

第二章作业

2.1 在什么条件下， x 有解且唯一？

在方程组的系数矩阵的秩与方程组增广矩阵的秩相等且均等于方程组中未知数个数的時候，方程组有唯一解。

2.2 高斯消元法的原理是什么？

通过用初等行变换将增广矩阵化为阶梯矩阵，再进一步化为行最简形式，然后通过回带求解线性方程组的解。

2.3 QR 分解的原理是什么？

利用 2 个正交矩阵的乘积一定是正交矩阵，正交矩阵的逆矩阵也是正交矩阵的特点，对正交矩阵进行多次分解，相乘。具体是将任意一个满秩矩阵可以分解成一个列向量正交矩阵 Q 与一个上三角矩阵 R 的积，即 $A = QR$ 。以施密特正交化方法为例，原理是利用正交投影的思想，将矩阵每个列作为一个基本单元，再将其转化为正交的基向量与在这个基向量上的投影长度的积。

2.4 Cholesky 分解的原理是什么？

将一个对称正定的矩阵表示成一个下三角矩阵 L 和其转置的乘积的分解。它要求矩阵的所有特征值必须大于零，故分解的下三角的对角元也是大于零的。如果矩阵 A 为 n 阶对称正定矩阵，则存在一个对角元素为正数的下三角实矩阵 L ，使得： $A = LL^T$ ，当限定 L 的对角元素为正时，这种分解是唯一的，称为 Cholesky 分解。

引申：

改进的 Cholesky 分解，大体思想不变，只是具体计算方法略微不同， $A = LD$ ，其中 L 为下三角矩阵， D 为对角线矩阵。

2.5 编程实现 A 为 100×100 随机矩阵时，用 QR 和 Cholesky 分解求 x 的程序。你可以参考本次课用到的 useEigen 例程。

思路：由于不可能任何矩阵都能化为正定矩阵，但任何实矩阵都可以分解为半正定矩阵和正交矩阵之积。所以本题应该使用 `ldlt`(半正定矩阵) 计算，但 Cholesky 分解需要正定矩阵，所以这里可能有一些出入。由于发现这个问题时间比较晚了，已经是周日晚上，所以仓促改了下，不知道对不对，待助教答疑时帮忙解答下。

```

luis@DESKTOP-78JRSUA: /mnt/d/ClionCppPrograma/homework/ch2/task2/build$ ./randomMatrix
QR result: 311.894 -262.248 34.83 -421.497 187.813 278.235 303.952 76.5881 64.9105 -360.269 -260.391 71.8179 64.0711 -121.382 -195.639 -84.35 -158.485 -29.532 127.649 214.244 -181.168 -89.5463 97.3587 -97.7664 162.279 130.757 -40.0229 110.906 -118.253 -95.6996 170.217 -100.967 137.335 -168.43 119.691 66.2849 -112.524 -107.59 -46.8291 65.7184 -5.11165 -17.6196 -46.6613 101.588 -114.76 -128.308 -336.293 41.7774 -195.239 19.7188 19.5892 -130.196 -202.256 53.083 -30.5273 -15.5751 224.245 -190.439 101.876 -377.009 -131.009 251.042 81.8799 -148.609 -39.8542 302.207 -223.552 87.4253 239.322 -92.2768 10.4051 55.0692 -66.0591 105.687 149.502 -36.2258 100.327 -185.897 -102.573 -220.762 111.167 -293.255 -306.691 -24.033 -27.1763 194.497 -163.067 27.6589 116.902 -38.9274 36.4682 409.048 39.1718 -155.387 46.1653 184.715 9.66797 -251.994 -72.5414 13.7519
Cholesky result: 311.894 -262.248 34.83 -421.497 187.813 278.235 303.952 76.5881 64.9105 -360.269 -260.391 71.8179 64.0711 -121.382 -195.639 -84.35 -158.485 -29.532 127.649 214.244 -181.168 -89.5463 97.3587 -97.7664 162.279 130.757 -40.0229 110.906 -118.253 -95.6996 170.217 -100.967 137.335 -168.43 119.691 66.2849 -112.524 -107.59 -46.8291 65.7184 -5.11165 -17.6196 -46.6613 101.588 -114.76 -128.308 -336.293 41.7774 -195.239 19.7188 19.5892 -130.196 -202.256 53.083 -30.5273 -15.5751 224.245 -190.439 101.876 -377.009 -131.009 251.042 81.8799 -148.609 -39.8542 302.207 -223.552 87.4253 239.322 -92.2768 10.4051 55.0692 -66.0591 105.687 149.502 -36.2258 100.327 -185.897 -102.573 -220.762 111.167 -293.255 -306.691 -24.033 -27.1763 194.497 -163.067 27.6589 116.902 -38.9274 36.4682 409.048 39.1718 -155.387 46.1653 184.715 9.66797 -251.994 -72.5414 13.7519

