^T\vec{b}$。

正交性有重要的几何解释，一个正交向量的集合是一些互相垂直的单位向量，而且 如果 Q 是正交的，说明 Q 所代表的变换不会改变向量的长度，也不会改变两个向量之间的角度。如下图所示，在 Q 的作用下，向量只会发生旋转或镜像，但是不会发送裁剪或者放缩。从一个抽象的角度来看，正交矩阵的线性代数很简单，因为它们所代表的的变化不会影响底层空间的几何特性。



非正交矩阵的策略

许多矩阵不是正交的。我们必须做一些额外的计算来把它们变成正交的。给定一个 $A\in R^{m\times n}$，我们用 col A 来表示它的列空间。col A 是 A 的所有列的生成空间。假设我们现在有一个可逆矩阵 $B\in R^{n\times n}$。

引理：列空间不变性，col A = col AB。
回顾一下消去矩阵。我们

我们回忆在高斯消元法中消去矩阵的定义：我们从一个矩阵 A 开始，不断左乘一个行操作矩阵 E_i，得到一个序列 $A,E_1A,E_2E_1A,\ldots$，最终规约到一个更容易求解的三角矩阵系统。上述的方法给出了求解最小二乘问题的另一种方法：我们对 A 进行列操作，也就是不断在右乘列操作矩阵，最终使得列都是正交的。因为这些操作是可逆的，所以最终得到的矩

阵的列空间和原先 A 的列空间相同。

换句话说，我们希望找到一个乘积 $Q = AE_1E_2\cdots E_k$，从 A 开始，不断地右乘可逆的列操作矩阵（operation matrix），直到 Q 是正交矩阵。正如之前提到的 $\mathrm{col}Q = \mathrm{col}A$，反转上述操作，我们得到了 $A = QR$，其中 $R = E_k^{-1}E_{k-1}^{-1}\cdots E_1^{-1}$。矩阵 Q 的列包含了 A 的列空间的正交基，通过一些精巧的构造，我们可以使得 R 是上三角矩阵。

当 A=QR 时，因为 Q 是正交矩阵，我们有 $A^TA = R^TQ^TQR = R^TR$。我们把它代入原式 $(A^TA)\vec{x} = A^T\vec{b}$，有：$(R^TR)\vec{x} = R^TQ^T\vec{b}$，或者等价地有 $R\vec{x} = Q^T\vec{b}$。如果我们把 R 设计成一个三角矩阵，那么我们就可以快速地对 $R\vec{x} = Q^T\vec{b}$ 使用反向替代法求解最小二乘系统 $(A^TA)\vec{x} = A^T\vec{b}$。

Reduced QR 分解

我们继续回到原先的最小二乘问题，即 $A\vec{x} \approx \vec{b}$，但 $A \in R^{m\times n}$ 不是一个方阵。之前我们讨论过的两种方法都可以把非方阵的 A 分解为 QR，但是 Q 和 R 的大小会因为方法的不同而有所不同。

·当我们使用格拉姆施密特（Gram-Schmidt）方法时，我们对 A 的列向量做正交化，从而得到 Q。所以 Q 和 A 的维数相同，即 $Q \in \mathrm{R}^{m\times n}, R \in \mathrm{R}^{n\times n}$。

格拉姆-施密特正交化法：

设 $\vec{\alpha}_1, \vec{\alpha}_2, \cdots, \vec{\alpha}_m$ 是欧氏空间中一组线性无关的向量。

（1）正交化

取 $\vec{\beta}_1 = \vec{\alpha}_1$；取 $\vec{\beta}_2 = \vec{\alpha}_2 - k\vec{\beta}_1$，其中 k 是待定常数。我们令 $\langle \vec{\beta}_1, \vec{\beta}_2 \rangle = 0$，那么有 $\langle \vec{\beta}_1, \vec{\alpha}_2 \rangle = k\langle \vec{\beta}_1, \vec{\beta}_1 \rangle$，从而 $k = \dfrac{\langle \vec{\beta}_1, \vec{\alpha}_2 \rangle}{\langle \vec{\beta}_1, \vec{\beta}_1 \rangle}$，故有 $\vec{\beta}_2 = \vec{\alpha}_2 - \dfrac{\langle \vec{\beta}_1, \vec{\alpha}_2 \rangle}{\langle \vec{\beta}_1, \vec{\beta}_1 \rangle}\vec{\beta}_1$。换句话说，就是 $\vec{\alpha}_2$ 在 $\vec{\beta}_1$ 上作了一根垂线，我们得到了此垂线的表达式为 $\vec{\beta}_2$。

推而广之，我们有 $\vec{\beta}_i = \vec{\alpha}_i - \dfrac{\langle \vec{\beta}_1, \vec{\alpha}_i \rangle}{\langle \vec{\beta}_1, \vec{\beta}_1 \rangle}\vec{\beta}_1 - \dfrac{\langle \vec{\beta}_2, \vec{\alpha}_i \rangle}{\langle \vec{\beta}_2, \vec{\beta}_2 \rangle}\vec{\beta}_2 - \cdots - \dfrac{\langle \vec{\beta}_{i-1}, \vec{\alpha}_i \rangle}{\langle \vec{\beta}_{i-1}, \vec{\beta}_{i-1} \rangle}\vec{\beta}_{i-1}$。这个的意思就是 $\vec{\alpha}_i$ 在我们已经得到的正交向量上分别作垂线，去除掉各自的成分，贡献出一个新的维度。

（2）单位化

令 $\vec{\varepsilon}_i = \frac{\vec{\beta}_i}{|\vec{\beta}_i|}$，那么我们就得到了一组正交的单位向量。

· 当我们使用 Householder 反射时，我们是从 $m \times m$ 的反射矩阵中得到 Q，则 $R \in \mathbb{R}^{m \times m}$。

假如，我们在最小二乘的典型例子中，有 $m >> n$。我们更喜欢使用 householder 方法，因为它的数值稳定性，但是 $m \times m$ 的矩阵 Q 可能太大以至于难以存储。为了节省空间，我们可以使用 R 的上三角矩阵结构。比如，考虑 5×3 的矩阵 R:

$$R = \begin{pmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

R 任何上三角之下的元素都是 0，满足下式：

$$A = QR = (Q_1, Q_2)\begin{pmatrix} R_1 \\ 0 \end{pmatrix} = Q_1 R_1$$

此时，$Q_1 \in R^{m \times n}$ 和 $R_1 \in \mathbb{R}^{n \times n}$，依旧保持了 R 的上三角。这叫做 A 的 reduced QR 分解，因为 Q_1 的列包含了 A 的列空间的基，而不是整个 R^m 空间。

Reduced QR分解同样可以求解over-determined线性最小二乘问题。形式类似Full QR分解：

$$\hat{R}x = \hat{Q}^T b \qquad (6)$$

其中 $\hat{R}x \in \mathbb{R}^{n \times 1}$， $\hat{Q}^T b \in \mathbb{R}^{n \times 1}$。