```

3.1 什么是正定矩阵和半正定矩阵？

正定矩阵：给定一个 n 乘 n 的实对称矩阵 A ，若对于任意长度为 n 的非零向量 x ，有 x 的转置乘以 A 再乘以 x 恒大于0，则矩阵 A 是一个正定矩阵。

半正定矩阵：给定一个 n 乘 n 的实对称矩阵 A ，若对于任意长度为 n 的非零向量 x ，有 x 的转置乘以 A 再乘以 x 恒大于等于0，则矩阵 A 是一个半正定矩阵。

3.2 对于方阵 A ，它的特征值是什么？特征向量是什么？特征值一定是实数吗？如何计算一个矩阵的特征值？

定义：给定一个 n 阶的矩阵 A ，若存在实数 λ 及 n 维非零向量 x ，使得 $Ax = \lambda x$ ，则称 λ 是矩阵 A 的特征值， x 是 A 属于特征值 λ 的特征向量。

为直观的理解这两个概念，下面将结合矩阵变换的几何意义来介绍这两个概念。对于一个 n 维的向量 x 来说，如果我们给它乘上一个 n 阶的方阵 A ，得到 Ax 。从几何角度来说，是对向量 x 进行了一个线性变换。变换之后得到的向量 y （也就是 Ax ）和原向量 x 的方向和长度都发生了改变。

但对于一个特定的矩阵 A 来说，总存在一些特定方向的向量 x ，使得 Ax 和 x 的方向没有发生变化，只是长度发生了变化。我们令这个长度发生的变化当做是系数 λ ，那么对于这样的向量 x 就称为是矩阵 A 的特征向量， λ 就是这个特征向量对应的特殊值，也就是矩阵 A 的特征值。

特征值不一定是实数，也可能是虚数，只有实对称矩阵的特征值才保证是实数，复矩阵的特征值也可能有实数。

求 n 阶矩阵 A 的特征值的基本方法：

根据定义可改写为关系式 $(A - \lambda E)x = 0$ ， E 为单位矩阵。若要求 x 有非零解，即求齐次线性方程组 $(A - \lambda E)x = 0$ 有非零解的值 λ 。即要求行列式 $\det(A - \lambda E) = 0$ 。解此行列式获得的 λ 值即为矩阵 A 的特征值。将此值回代入原式求得相应的 x ，即为输入这个行列式的特征向量。

3.3 什么是矩阵的相似性？相似性反映了什么几何意义？

设 A, B 都是 n 阶矩阵, 若存在可逆矩阵 P , 使 $P^{-1}AP=B$, 则称 B 是 A 的相似矩阵, 并称矩阵 A 与 B 相似, 记为 $A \sim B$ 。对进行运算称为对进行相似变换, 称可逆矩阵为相似变换矩阵。

相似的矩阵是同一个线性变换在不同基坐标系下的的不同描述。

同一个特定运动, 但坐标系改变的话这个运动的描述函数就会不一样, 如果这个函数可用矩阵等价替代的话, 一个坐标系就对应着一个矩阵, 因此这些矩阵就不同, 但这些矩阵必有关系, 这个关系就是相似。

3.4 矩阵一定可以对角化吗? 什么样的矩阵能保证对角化? 不能对角化的矩阵能够形成什么样的形式 (Jordan 标准形)?

只有主对角线上含有非零元素的矩阵称为对角矩阵。矩阵不一定可以对角化。

如果一个矩阵与一个对角矩阵相似, 我们就称这个矩阵可经相似变换对角化, 简称可对角化; 即 n 阶方阵可对角化的充分必要条件是它有 n 个线性无关的特征向量。

不能对角化的矩阵一定具有多重特征值, 对于不能对角化的矩阵也希望找到某种标准形式, 使之尽量接近对角化的形式。Jordan 型矩阵为上双对角矩阵, 其上对角线的元素为 1 或 0。这种标准型总能够实现, 但不一定数值稳定。该标准型是一组“几乎对角的”矩阵。

3.5 奇异值分解 (SVD) 是什么意思?

答: 奇异值分解 (SVD) 是一种用于将矩阵归约成其组成部分的矩阵分解方法, 以使后面的某些矩阵计算更简单。 $X=U\Sigma V^*$, $X=U\Sigma V^*$, 其中 U 是 $m \times m$ 阶酉矩阵; Σ 是 $m \times n$ 阶非负实数对角矩阵; 而 V^* , 即 V 的共轭转置, 是 $n \times n$ 阶酉矩阵。这样的分解就称作 X 的奇异值分解。

3.6 矩阵的伪逆是什么意思 (Pseudo inverse)? 莫尔——彭多斯逆是如何定义的? 怎么计算一个矩阵的伪逆?

伪逆矩阵是逆矩阵的广义形式。由于奇异矩阵或非方阵的矩阵不存在逆矩阵, 但可以用函数 `pinv(A)` 求其伪逆矩阵。基本语法为 `X=pinv(A)`, `X=pinv(A,tol)`, 其中 `tol` 为误差: `max(size(A))*norm(A)*eps`。函数返回一个与 A 的转置矩阵 A' 同型的矩阵 X , 并且满足: $AXA=A$, $XAX=X$ 。此时, 称矩阵 X 为矩阵 A 的伪逆, 也称为广义逆矩阵。`pinv(A)` 具有 `inv(A)` 的部分特性, 但不与 `inv(A)` 完全等同。

如果矩阵 B 满足以下四个条件, 那么我们可以称 B 为 A 的莫尔——彭多斯逆

- $BAB = B$
- $ABA = A$
- $(AB)^H = AB$
- $(BA)^H = BA$

计算矩阵的伪逆：

1. 计算矩阵 $A_{m \times n}$ 的满秩分解 $A = FG$ 。
2. 求广义逆矩阵，也就是矩阵的伪逆 $A^- = G^T(F^T A G^T)^{-1} F^T$ 。

3.7 对于超定方程： $Ax = b$ 且 A 不可逆时，我们通常计算最小二乘解： $x = \arg \min \|Ax - b\|$ 。线性方程的最小二乘解在代数意义上是可以解析地写出来的。请回答以下小问题：

(a) 在 b 非零时， x 的解是什么形式？事实上，我们可以对 A 求奇异值或对于 ATA 求特征值。请阐述两者之间的关系。

当 b 非零时，原方程可变换成求解非齐次线性方程组，则会出现以下几种可能性：

如果增广矩阵的阶梯型出现 $0 = \text{常数}$ 的不成立等式，则方程组无解；

如果他的秩小于方程组未知数的个数，则方程组有无穷多个解

如果他的秩等于方程组未知量的个数，则方程组有唯一解，解就是化简的增广矩阵最后一列。

特征值和奇异值的关系：对于非奇异矩阵进行奇异值分解（SVD），得到的奇异值其实就是特征值。对于奇异矩阵，就需要进行奇异值分解，对应奇异值。对于奇异矩阵，从 $A^T A$ 得到一个方阵，再求特征值。

(b) 当 $b = 0$ 时，我们希望求 x 的非零解。请说明如何求解 x 。

当秩小于未知数个数时，原方程有非零解，然后将系数矩阵 A 化为行最简形矩阵，并写出通解方程组；选取合适的未知自由量，并取出相应的基本向量组，代入通解方程组，得到原方程组的基础解系，进而写出通解。

(c) 请谈谈你对上述解法在几何意义上的理解。该问题为开放问题。

最小二乘解感觉是在方程无解的时候找到一个最近似的解，也即当向量 A 、 B 无法表示出向量 C 时，需要用 A 、 B 表示出一个与 C 最近似的向量，则可以用向量 C 向