手算 QR 分解

例 1：求矩阵 $A = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ 的 QR 分解。

解：

容易判断出 $A \in R_3^{4 \times 3}$，即 A 是一个列满秩矩阵。按照定理的证明过程，

1. 将 $A = (\alpha_1 \quad \alpha_2 \quad \alpha_3)$ 的三个列向量正交化和单位化得到一个正交向量组。

$$\beta_1=\alpha_1=(1\quad 1\quad 0\quad 0)^T$$

$$\beta_2=\alpha_2-\frac{(\alpha_2,\beta_1)}{(\beta_1,\beta_1)}\beta_1=\alpha_2-\frac{1}{2}\beta_1=(0.5\quad -0.5\quad 1\quad 0)^T$$

$$\beta_3=\alpha_3-\frac{(\alpha_3,\beta_1)}{(\beta_1,\beta_1)}\beta_1-\frac{(\alpha_3,\beta_2)}{(\beta_2,\beta_2)}\beta_2=\alpha_3+\frac{1}{2}\beta_1+\frac{1}{3}\beta_2=(-1/3\quad 1/3\quad 1/3\quad 1)^T$$

然后我们进一步把 β_i 单位化，得到一组标准正交向量组：

$$\eta_1=\frac{1}{\|\beta_1\|}\beta_1=\left(\frac{\sqrt{2}}{2}\quad \frac{\sqrt{2}}{2}\quad 0\quad 0\right)^T$$

$$\eta_2=\frac{1}{\|\beta_2\|}\beta_2=\left(\frac{\sqrt{6}}{6}\quad \frac{-\sqrt{6}}{6}\quad \frac{\sqrt{6}}{3}\quad 0\right)^T$$

$$\eta_3=\frac{1}{\|\beta_3\|}\beta_3=\left(\frac{-\sqrt{3}}{6}\quad \frac{\sqrt{3}}{6}\quad \frac{\sqrt{3}}{6}\quad \frac{\sqrt{3}}{2}\right)^T$$

这样，原来的向量组与标准正交向量组之间的关系可以表示成：

$$\alpha_1=\sqrt{2}\eta_1$$

$$\alpha_2=\frac{\sqrt{6}}{2}\eta_2+\frac{\sqrt{2}}{2}\eta_1$$

$$\alpha_3=\frac{2\sqrt{3}}{3}\eta_3-\frac{\sqrt{6}}{6}\eta_2-\frac{\sqrt{2}}{2}\eta_1$$

将上式矩阵化，即为

$$A=(\alpha_1\quad \alpha_2\quad \alpha_3)=(\eta_1\quad \eta_2\quad \eta_3)\begin{pmatrix}\sqrt{2} & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}\\ & \frac{\sqrt{6}}{2} & \frac{\sqrt{6}}{6}\\ & & \frac{2\sqrt{3}}{3}\end{pmatrix}$$

$$=\begin{pmatrix}\frac{\sqrt{2}}{2} & \frac{\sqrt{6}}{6} & -\frac{\sqrt{3}}{6}\\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{6}}{6} & \frac{\sqrt{3}}{6}\\ 0 & \frac{\sqrt{6}}{3} & \frac{\sqrt{3}}{6}\\ 0 & 0 & \frac{\sqrt{3}}{2}\end{pmatrix}\begin{pmatrix}\sqrt{2} & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}\\ & \frac{\sqrt{6}}{2} & \frac{\sqrt{6}}{6}\\ & & \frac{2\sqrt{3}}{3}\end{pmatrix}=QR$$

例 2：

例2: 已知矩阵 $A=\begin{pmatrix}0&3&1\\0&4&-2\\2&1&1\end{pmatrix}$, 利用Householder变换求A的QR分解

因为$\alpha_1=(0,0,2)^T$, 记 $a_1=\|\alpha_1\|_2=2$, 令

$\omega_1=\dfrac{\alpha_1-a_1e_1}{\|\alpha_1-a_1e_1\|_2}=\dfrac{1}{\sqrt{2}}(-1,0,1)^T$ 则 $H_1=I-2\omega_1\omega_1^H=\begin{pmatrix}0&0&1\\0&1&0\\1&0&0\end{pmatrix}$,

从而 $H_1A=\begin{pmatrix}2&1&2\\0&4&-2\\0&3&1\end{pmatrix}$ 记 $\beta=(4,3)^T$, 则 $b_2=\|\beta_2\|_2=5$,

令 $\mu_1=\dfrac{\beta_2-b_2e_2}{\|\beta_2-b_2e_2\|_2}=\dfrac{1}{\sqrt{10}}(-1,3)^T$, $\tilde{H}_2=I-2\mu_2\mu_2^H=\dfrac{1}{5}\begin{pmatrix}4&3\\3&-4\end{pmatrix}$,

记 $H_2=\begin{pmatrix}1&0^T\\0&\tilde{H}_2\end{pmatrix}=\begin{pmatrix}1&0&0\\0&4&3\\0&3&-4\end{pmatrix}$, 则

$H_2(H_1A)=\begin{pmatrix}2&1&2\\0&5&-1\\0&0&-2\end{pmatrix}=R$ 取 $Q=H_1H_2=\dfrac{1}{5}\begin{pmatrix}0&3&-4\\0&4&3\\5&0&0\end{pmatrix}$

则 $A=QR$

说明: 1、利用Householder变换进行QR分解, 即使A不是列满秩矩阵也可进行, 但此时R是奇异矩阵;

2、设$A\in C^{m\times n}$, 则也有相应的QR分解;

2、QR分解在求解线性方程组最小二乘问题中有重要应用。见P121。

2021/5/24

我们现在把注意力放在一个非线性的矩阵问题:找到矩阵的特征值和特征向量。矩阵 A

的特征向量 $\vec{x}$ 及其对应的特征值 λ 是由方程 $A\vec{x}=\lambda\vec{x}$ 所确定的。有很多方法来证明这个问题是非线性的。比如说，有一个未知数的乘积的形式 $\lambda\vec{x}$。以及为了避免平凡解 $\vec{x}=\vec{0}$， 我们约束 $\|\vec{x}\|_2=1$ 这个约束保证了 $\vec{x}$ 在单位圆上，不是一个向量空间。因为这个结构，找到特征空间的算法和求解并分析线性系统等式的方法就完全不一样了。

我们先给出几个能归到这个求解特征值和特征向量问题的一些例子。

Eg：最小化投影误差的问题，找到一个低维的平面进行投影，最小化投影误差的平方，这是主成分分析法的 2D 版本。

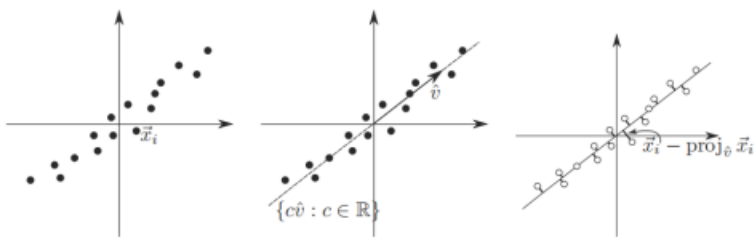


Figure 6.1 (a) A dataset with correlation between the horizontal and vertical axes; (b) we seek the unit vector $\hat{v}$ such that all data points are well-approximated by some point along span $\{\hat{v}\}$; (c) to find $\hat{v}$, we can minimize the sum of squared residual norms $\sum_i \|\vec{x}_i - \text{proj}_{\hat{v}}\,\vec{x}_i\|_2^2$ with the constraint that $\|\hat{v}\|_2 = 1$.

当 A 是对称矩阵，A 的特征向量是函数 $\vec{x}^T A\vec{x}$ 在 $\|\vec{x}\|_2=1$ 约束下的驻点。

Eg2:
比如说我们现在能收集到

假设存在一个一维的空间可以近似我们的数据集，我们希望存在一个向量 $\vec{v}$ 使得 $\vec{x}_i \approx c_i\vec{v}$，对于某个 $c_i \in R$。也就是把我们原先的四维数据 $\vec{x}_i \in \mathrm{R}^4$ 降维到 $c_i \in \mathrm{R}$。在之前，我们已经知道了 $\vec{x}_i$ 平行于 $\vec{v}$ 的最优估计是 $\text{proj}_{\vec{v}}\vec{x}_i$，我们定义 $\hat{v} \equiv \vec{v}/\|\vec{v}\|_2$，那么有：

$$\begin{aligned}\text{proj}_{\vec{v}}\vec{x}_i &= \frac{\vec{x}_i\cdot\vec{v}}{\vec{v}\cdot\vec{v}}\vec{v} \text{ by definition}\\ &= (\vec{x}_i\cdot\hat{v})\hat{v} \text{ since } \vec{v}\cdot\vec{v}=\|\vec{v}\|_2^2.\end{aligned}$$

$\text{proj}_{\vec{v}}\vec{x}_i$ 也就是 $\vec{x}_i$ 投影到主轴 $\vec{v}$ 上所对应的点。向量 $\vec{v}$ 的模长是无所谓的，只要它非零，投影的结果都一样，我们只需要把问题限制在单位向量 $\hat{v}$ 上即可。

遵循最小二乘模型，我们可以构建出一个新的最优化问题：

$$\begin{aligned}&\text{minmize}_{\hat{v}} \sum_i \|\vec{x}_i - \text{proj}_{\hat{v}}\vec{x}_i\|_2^2\\ &\text{subject to } \|\hat{v}\|_2 = 1\end{aligned}$$

(a) Input data (b) Principal axis (c) Projection error

This problem minimizes the sum of squared differences between the data points $\vec{x}_i$ and their best approximation as a multiple of $\hat{v}$, as in Figure 6.1(c). We can simplify our optimization objective using the observations we already have made and some linear algebra:

$$\begin{aligned}
\sum_i \|\vec{x}_i - \text{proj}_{\hat{v}} \vec{x}_i\|_2^2 &= \sum_i \|\vec{x}_i - (\vec{x}_i \cdot \hat{v})\hat{v}\|_2^2 \text{ as explained above} \\
&= \sum_i \left(\|\vec{x}_i\|_2^2 - 2(\vec{x}_i \cdot \hat{v})(\vec{x}_i \cdot \hat{v}) + (\vec{x}_i \cdot \hat{v})^2 \|\hat{v}\|_2^2\right) \text{ since } \|\vec{w}\|_2^2 = \vec{w} \cdot \vec{w} \\
&= \sum_i \left(\|\vec{x}_i\|_2^2 - (\vec{x}_i \cdot \hat{v})^2\right) \text{ since } \|\hat{v}\|_2 = 1 \\
&= \text{const.} - \sum_i (\vec{x}_i \cdot \hat{v})^2 \text{ since the unknown here is } \hat{v} \\
&= \text{const.} - \|X^\top \hat{v}\|_2^2, \text{ where the columns of } X \text{ are the vectors } \vec{x}_i.
\end{aligned}$$

我们移去符号之后，可以得到如下等价问题：

$$\begin{aligned}
&\text{maximize } \left\|X^T\hat{v}\right\|_2^2 \\
&\text{subject to } \left\|\hat{v}\right\|_2^2 = 1
\end{aligned}$$

Eg：特征根问题

假设 A 是一个对称正定矩阵，也就是 $A^T = A$，且对于任意 $\vec{x} \in R^n \setminus \{\vec{0}\}$ 都有 $\vec{x}^T A\vec{x} > 0$。我们希望对给定的 $A \in R^{n\times n}$ 在满足条件 $\left\|\vec{x}\right\|_2^2 = 1$ 的条件下最小化 $\vec{x}^T A\vec{x}$。我们可以定义拉格朗日函数为：$\Lambda(\vec{x},\lambda) = \vec{x}^T A\vec{x} - \lambda\left(\left\|\vec{x}\right\|_2^2 - 1\right) = \vec{x}^T A\vec{x} - \lambda\left(\vec{x}^T\vec{x} - 1\right)$。

我们对此函数求导，取到极值时 $\vec{x}$ 应满足 $\nabla_{\vec{x}}\Lambda = 2A\vec{x} - 2\lambda\vec{x} = 0 \Leftrightarrow A\vec{x} = \lambda\vec{x}$，也就是说此时驻点 $\vec{x}$ 是矩阵 A 的特征向量：$A\vec{x} = \lambda\vec{x}$，且 $\left\|\vec{x}\right\|_2^2 = 1$。在驻点条件下，我们的目标函数满足 $\vec{x}^T A\vec{x} = \vec{x}^T\lambda\vec{x} = \lambda\left\|\vec{x}\right\|_2^2 = \lambda$。因此，在满足条件 $\left\|\vec{x}\right\|_2^2 = 1$ 的条件下最小化 $\vec{x}^T A\vec{x}$ 就是求 A 的最小特征根 λ 对应的特征向量。

我们知道 $\left\|X^T\hat{v}\right\|_2^2 = \hat{v}^T XX^T\hat{v}$，如上例所示，$\hat{v}$ 是 XX^T 的最大特征根所对应的特征向量。向量 $\hat{v}$ 就被称作这个数据集的第一主成分。

我们需要把对称矩阵推广到复数域 $C^{n\times n}$ 上。

共轭矩阵：复矩阵 $A \in C^{n\times n}$ 的共轭矩阵 $\overline{A}$ 就是每个元素都取共轭。

共轭转置（conjugate transpose）：复矩阵 $A \in C^{n\times n}$ 的共轭转置定义为 $A^H \equiv (\overline{A})^T$

厄米特矩阵（Hermitian matrix）：如果 $A = A^H$，那么 A 是厄米特矩阵。

实对称矩阵因为没有虚部，所以是厄米特矩阵。我们继续把向量点积推广到复数域的向量 $\vec{x}$ 和 $\vec{y}$ 中：$\langle \vec{x}, \vec{y} \rangle \equiv \sum x_i \bar{y}_i$，在复数域的点积中，对称性变为 $\overline{\langle \vec{v}, \vec{w} \rangle} = \langle \vec{w}, \vec{v} \rangle$.

Proposition 6.3. All eigenvalues of Hermitian matrices are real.