A、B 所能表示出的区域做投影 D，则可得一个新的向量 $W = C - D$ ，那么想要找到最近似向量 C 的向量 D，只需要使向量 W 的模最小即可

4.1 说明一个激光传感器下的看到的点应该如何计算它的世界坐标。

设立坐标的转换关系公式并求出即可

p_L 表示该点在激光传感器坐标系下的坐标；

$T_{B,L}$ 表示激光传感器坐标系到Body系的转换关系；

$T_{R,B}$ 表示Body坐标系到定义在机器人脚部的R系的转换关系；

$T_{W,R}$ 表示机器人脚部的R系到世界坐标系的转换关系；

$$p_W = T_{W,R} T_{R,B} T_{B,L} p_L$$

4.2 取： $q_{WR} = [0.55, 0.3, 0.2, 0.2]$, $t_{WR} = [0.1, 0.2, 0.3]T$, $q_{RB} = [0.99, 0, 0, 0.01]$, $t_{RB} = [0.05, 0, 0.5]T$, $q_{BL} = [0.3, 0.5, 0, 20.1]$, $t_{BL} = [0.4, 0, 0.5]T$, $q_{BC} = [0.8, 0.2, 0.1, 0.1]$, $t_{BC} = [0.5, 0.1, 0.5]T$ 。现在假设相机传感器观察到自身坐标系下的点 $[0.3, 0.2, 1.2]$ ，请计算：(a)这个点在激光系下的坐标；(b)这个点在世界系下的坐标。

```
luis@DESKTOP-78JR5UA:/mnt/d/ClionCppPrograma/homework/ch2/task4/build$ ./coordinateTransformation
激光坐标系下该点位置:-0.641551 0.137248 1.10744
世界坐标系下该点位置 2.37806 -0.134779 0.873485
```

5.1 设有旋转矩阵 R ，证明 $RT R = I$ 且 $\det R = +1$ 。

思路：向量点积满足交换律

坐标系 $[e_1, e_2, e_3]$ 发生了旋转, 变成 $[e'_1, e'_2, e'_3]$ 对应坐标变换如下:

$$[e_1, e_2, e_3] \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = [e'_1, e'_2, e'_3] \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix}$$

上式同时左乘 $\begin{bmatrix} e_1^T \\ e_2^T \\ e_3^T \end{bmatrix}$ 得:

$$\boldsymbol{a} \triangleq \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} e_1^T e'_1 & e_1^T e'_2 & e_1^T e'_3 \\ e_2^T e'_1 & e_2^T e'_2 & e_2^T e'_3 \\ e_3^T e'_1 & e_3^T e'_2 & e_3^T e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \triangleq \boldsymbol{R} \boldsymbol{a}'$$

同理若上式同时左乘 $\begin{bmatrix} e'^T_1 \\ e'^T_2 \\ e'^T_3 \end{bmatrix}$ 得:

$$\boldsymbol{R}^{-1} \boldsymbol{a} \triangleq \begin{bmatrix} e'^T_1 e_1 & e'^T_1 e_2 & e'^T_1 e_3 \\ e'^T_2 e_1 & e'^T_2 e_2 & e'^T_2 e_3 \\ e'^T_3 e_1 & e'^T_3 e_2 & e'^T_3 e_3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \triangleq \boldsymbol{a}'$$

即

$$\boldsymbol{R}^{-1} = \begin{bmatrix} e'^T_1 e_1 & e'^T_1 e_2 & e'^T_1 e_3 \\ e'^T_2 e_1 & e'^T_2 e_2 & e'^T_2 e_3 \\ e'^T_3 e_1 & e'^T_3 e_2 & e'^T_3 e_3 \end{bmatrix}$$

而

$$\mathbf{R}^T = \begin{bmatrix} \mathbf{e}_1^T \mathbf{e}'_1 & \mathbf{e}_2^T \mathbf{e}'_1 & \mathbf{e}_3^T \mathbf{e}'_1 \\ \mathbf{e}_1^T \mathbf{e}'_2 & \mathbf{e}_2^T \mathbf{e}'_2 & \mathbf{e}_3^T \mathbf{e}'_2 \\ \mathbf{e}_1^T \mathbf{e}'_3 & \mathbf{e}_2^T \mathbf{e}'_3 & \mathbf{e}_3^T \mathbf{e}'_3 \end{bmatrix}$$