Proof. Suppose $A \in \mathbb{C}^{n\times n}$ is Hermitian with $A\vec{x} = \lambda\vec{x}$. By scaling, we can assume $\|\vec{x}\|_2^2 = \langle \vec{x}, \vec{x} \rangle = 1$. Then:

$$\begin{aligned}
\lambda &= \lambda\langle \vec{x}, \vec{x} \rangle \text{ since } \vec{x} \text{ has norm 1} \\
&= \langle \lambda\vec{x}, \vec{x} \rangle \text{ by linearity of } \langle \cdot, \cdot \rangle \\
&= \langle A\vec{x}, \vec{x} \rangle \text{ since } A\vec{x} = \lambda\vec{x} \\
&= (A\vec{x})^\top \vec{x} \text{ by definition of } \langle \cdot, \cdot \rangle \\
&= \vec{x}^\top \overline{(A^\top \vec{x})} \text{ by expanding the product and applying the identity } \overline{ab} = \bar{a}\bar{b} \\
&= \langle \vec{x}, A^H \vec{x} \rangle \text{ by definition of } A^H \text{ and } \langle \cdot, \cdot \rangle \\
&= \langle \vec{x}, A\vec{x} \rangle \text{ since } A = A^H \\
&= \bar{\lambda}\langle \vec{x}, \vec{x} \rangle \text{ since } A\vec{x} = \lambda\vec{x} \\
&= \bar{\lambda} \text{ since } \vec{x} \text{ has norm 1.}
\end{aligned}$$

Thus $\lambda = \bar{\lambda}$, which can happen only if $\lambda \in \mathbb{R}$, as needed. □

Not only are the eigenvalues of Hermitian (and symmetric) matrices real, but also their eigenvectors must be orthogonal:

性质：厄米特矩阵不同的特征值所对应的特征向量一定是正交的。

Proposition 6.4. Eigenvectors corresponding to distinct eigenvalues of Hermitian matrices must be orthogonal.

Proof. Suppose $A \in \mathbb{C}^{n\times n}$ is Hermitian, and suppose $\lambda \neq \mu$ with $A\vec{x} = \lambda\vec{x}$ and $A\vec{y} = \mu\vec{y}$. By the previous proposition we know $\lambda, \mu \in \mathbb{R}$. Then, $\langle A\vec{x}, \vec{y} \rangle = \lambda\langle \vec{x}, \vec{y} \rangle$. But since A is Hermitian we can also write $\langle A\vec{x}, \vec{y} \rangle = \langle \vec{x}, A^H\vec{y} \rangle = \langle \vec{x}, A\vec{y} \rangle = \mu\langle \vec{x}, \vec{y} \rangle$. Thus, $\lambda\langle \vec{x}, \vec{y} \rangle = \mu\langle \vec{x}, \vec{y} \rangle$. Since $\lambda \neq \mu$, we must have $\langle \vec{x}, \vec{y} \rangle = 0$. □

Finally, we state (without proof) a crowning result of linear algebra, the Spectral Theorem. This theorem states that all symmetric or Hermitian matrices are non-defective and therefore must have exactly n orthogonal eigenvectors.

计算一个矩阵的特征值是一个研究充分的问题，有很多数学方法。每个方法都适合一个特定的情况，接近最优情况和速度需要许多技巧。以下，我们介绍一些著名的方法。

幂迭代

假设 $A \in R^{n\times n}$ 是 non-defective 且所有实特征值非零，比如 A 是对称矩阵。根据定义，A 有特征向量集合 $\vec{x}_1, \ldots, \vec{x}_n \in R^n$，我们按照对应的特征值排序 $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$.

任意一个向量 $\vec{v} \in R^n$，因为 A 的特征向量张成 R^n（生成子空间为 R^n），我们可以把 $\vec{v}$ 写成 $\vec{x}_i$ 作为基的形式，即 $\vec{v} = c_1\vec{x}_1 + \cdots + c_n\vec{x}_n$，我们两边同时左乘 A 矩阵：

$$
\begin{aligned}
A\vec{v} &= c_1 A\vec{x}_1 + \cdots + c_n A\vec{x}_n \\
&= c_1\lambda_1\vec{x}_1 + \cdots + c_n\lambda_n\vec{x}_n \text{ since } A\vec{x}_i = \lambda_i\vec{x}_i \\
&= \lambda_1\left(c_1\vec{x}_1 + \frac{\lambda_2}{\lambda_1}c_2\vec{x}_2 + \cdots + \frac{\lambda_n}{\lambda_1}c_n\vec{x}_n\right) \\
A^2\vec{v} &= \lambda_1^2\left(c_1\vec{x}_1 + \left(\frac{\lambda_2}{\lambda_1}\right)^2 c_2\vec{x}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_1}\right)^2 c_n\vec{x}_n\right) \\
&\vdots \\
A^k\vec{v} &= \lambda_1^k\left(c_1\vec{x}_1 + \left(\frac{\lambda_2}{\lambda_1}\right)^k c_2\vec{x}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_1}\right)^k c_n\vec{x}_n\right).
\end{aligned}
$$

随着 $k \to \infty$，比值 $\left(\frac{\lambda_i}{\lambda_1}\right)^k \to 0$（除了 $\lambda_i = \pm\lambda_1$ 的情况）。如果 $\vec{x}$ 是 $\vec{v}$ 在特征值 λ_1 的特征向量空间的投影上，那么因为特征值的绝对值是唯一的.

换句话说，我们构造任意选取向量 $\vec{v}_0 \in R^n$，构造向量序列满足：$\vec{v}_k = A^k v_0$，我们有

$$
\begin{aligned}
\vec{v}_k = A^k\vec{v}_0 &= c_1A^k\vec{x}_1 + \cdots + c_nA^k\vec{x}_n \\
&= c_1\lambda_1^k\vec{x}_1 + \cdots + c_n\lambda_n^k\vec{x}_n \\
&= \lambda_1^k(c_1\vec{x}_1 + \sum_{i=2}^{n} c_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\vec{x}_i) \overset{k\to\infty}{=} \lambda_1^k c_1\vec{x}_1
\end{aligned}
$$

$$
\vec{v}_{k+1} \overset{k\to\infty}{=} \lambda_1^{k+1}c_1\vec{x}_1
$$

所以，我们观察可以得到，迭代收敛时 $\vec{v}_{k+1}$ 和 $\vec{v}_k$ 的各个元素都有固定比值 λ_1，得到了主特征值 λ_1 后，我们还可以进一步计算对应 λ_1 的一个特征向量，即 $c_1\vec{x}_1 = \lim_{k\to\infty}\frac{\vec{v}_k}{\lambda_1^k}$，因为 c_1 非零，我们并不关心这个特征向量的长度是多少。

这个结论就能得到计算 A 的最大特征值 λ_1 的特征向量 $\vec{x}_1$ 的算法：

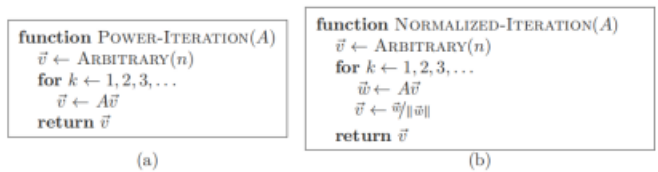
1. 取任意向量 $\vec{v}_1 \in R^n$
2. 迭代计算 $\vec{v}_k = A\vec{v}_{k-1}$，直到收敛。

幂迭代算法使得 $k \to \infty$ 的时候，$\vec{v}_k$ 越来越与 $\vec{x}_1$ 平行。

这个方法有一个会失效的情况，也就是我们随机选的 $\vec{v}_1$ 使得 $c_1 = 0$，但是这个概率非常小。同样地，幂迭代在有最大特征值的绝对值 λ 和 $-\lambda$ 时候也会失败。这个算法收敛的速度取决于第 2 项到第 n 项衰减的速度，这是由 A 的第二大的特征值和最大特征值的比值 $|\lambda_2|/|\lambda_1|$ 所

决定的。

如果$|\lambda_1|>1$，随着$k\to\infty$有$\|v_k\|\to\infty$，这是非常不好的。因为我们只需要关心特征向量的方向而不是模长，所以每次迭代的时候我们可以归一化一下，如下方的 b 图所示。



(a) (b)

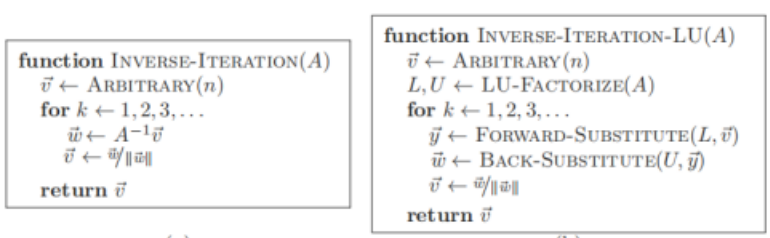
Figure 6.3 Power iteration (a) without and (b) with normalization for finding the largest eigenvalue of a matrix.

反幂迭代法（Inverse Iteration）

我们现在有了估计一个矩阵的最大特征值λ_1的迭代算法。我们假定 A 是可逆的，所以我们可以通过求解$A\vec{y}=\vec{v}$，来得到$\vec{y}=A^{-1}\vec{v}$.

我们使用之前章节的技巧，如果有$A\vec{x}=\lambda\vec{x}$，那么有$\vec{x}=\lambda A^{-1}\vec{x}$，或者等价地有$A^{-1}\vec{x}=\frac{1}{\lambda}\vec{x}$。那么，$1/\lambda$是$A^{-1}$的特征值，且对应的特征向量为$\vec{x}$。

根据基本不等式如果$|a|>|b|$，那么$|b|^{-1}>|a|^{-1}$，那么 A 的最小幅值的特征值的对应了A^{-1}的最大幅值的特征值。这个给出了找到第 n 大特征值的算法，叫做 inverse power iteration（反幂迭代法），如下图所示：



(a) (b)

Figure 6.4 Inverse iteration (a) without and (b) with LU factorization.

这个方法其实就是对A^{-1}使用幂迭代法。我们重复地在右边求解相同的方程。我们可以使用之前提到的分解技巧，比如说$A=LU$，那么我们可以我们可以构建一个等价但是更加高效的反幂迭代法（见上图右边）。有了这个简化，求解$A^{-1}\vec{v}$可以分成两步：先求解$L\vec{y}=\vec{v}$，

然后求解 $U\vec{w}=\vec{y}$。

我们假设 A 的第二大幅值的特征值为 λ_2。幂迭代法只有在$|\lambda_2/\lambda_1|$很小的时候收敛速度才很快，因为此时$|\lambda_2/\lambda_1|^k$收敛得很快。如果这个比值接近于 1，那么需要非常多的迭代才能分理出单个特征向量。

我们记矩阵 A 的特征值 $\lambda_1,\ldots,\lambda_n$，对应的特征向量为 $\vec{x}_1,\ldots,\vec{x}_n$，那么矩阵 $A-\sigma I_{n\times n}$ 的特征值为 $\lambda_1-\sigma,\ldots,\lambda_n-\sigma$，证明如下：

$$(A-\sigma I_{n\times n})\vec{x}_i=A\vec{x}_i-\sigma\vec{x}_i=\lambda\vec{x}_i-\sigma\vec{x}_i=(\lambda-\sigma)\vec{x}_i$$

我们有这个性质的情况下，我们希望选择一个 σ 使得

$$\left|\frac{\lambda_2-\sigma}{\lambda_1-\sigma}\right|<\left|\frac{\lambda_2}{\lambda_1}\right|$$

所以，我们希望求解 $A-\sigma I_{n\times n}$ 的特征向量，而不是直接去求解 A，我们要选取合适的 σ 来扩大第一个和第二个特征值的距离来改善收敛的速度。因为我们没有任何 A 的特征值的先验信息，所以选取 σ 是一个玄学。

一般来说，如果我们选取 σ 使得 σ 接近 A 的一个特征值，那么 $A-\sigma I_{n\times n}$ 有一个接近于 0 的特征值，我们可以通过反幂迭代法求得。换句话说，我们使用幂迭代法得到了 A 的一个特定的特征值，而不是最大或者最小的特征值。我们可以对 A 施加偏移，使得我们希望的特征值接近于 0，就可以通过反幂迭代法求得。

如果我们一开始猜测的 σ 不精确，我们逐步更新它。比如说，我们猜测了 A 的一个特征向量为 $\vec{x}$，那么通过最小二乘估计，对应的特征值可以通过下式得到：

$$\sigma\approx\frac{\vec{x}^T A\vec{x}}{\|\vec{x}\|_2^2}$$

这个分式叫做瑞利商（Rayleigh quotient）。我们可以通过瑞利商迭代法来加速迭代的收敛，也就是在每一步迭代中使用瑞利商来估计偏移量 σ

function RAYLEIGH-QUOTIENT-ITERATION(A,σ)
  $\vec{v}\leftarrow$ ARBITRARY(n)
  for $k\leftarrow 1,2,3,\ldots$
    $\vec{w}\leftarrow(A-\sigma I_{n\times n})^{-1}\vec{v}$
    $\vec{v}\leftarrow\vec{w}/\|\vec{w}\|$
    $\sigma\leftarrow\vec{v}^T A\vec{v}/\|\vec{v}\|_2^2$
  return $\vec{v}$

Figure 6.5 Rayleigh quotient iteration for finding an eigenvalue close to an initial guess σ.

瑞利商迭代法使用更少的步数就可以收敛，但是，因为每一步迭代中的矩阵 $A-\sigma_k I_{n\times n}$ 都是

不同的，我们没有办法像反幂迭代法中一样对 A 在循环外进行 LU 分解，从而加速计算。换句话说，虽然我们迭代次数更少了，但是迭代每次的耗时增加了。需要提醒的是，如果 σ_k 很完美，会使得矩阵 $A-\sigma_k I$ 趋近于奇异矩阵，这会导致逆迭代中出现一些问题。

SVD 分解

对于矩阵 $A\in R^{m\times n}$，我们可以考虑一个函数 $\vec{v}\to A\vec{v}$，这个函数把 n 维空间中的向量 $\vec{v}$ 映射为 m 维空间中的向量 $A\vec{v}$。从这个角度来看，我们希望度量 A 对变换前后的向量的长度影响，可以通过检测函数 $R(\vec{v})=\dfrac{\|A\vec{v}\|_2}{\|\vec{v}\|_2}$ 的驻点。

上式度量了 $\vec{v}$ 在 A 这个变换下的相对伸缩。容易检验，$\vec{v}$ 的数乘 $\alpha\vec{v}$ 在变换 A 下，效应相同。自此，我就可以限制我们的研究在 $\|\vec{v}\|_2=1$ 下，因为 $R(\vec{v})\geq 0$，所以我们希望找到 $R(\vec{v})$ 的驻点就等于希望找到 $R^2(\vec{v})=\|A\vec{v}\|_2^2=\vec{v}^TA^TA\vec{v}$ 的驻点。而正如我们在之前章节证明的那样，找到满足 $\|\vec{v}\|_2=1$ 的 $\vec{v}^TA^TA\vec{v}$ 的驻点等价于求解满足 $A^TA\vec{v}_i=\lambda_i\vec{v}_i$ 的 A^TA 的特征向量，因为我们知道 A^TA 是对称半正定矩阵，所以有 $\lambda_i\geq 0$，且对于 $i\neq j$ 有 $\vec{v}_i\cdot\vec{v}_j=0$。