同时

$$[\mathbf{e}_{i1}, \mathbf{e}_{i2}, \mathbf{e}_{i3}] \begin{bmatrix} \mathbf{e}'_{j1} \\ \mathbf{e}'_{j2} \\ \mathbf{e}'_{j3} \end{bmatrix} = \mathbf{e}_i^T \mathbf{e}'_j = \mathbf{e}'_j{}^T \mathbf{e}_i = [\mathbf{e}'_{j1}, \mathbf{e}'_{j2}, \mathbf{e}'_{j3}] \begin{bmatrix} \mathbf{e}_{i1} \\ \mathbf{e}_{i2} \\ \mathbf{e}_{i3} \end{bmatrix}$$

故

$$\mathbf{R}^{-1} = \mathbf{R}^T$$

$$|\mathbf{R}| = \begin{vmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ \mathbf{e}_3^T \end{vmatrix} |\mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3| = 1$$

5.2 设有四元数 q ，我们把虚部记为 ε ，实部记为 η ，那么 $q = (\varepsilon, \eta)$ 。请说明 ε 和 η 的维度。

四元数的实部 η 是一维，虚部 ε 是三维

5.3 请证明对任意单位四元数 q_1, q_2 ，四元数乘法可写成矩阵乘法： $q_1 q_2 = q_1 + q_2$ 或者 $q_1 q_2 = q_2 \oplus q_1$ 。

思路：根据四元数运算法则计算即可证明

记

$$\mathbf{q}_1 = [\boldsymbol{\varepsilon}_1, \eta_1]^T, \quad \mathbf{q}_2 = [\boldsymbol{\varepsilon}_2, \eta_2]^T$$

或原始四元数表示法:

$$\mathbf{q}_1 = \eta_1 + x_1 \mathbf{i} + y_1 \mathbf{j} + z_1 \mathbf{k}, \quad \mathbf{q}_2 = \eta_2 + x_2 \mathbf{i} + y_2 \mathbf{j} + z_2 \mathbf{k}$$

有

$$\begin{aligned} \mathbf{q}_1 \mathbf{q}_2 = & \eta_1 \eta_2 - x_1 x_2 - y_1 y_2 - z_1 z_2 \\ & + (\eta_1 x_2 + x_1 \eta_2 + y_1 z_2 - z_1 y_2) \mathbf{i} \\ & + (\eta_1 y_2 - x_1 z_2 + y_1 \eta_2 + z_1 x_2) \mathbf{j} \\ & + (\eta_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 \eta_2) \mathbf{k} \end{aligned}$$

$$\begin{aligned} \mathbf{q}_1 \mathbf{q}_2 = & \eta_1 \eta_2 - x_1 x_2 - y_1 y_2 - z_1 z_2 \\ & + (x_1 \eta_2 + \eta_1 x_2 - z_1 y_2 + y_1 z_2) \mathbf{i} \\ & + (y_1 \eta_2 + z_1 x_2 + \eta_1 y_2 - x_1 z_2) \mathbf{j} \\ & + (z_1 \eta_2 - y_1 x_2 + x_1 y_2 + \eta_1 z_2) \mathbf{k} \end{aligned}$$

写成向量形式并利用内外积运算表示:

$$\mathbf{q}_1 \mathbf{q}_2 = [\eta_1 \boldsymbol{\varepsilon}_2 + \eta_2 \boldsymbol{\varepsilon}_1 + \boldsymbol{\varepsilon}_1 \times \boldsymbol{\varepsilon}_2, \eta_1 \eta_2 - \boldsymbol{\varepsilon}_1^T \boldsymbol{\varepsilon}_2]^T$$

记

$$\mathbf{q}_1^+ = \begin{bmatrix} \eta_1 \mathbf{1} + \boldsymbol{\varepsilon}_1^\times & \boldsymbol{\varepsilon}_1 \\ -\boldsymbol{\varepsilon}_1^T & \eta_1 \end{bmatrix}, \quad \mathbf{q}_2^\oplus = \begin{bmatrix} \eta_2 \mathbf{1} - \boldsymbol{\varepsilon}_2^\times & \boldsymbol{\varepsilon}_2 \\ -\boldsymbol{\varepsilon}_2^T & \eta_2 \end{bmatrix}$$

则

$$\mathbf{q}_1^+ \mathbf{q}_2 = \begin{bmatrix} \eta_1 \mathbf{1} + \boldsymbol{\varepsilon}_1^\times & \boldsymbol{\varepsilon}_1 \\ -\boldsymbol{\varepsilon}_1^T & \eta_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\varepsilon}_2 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} \eta_1 \boldsymbol{\varepsilon}_2 + \boldsymbol{\varepsilon}_1 \times \boldsymbol{\varepsilon}_2 + \eta_2 \boldsymbol{\varepsilon}_1 \\ -\boldsymbol{\varepsilon}_1^T \boldsymbol{\varepsilon}_2 + \eta_1 \eta_2 \end{bmatrix} = \mathbf{q}_1 \mathbf{q}_2$$

$$\mathbf{q}_2^\oplus \mathbf{q}_1 = \begin{bmatrix} \eta_2 \mathbf{1} - \boldsymbol{\varepsilon}_2^\times & \boldsymbol{\varepsilon}_2 \\ -\boldsymbol{\varepsilon}_2^T & \eta_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{\varepsilon}_1 \\ \eta_1 \end{bmatrix} = \begin{bmatrix} \eta_2 \boldsymbol{\varepsilon}_1 - \boldsymbol{\varepsilon}_2 \times \boldsymbol{\varepsilon}_1 + \eta_1 \boldsymbol{\varepsilon}_2 \\ -\boldsymbol{\varepsilon}_2^T \boldsymbol{\varepsilon}_1 + \eta_1 \eta_2 \end{bmatrix} = \mathbf{q}_1 \mathbf{q}_2$$

6.1 我们在课程中仅指出了罗德里格斯公式成立，但没有给出证明。请你证明此式。提示：参考

https://en.wikipedia.org/wiki/Rodrigues%27_rotation_formula。

思路：先将其退化为二维空间的向量分解，再通过旋转这个二维平面获得其在三维空间的向量分解，最后通过向量的加、减、点积、叉积等运算法则并辅以三角函数，求出所有分向量并汇总即可得证

令 \mathbf{k} 为定义旋转轴的单位向量， θ 为围绕 \mathbf{k} 旋转角度 θ 的任意向量（右手定则，图中逆时针方向）。

使用点积和叉积，向量 \mathbf{v} 可以分解为平行和垂直于轴 \mathbf{k} 的分量。

$$\mathbf{v} = \mathbf{v}_{\parallel} + \mathbf{v}_{\perp},$$

其中与 \mathbf{k} 平行的分量是

$$\mathbf{v}_{\parallel} = (\mathbf{v} \cdot \mathbf{k}) \mathbf{k}$$

称为 \mathbf{v} 在 \mathbf{k} 上的向量投影，垂直于 \mathbf{k} 的分量为

$$\mathbf{v}_{\perp} = \mathbf{v} - \mathbf{v}_{\parallel} = \mathbf{v} - (\mathbf{k} \cdot \mathbf{v}) \mathbf{k} = -\mathbf{k} \times (\mathbf{k} \times \mathbf{v})$$

称为 \mathbf{v} 从 \mathbf{k} 的向量拒绝。

向量 $\mathbf{k} \times \mathbf{v}$ 可以看作是 \mathbf{v}_{\perp} 关于 \mathbf{k} 逆时针旋转 90° 的副本，因此它们的大小相等但方向垂直。同样，向量 $\mathbf{k} \times (\mathbf{k} \times \mathbf{v})$ 的副本关于 \mathbf{k} 逆时针旋转了 180° ，因此 $\mathbf{k} \times (\mathbf{k} \times \mathbf{v})$ 和 \mathbf{v}_{\perp} 大小相等但方向相反（即它们是彼此的负数其他，因此是减号）。扩大向量 $\mathbf{k} \times \mathbf{v}$ 建立了平行分量和垂直分量之间的连接。供参考，公式为 $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) = (\mathbf{a} \cdot \mathbf{c}) \mathbf{b} - (\mathbf{a} \cdot \mathbf{b}) \mathbf{c}$ 给定任意三个向量 $\mathbf{a}, \mathbf{b}, \mathbf{c}$ 。