根据我们对度量函数 R 的使用，我们可以使用 $\{\vec{v}_i\}$ 基来学习 A 所代表的变换的影响。我们令 $\vec{u}_i=A\vec{v}_i$，那么我们有

$$
\begin{aligned}
\lambda_i\vec{u}_i &= \lambda_i\cdot A\vec{v}_i \text{ by definition of } \vec{u}_i\\
&= A(\lambda_i\vec{v}_i)\\
&= A(A^\top A\vec{v}_i) \text{ since } \vec{v}_i \text{ is an eigenvector of } A^\top A\\
&= (AA^\top)(A\vec{v}_i) \text{ by associativity}\\
&= (AA^\top)\vec{u}_i.
\end{aligned}
$$

我们对 $\vec{u}_i$ 取范数，有 $\|\vec{u}_i\|_2=\|A\vec{v}_i\|_2=\sqrt{\|A\vec{v}_i\|_2^2}=\sqrt{\vec{v}_i^TA^TA\vec{v}_i}=\sqrt{\lambda_i}\|\vec{v}_i\|_2$，这个等式可以得出两个结论：

1. 假设 $\vec{u}_i$ 不是一个零向量，那么 $\vec{u}_i=A\vec{v}_i$ 是一个矩阵 AA^T（注意！之前所述的矩阵是 A^TA！）的对应特征向量，且满足 $\|\vec{u}_i\|_2=\sqrt{\lambda_i}\|\vec{v}_i\|_2$。
2. 否则 $\vec{u}_i$ 就是一个零向量。

一个等价的证明展示了如果 $\vec{u}$ 是 AA^T 的特征向量，那么 $\vec{v} \equiv A^T\vec{u}$ 要么是零向量要么是 A^TA 相同特征值的特征向量。

我们令 k 是严格为正的特征值的个数，即 $\lambda_i > 0, \quad for\ i \in \{1,\ldots,k\}$。根据我们上面的构造，我们可以令 $\vec{v}_1,\ldots,\vec{v}_k \in R^n$ 是 A^TA 的特征向量，我们可以得到 AA^T 的对应特征向量为 $\vec{u}_1,\ldots,\vec{u}_k \in R^m$，即：

$$A^TA\vec{v}_i = \lambda_i\vec{v}_i$$
$$AA^T\vec{u}_i = \lambda_i\vec{u}_i$$

对于这些恒正的特征值 λ_i，我们可以归一化使得 $\|\vec{u}_i\| = \|\vec{v}_i\| = 1$。我们定义矩阵 $\overline{V} \in R^{n\times k}$ 和 $\overline{U} \in R^{m\times k}$，其中它们的列分别为 $\vec{v}_i$ 和 $\vec{u}_i$。根据构造，我们发现 $\overline{V}$ 包含了 A 的行空间的一组正交基，而 $\overline{U}$ 包含了 A 的列空间的一组正交基。

我们可以在新的基矩阵下来考察 A 的作用效果。我们令 $\vec{e}_i$ 是第 i 个标准基向量，那么我们有：

$$\begin{aligned}
\overline{U}^TA\overline{V}\vec{e}_i &= \overline{U}^TA\vec{v}_i[\text{根据}\overline{V}\text{和}\vec{e}_i\text{的定义}] \\
&= \frac{1}{\lambda_i}\overline{U}^TA(\lambda_i\vec{v}_i)[\text{因为我们假设了}\ \lambda_i > 0] \\
&= \frac{1}{\lambda_i}\overline{U}^TA(A^TA\vec{v}_i)[\vec{v}_i\text{是矩阵}A^TA\text{的特征向量}] \\
&= \frac{1}{\lambda_i}\overline{U}^T(AA^T)A\vec{v}_i[\text{矩阵乘法的结合律}] \\
&= \frac{1}{\lambda_i}\overline{U}^T(AA^T)\sqrt{\lambda_i}\vec{u}_i[\text{原本是}\|\vec{u}_i\|_2 = \sqrt{\lambda_i}\|\vec{v}_i\|_2,\text{但是我们在这里归一化了,}\|\vec{u}_i\|_2 = 1] \\
&= \frac{1}{\sqrt{\lambda_i}}\overline{U}^T(AA^T\vec{u}_i) \\
&= \sqrt{\lambda_i}\overline{U}^T\vec{u}_i[AA^T\vec{u}_i = \lambda_i\vec{u}_i] \\
&= \sqrt{\lambda_i}\vec{e}_i
\end{aligned}$$

我们令 $\Sigma = \begin{bmatrix} \sqrt{\lambda_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sqrt{\lambda_k} \end{bmatrix}_{k\times k}$，我们就可以得到 $\overline{U}^TA\overline{V} = \Sigma$

我们可以通过添加正交零空间向量 $\vec{v}_i$ 和 $\vec{u}_i$，其满足 $A^TA\vec{v}_i = \vec{0}$，且 $AA^T\vec{u}_i = \vec{0}$，这样

可以把 $\overline{U}$ 和 $\overline{V}$ 扩展到 $U \in R^{m\times m}$ 和 $V \in R^{n\times n}$。在扩展之后，对于 $i > k$ 的情况，有 $U^T A\overline{e}_i = 0$。我们可以构造新的奇异值矩阵元素为：$\Sigma_{ij} \equiv \begin{cases} \sqrt{\lambda_i}, i = j \text{ and } i \le k \\ 0, otherwise \end{cases}$，那么我们可以把我们之前的关系扩展到 $U^T AV = \Sigma$，有因为 U 和 V 都是正交的，我们有：

$$A = U\Sigma V^T$$

上述这个分解就是 A 的奇异值分解。U 的列向量叫做左奇异向量，V 的列向量叫做右奇异向量。Σ 的对角线上的元素 σ_i 叫做 A 的奇异值。通常奇异值是按照递减的顺序排序的，下限为 0。U 和 V 都是正交矩阵，对应了 $\sigma_i \neq 0$ 的 U 和 V 的列向量分别生成了 A 的列空间和行空间。

SVD 对矩阵 A 所代表的变换提供了一个几何的解释。因为 U 和 V 都是正交的，它们各自所代表的变换是长度不变和角度不变的。而对角矩阵，Σ 是对各个坐标轴进行了不同程度的拉伸。因为一个矩阵的 SVD 分解约定存在，任意矩阵 $A \in R^{m\times n}$ 都可以分解成一个全等变换，一个坐标轴的拉伸，和第二个全等变换，如下图所示：

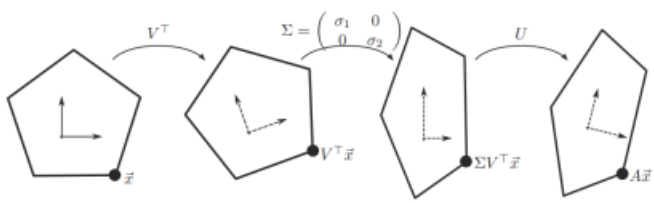


Figure 7.1 Geometric interpretation for the singular value decomposition $A = U\Sigma V^\top$. The matrices U and $V^\top$ are orthogonal and hence preserve lengths and angles. The diagonal matrix Σ scales the horizontal and vertical axes independently.