平行于轴的分量在旋转下不会改变大小和方向。

$$\mathbf{v}_{\parallel \text{rot}} = \mathbf{v}_{\parallel},$$

只有垂直分量会改变方向但保持其大小，根据

$$|\mathbf{v}_{\perp \text{rot}}| = |\mathbf{v}_{\perp}|, \\ \mathbf{v}_{\perp \text{rot}} = \cos(\theta) \mathbf{v}_{\perp} + \sin(\theta) \mathbf{k} \times \mathbf{v}_{\perp},$$

由于 \mathbf{k} 和 \mathbf{v}_{\perp} 是平行的，它们的叉积为零 $\mathbf{k} \times \mathbf{v}_{\parallel} = 0$ ，所以

$$\mathbf{k} \times \mathbf{v}_{\perp} = \mathbf{k} \times (\mathbf{v} - \mathbf{v}_{\parallel}) = \mathbf{k} \times \mathbf{v} - \mathbf{k} \times \mathbf{v}_{\parallel} = \mathbf{k} \times \mathbf{v}$$

它紧随其后

$$\mathbf{v}_{\perp \text{rot}} = \cos(\theta) \mathbf{v}_{\perp} + \sin(\theta) \mathbf{k} \times \mathbf{v}.$$

这种旋转是正确的，因为向量 \mathbf{v}_{\perp} 和 $\mathbf{k} \times \mathbf{v}$ 具有相同的长度，并且 $\mathbf{k} \times \mathbf{v}$ 是 \mathbf{v}_{\perp} 围绕 \mathbf{k} 逆时针旋转了 90° 。使用三角函数 \sin 和 \cos 对 \mathbf{v}_{\perp} 和 $\mathbf{k} \times \mathbf{v}$ 进行适当的缩放，得到旋转的垂直分量。旋转分量的形式类似于二维平面极坐标中的径向向量 (r, θ) 在笛卡尔基上

$$\mathbf{r} = r \cos(\theta) \mathbf{e}_x + r \sin(\theta) \mathbf{e}_y,$$

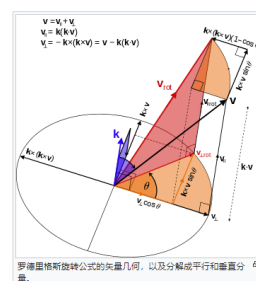
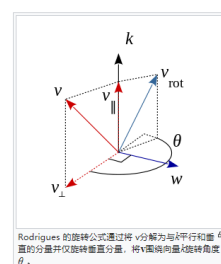
其中 $\mathbf{e}_x, \mathbf{e}_y$ 是其指示方向上的单位向量。

现在完整的旋转向量是

$$\mathbf{v}_{\text{rot}} = \mathbf{v}_{\parallel \text{rot}} + \mathbf{v}_{\perp \text{rot}},$$

通过代入方程 $\mathbf{v}_{\parallel \text{rot}}$ 和 $\mathbf{v}_{\perp \text{rot}}$ 的定义，得到

$$\begin{aligned} \mathbf{v}_{\text{rot}} &= \mathbf{v}_{\parallel} + \cos(\theta) \mathbf{v}_{\perp} + \sin(\theta) \mathbf{k} \times \mathbf{v} \\ &= \mathbf{v}_{\parallel} + \cos(\theta) (\mathbf{v} - \mathbf{v}_{\parallel}) + \sin(\theta) \mathbf{k} \times \mathbf{v} \\ &= \cos(\theta) \mathbf{v} + (1 - \cos(\theta)) \mathbf{v}_{\parallel} + \sin(\theta) \mathbf{k} \times \mathbf{v} \\ &= \cos(\theta) \mathbf{v} + (1 - \cos(\theta)) (\mathbf{k} \cdot \mathbf{v}) \mathbf{k} + \sin(\theta) \mathbf{k} \times \mathbf{v} \end{aligned}$$



以上摘抄自维基百科

6.2 请使用此式请明 $R^{-1} = R^T$ 。

思路：由罗德里格斯公式可知，其转置后只会使 $\sin(\theta) \mathbf{k}^\wedge$ 这一项变为相反数，由此得出转置后的结果，又通过罗德里格斯公式旋转思想可知，他的逆就是其向反方向旋转相同角度，那么也能得出只会使 $\sin(\theta) \mathbf{k}^\wedge$ 这一项变为相反数的结果，即得证

$$R^T = -\sin(\theta) \mathbf{k}^\wedge + \cos(\theta) \mathbf{I} + (1 - \cos(\theta)) \mathbf{K} \mathbf{K}^T = \sin(-\theta) \mathbf{k}^\wedge + \cos(\theta) \mathbf{I} + (1 - \cos(\theta)) \mathbf{K} \mathbf{K}^T$$

7 课程中介绍了单位四元数可以表达旋转。其中，在谈论用四元数 q 旋转点 p 时，结果为： $p' = qpq^{-1}$ 。我们说，此时 p' 必定为虚四元数（实部为零）。请你验证上述说法。此外，上式亦可写成矩阵运算： $p' = Qp$ 。请根据你的推导，给出矩阵 Q 。注意此时 p 和 p' 都是四元数形式的变量，所以 Q 为 4×4 的矩阵。

思路：四元数运算法则结合矩阵基础运算即可得证

记

$$\mathbf{q} = [s, \mathbf{v}]^T, \mathbf{p} = [0, \mathbf{u}]^T$$

有

$$\mathbf{q}^+ (\mathbf{q}^{-1})^\oplus = \begin{bmatrix} s & -\mathbf{v}^T \\ \mathbf{v} & s\mathbf{I} + \mathbf{v}^\wedge \end{bmatrix} \begin{bmatrix} s & \mathbf{v}^T \\ -\mathbf{v} & s\mathbf{I} + \mathbf{v}^\wedge \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0}^T & \mathbf{v}\mathbf{v}^T + s^2\mathbf{I} + 2s\mathbf{v}^\wedge + (\mathbf{v}^\wedge)^2 \end{bmatrix}$$

$$\mathbf{p}' = \mathbf{q}\mathbf{p}\mathbf{q}^{-1} = \mathbf{q}^+ (\mathbf{q}^{-1})^\oplus \mathbf{p} = \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{v}\mathbf{v}^T + s^2\mathbf{I} + 2s\mathbf{v}^\wedge + (\mathbf{v}^\wedge)^2 \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} 0 \\ ((\mathbf{v}\mathbf{v}^T + s^2\mathbf{I} + 2s\mathbf{v}^\wedge + (\mathbf{v}^\wedge)^2) \mathbf{u}) \end{bmatrix}$$

由上式结果看 \mathbf{p}' 实部为0即 \mathbf{p}' 为虚四元数

若记 $\mathbf{q} = s + xi + yj + zk$ 则旋转矩阵 \mathbf{Q} 可表示为：

$$\mathbf{Q} = \mathbf{q}^+ (\mathbf{q}^{-1})^\oplus = \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0}^T & \mathbf{v}\mathbf{v}^T + s^2\mathbf{I} + 2s\mathbf{v}^\wedge + (\mathbf{v}^\wedge)^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 - 2y^2 - 2z^2 & 2xy - 2sz & 2sy + 2xz \\ 0 & 2xy + 2sz & 1 - 2x^2 - 2z^2 & 2yz - 2sx \\ 0 & 2xz - 2sy & 2sx + 2yz & 1 - 2x^2 - 2y^2 \end{bmatrix}$$

8 请说明该程序中哪些地方用到了 C++11 标准的内容。

line17：范围for语句，不用下标就能访问元素；c++11新标准引入的一种更简单的for语句，其语法形式是：

```
for (declaration: expression)
    statement
```

line17：auto 关键字实现自动类型推导，c++11新标准引入，用它就能让编译器替我们去分析表达式所属的类型。

line16：`[](const A&a1, const A&a2) {return a1.index<a2.index;}`：lambda表达式，一个lambda表达式表示一个可调用的代码单元。我们可以理解为一个未命名的内联函数。于任何函数类似，一个lambda具有一个返回类型、一个参数列表和一个函数体。但与函数不同，lambda可能定义在函数内部。一个lambda表达式具有如下形式：

```
[capture list](parameter list) -> return type {function body}
```

line15：`vector<A> avec {a1, a2, a3};` 使用了C++11新标准提出的列表初始化vector对象。用花括号括起来的0个或者多个初始元素值被赋给了avec 这个vector对象。