手算 SVD 分解

首先我们要回顾求解一个方程组的基础解系的方法：

例 1：

求方程组 $\begin{cases} x_1 + 2x_2 - x_3 = 0 \\ 2x_1 + 3x_2 + x_3 = 0 \\ 4x_1 + 7x_2 - x_3 = 0 \end{cases}$ 的通解

解：首先把 A 化为行最简式，

$$A = \begin{pmatrix} 1 & 2 & -1 \\ 2 & 3 & 1 \\ 4 & 7 & -1 \end{pmatrix} \underset{r_2 - 2r_1, r_3 - 4r_1}{\rightarrow} \begin{pmatrix} 1 & 2 & -1 \\ 0 & -1 & 3 \\ 0 & -1 & 3 \end{pmatrix} \underset{r_3 - r_2}{\rightarrow} \begin{pmatrix} 1 & 2 & -1 \\ 0 & -1 & 3 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 5 \\ 0 & 1 & -3 \\ 0 & 0 & 0 \end{pmatrix}$$

因为对于 $A\in R^{m\times n}$ 且满足 $r(A)=r<n$ 的情况下，我们化为行最简式后，有结论：$x_1,x_2,\ldots,x_r$ 是真未知量，而 $x_{r+1},x_{r+2},\ldots,x_n$ 是自由未知量。真未知量由自由未知量唯一确定。

故在本题中，$r(A)=2<3$，自由未知量是 x_3，我们选定 $x_3=1$，此时有 $\begin{cases}x_1=-5\\x_2=3\end{cases}$，

故基础解系为 $\xi=\begin{pmatrix}-5\\3\\1\end{pmatrix}$，通解为 $k\xi=\begin{pmatrix}-5k\\3k\\k\end{pmatrix}$

例 2：求方程组的通解 $\begin{cases}x_1-x_2-x_3+x_4=0\\x_1-x_2+x_3-3x_4=0\\x_1-x_2-2x_3+3x_4=0\end{cases}$

解：

$$A=\begin{pmatrix}1&-1&-1&1\\1&-1&1&-3\\1&-1&-2&3\end{pmatrix}\to\begin{pmatrix}1&-1&-1&1\\0&0&2&-4\\0&0&-1&2\end{pmatrix}$$

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

故同解方程组为 $\begin{cases}x_1=x_2+x_4\\x_3=2x_4\end{cases}$，因为 A 的秩为 2，我们可以选定 x_2,x_4 为自由变量，分别令 $\begin{pmatrix}x_2\\x_4\end{pmatrix}=\begin{pmatrix}1\\0\end{pmatrix},\begin{pmatrix}0\\1\end{pmatrix}$，那么我们可以得到 $\begin{pmatrix}x_1\\x_3\end{pmatrix}=\begin{pmatrix}1\\0\end{pmatrix},\begin{pmatrix}0\\2\end{pmatrix}$。

故我们有基础解系为 $\xi_1=\begin{pmatrix}1\\1\\0\\0\end{pmatrix}$，$\xi_2=\begin{pmatrix}0\\0\\2\\1\end{pmatrix}$，通解为 $k_1\xi_1+k_2\xi_2$

知道了基础解系的求法，我们就可以尝试来计算一个矩阵的特征值及其对应的特征向量了。

例 3：

求矩阵 $A=\begin{pmatrix}3&5\\5&3\end{pmatrix}$ 的特征值和特征向量。

解：

(1.求解特征值)设 A 的特征值为 λ，求解多项式方程 $|A-\lambda E|=0$

$$|A-\lambda E|=0 \Leftrightarrow \begin{vmatrix} 3-\lambda & 5 \\ 5 & 3-\lambda \end{vmatrix}=0 \Leftrightarrow \lambda_1=8, \lambda_2=2$$

故 A 的特征值矩阵为 $\Sigma=\begin{bmatrix} 8 & 0 \\ 0 & -2 \end{bmatrix}$

(2. 求解特征向量)通过式 $(A-\lambda E)\vec{x}=0$ 求解特征向量：

$(A-\lambda_1 E)\vec{x}_1=0 \Rightarrow \begin{pmatrix} -5 & 5 \\ 5 & -5 \end{pmatrix}\vec{x}_1=0 \Rightarrow \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}\vec{x}_1=0$，我们取 $\vec{x}_1$ 的第 2 维为 1，那么第 1 维也为 1。即 $\vec{x}_1=\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$

$(A-\lambda_2 E)\vec{x}_2=0 \Rightarrow \begin{pmatrix} 5 & 5 \\ 5 & 5 \end{pmatrix}\vec{x}_2=0 \Rightarrow \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}\vec{x}_2=0$，我们取 $\vec{x}_1$ 的第 2 维为 1，那么第 1 维也为 1。即 $\vec{x}_2=\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$

自此，我们就求出了 A 的两个特征向量。

那么矩阵 A 的特征矩阵为 $Q=[x_1,x_2]=\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

(3. 验证)

$$Q\Sigma Q^T=\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 8 & 0 \\ 0 & -2 \end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}=\begin{pmatrix} 4 & -1 \\ 4 & 1 \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}=\begin{pmatrix} 3 & 5 \\ 5 & 3 \end{pmatrix}=A$$

自此，我们已经为进行 SVD 分解做好了所有的数学准备，求解一个矩阵 A 的 SVD 分解的步骤如下：

1. 计算矩阵 AA^T 和 A^TA
2. 计算矩阵 AA^T 的特征值分解
3. 计算矩阵 A^TA 的特征值分解
4. 根据矩阵 AA^T 和 A^TA 的特征值分解结果得到矩阵 A 的奇异值（SVD）分解

例 4：

求矩阵 $A=\begin{pmatrix}1&0\\2&1\\0&1\end{pmatrix}$ 的奇异值分解。

解：

1. 计算矩阵 AA^T 和 A^TA

$$AA^T=\begin{pmatrix}1&0\\2&1\\0&1\end{pmatrix}\begin{pmatrix}1&2&0\\0&1&1\end{pmatrix}=\begin{pmatrix}1&2&0\\2&5&1\\0&1&1\end{pmatrix}$$

$$A^TA=\begin{pmatrix}1&2&0\\0&1&1\end{pmatrix}\begin{pmatrix}1&0\\2&1\\0&1\end{pmatrix}=\begin{pmatrix}5&2\\2&2\end{pmatrix}$$

2.计算矩阵 AA^T 的特征值分解

$$\left|AA^T-\lambda E\right|=0\Leftrightarrow\begin{vmatrix}1-\lambda&2&0\\2&5-\lambda&1\\0&1&1-\lambda\end{vmatrix}=0\Leftrightarrow\lambda(\lambda-1)(\lambda-6)=0\Leftrightarrow\begin{cases}\lambda_1=0\\\lambda_2=1\\\lambda_3=6\end{cases}$$

求解

$$\left(AA^T-\lambda_1E\right)\vec{u}_1=0\Leftrightarrow\begin{pmatrix}-5&2&0\\2&-1&1\\0&1&-5\end{pmatrix}\vec{u}_1=0$$

我们可以得到对应特征值 $\lambda_1=6$ 的特征向量 $\vec{u}_1=\frac{1}{\sqrt{30}}\begin{pmatrix}2\\5\\1\end{pmatrix}$，同理可以得到对应特征值 $\lambda_2=1$ 的特征向量 $\vec{u}_2=\frac{1}{\sqrt{5}}\begin{pmatrix}1\\0\\-2\end{pmatrix}$，对应特征值 $\lambda_3=0$ 的特征向量 $\vec{u}_3=\frac{1}{\sqrt{6}}\begin{pmatrix}2\\1\\-1\end{pmatrix}$。故我们得到了矩阵 AA^T 的特征矩阵

$$U=\begin{pmatrix}\vec{u}_1&\vec{u}_2&\vec{u}_3\end{pmatrix}=\begin{pmatrix}2/\sqrt{30}&1/\sqrt{5}&2/\sqrt{6}\\5/\sqrt{30}&0&1/\sqrt{6}\\5/\sqrt{30}&-2/\sqrt{5}&-1/\sqrt{6}\end{pmatrix}\begin{pmatrix}2/\sqrt{5}&1/\sqrt{5}\\-1/\sqrt{5}&2/\sqrt{5}\end{pmatrix}$$

3.计算矩阵 A^TA 的特征值分解

$$\left|A^TA-\lambda E\right|=0\Leftrightarrow\begin{vmatrix}5-\lambda & 2\\ 2 & 2-\lambda\end{vmatrix}=0\Leftrightarrow(\lambda-1)(\lambda-6)=0\Leftrightarrow\begin{cases}\lambda_1=6\\ \lambda_2=1\end{cases}$$

求解

$$(A^TA-\lambda_1E)\vec{v}_1=0\Leftrightarrow\begin{pmatrix}-1 & 2\\ 2 & -4\end{pmatrix}\vec{v}_1=0$$

得到对应特征值 $\lambda_1=6$ $\lambda_1=6$ 的特征向量 $\vec{v}_1=\frac{1}{\sqrt{5}}\begin{pmatrix}2\\1\end{pmatrix}$，同理可以得到对应特征值 $\lambda_2=1$ 的特征向量 $\vec{v}_2=\frac{1}{\sqrt{5}}\begin{pmatrix}-1\\2\end{pmatrix}$。故我们得到了矩阵 A^TA 的特征矩阵

$$V=\begin{pmatrix}\vec{v}_1 & \vec{v}_2\end{pmatrix}=\begin{pmatrix}2/\sqrt{5} & 1/\sqrt{5}\\ -1/\sqrt{5} & 2/\sqrt{5}\end{pmatrix}$$

5. 根据矩阵 AA^T 和 A^TA 的特征值分解结果得到矩阵 A 的奇异值（SVD）分解

$$A=U\Sigma V^T=U\begin{pmatrix}\sqrt{\lambda_1} & 0 & 0\\ 0 & \sqrt{\lambda_2} & 0\\ 0 & 0 & 0\end{pmatrix}V^T$$

$$=\begin{pmatrix}2/\sqrt{30} & 1/\sqrt{5} & 2/\sqrt{6}\\ 5/\sqrt{30} & 0 & 1/\sqrt{6}\\ 5/\sqrt{30} & -2/\sqrt{5} & -1/\sqrt{6}\end{pmatrix}\begin{pmatrix}\sqrt{6} & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix}2/\sqrt{5} & 1/\sqrt{5}\\ -1/\sqrt{5} & 2/\sqrt{5}\end{pmatrix}$$

第八章

不动点迭代法

二分法保证了对于任意一个连续函数 f 都可以收敛到一个零点上，但是如果我们知道更多 f 的性质，我们可以得到收敛速度更快的一些算法。

比如说，我们希望找到满足 $g(x^*)=x^*$ 的一个根，作为一个额外信息，我们知道 g 是满足利普希茨连续条件 $0\le c<1$ 的，即满足对于任意 x,y 满足 $|g(x)-g(y)|\lhd|x-y|$。

$g(x)=x$ 这个系统提供了一个求解的方法：

1. 取 x_0 作为 x*的猜测值。

2. 迭代计算 $x_k = g(x_{k-1})$

如果迭代收敛了，说明这个结果是一个满足 g 的一个不动点，满足上述条件。下图左图展示了一个收敛的情况，右图展示了一个发散的情况。

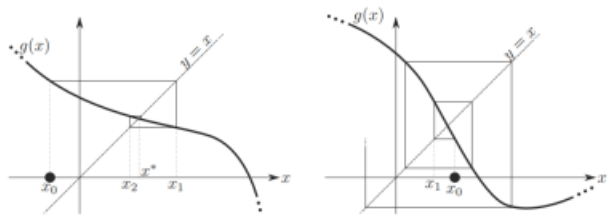


Figure 8.2 Convergence of fixed point iteration. Fixed point iteration searches for the intersection of $g(x)$ with the line $y = x$ by iterating $x_k = g(x_{k-1})$. One way to visualize this method on the graph of $g(x)$ visualized above is that it alternates between moving horizontally to the line $y = x$ and vertically to the position $g(x)$. Fixed point iteration (a) converges when the slope of $g(x)$ is small and (b) diverges otherwise.

当 c<1 时，利普希茨连续条件保证了如果有根，那么一定会收敛。证明如下：

$$\begin{aligned} E_k &= |x_k - x^*| \\ &= |g(x_{k-1}) - g(x^*)| \\ &\le c\,|x_{k-1} - x^*| \\ &= cE_{k-1} \end{aligned}$$

我们应用这个性质，那么就有当 $k \to \infty$ 时，有 $E_k \le c^k E_0 \to 0$。如果 g 是根的邻域中 $[x^*-\delta, x^*+\delta]$ 是利普希茨连续的且 c<1，那么只要我们把 x_0 选定在这个区间中，不动点迭代法会收敛。

牛顿法

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$$

弦截法(Secant Method)

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}.$$

第九章 无约束优化

第九章是一些机器学习的东西，具体就不展开写了。

从本质上去看，牛顿法是二阶收敛，梯度下降是一阶收敛，所以牛顿法就更快。如果更通俗地说的话，比如你想找一条最短的路径走到一个盆地的最底部，梯度下降法每次只从你当前所处位置选一个坡度最大的方向走一步，牛顿法在选择方向时，不仅会考虑坡度是否够大，还会考虑你走了一步之后，坡度是否会变得更大。所以，可以说牛顿法比梯度下降法看得更远一点，能更快地走到最底部。（牛顿法目光更加长远，所以少走弯路；相对而言，梯度下降法只考虑了局部的最优，没有全局思想。）

根据wiki上的解释，从几何上说，牛顿法就是用一个二次曲面去拟合你当前所处位置的局部曲面，而梯度下降法是用一个平面去拟合当前的局部曲面，通常情况下，二次曲面的拟合会比平面更好，所以牛顿法选择的下降路径会更符合真实的最优下降路径。

机器学习优化算法中梯度下降,牛顿法和拟牛顿法的优缺点详细介绍
http://www.elecfans.com/d/722244.html