

第二节课习题

✓ 2. 熟悉Eigen矩阵运算

设线性方程 $Ax = b$ ，在 A 为方阵的前提下，请回答以下问题：

1. 在什么条件下， x 有解且唯一？

方阵 A 满秩，且系数矩阵 A 的秩和增广矩阵 \bar{A} 的秩相等，即 $r(A) = r(\bar{A}) = n$ 。

2. 高斯消元法的原理是什么？

通过初等行变换将增广矩阵转化为行阶梯型矩阵，最后再将得到的行阶梯型矩阵还原为线性方程组求出原方程组的解。从下边的例子可以看出方程组中每个方程含有的未知数的个数降到最低，并且最下面的方程含有的未知数的个数最少。

例：利用高斯消元法求解方程组
$$\begin{cases} x_1 + x_2 + x_3 = 3 \\ x_1 + 2x_2 + 4x_3 = 7 \\ x_1 + 3x_2 + 9x_3 = 13 \end{cases}$$

解：对增广矩阵作如下变换：

$$\begin{bmatrix} 1 & 1 & 1 & 3 \\ 1 & 2 & 4 & 7 \\ 1 & 3 & 9 & 13 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 4 \\ 0 & 2 & 8 & 10 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 4 \\ 0 & 1 & 4 & 5 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 4 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

进行回代操作：

$$\begin{cases} x_1 + x_2 + x_3 = 3 \\ x_2 + 3x_3 = 4 \\ x_3 = 1 \end{cases}$$

最后得到原线性方程组的解为
$$\begin{cases} x_1 = 1 \\ x_2 = 1 \\ x_3 = 1 \end{cases}$$

3. QR 分解的原理是什么？

参考资料：<https://blog.csdn.net/u010945683/article/details/45972819>

<https://zhuanlan.zhihu.com/p/84415000>

首先，QR分解的定义：

一个矩阵 $A \in \mathbb{R}^{m \times n}, m \geq n$ 可以被分解成 $A = QR$, 其中 :

- $Q \in \mathbb{R}^{m \times n}$ 是正交矩阵
- $R \equiv \begin{bmatrix} \hat{R} \\ 0 \end{bmatrix} \in \mathbb{R}^{m \times n}$
- $\hat{R} \in \mathbb{R}^{m \times n}$ 是上三角矩阵

计算QR分解一共有三种方法:

- Gram-Schmidt Orthogonalization (也就是常说的施密特正交化)
- Householder Triangularization
- Givens Rotations

QR分解原理:

QR矩阵分解, 就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, 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。这一过程的具体介绍可参考<https://blog.csdn.net/u010945683/article/details/45972819>, 讲的非常详细。

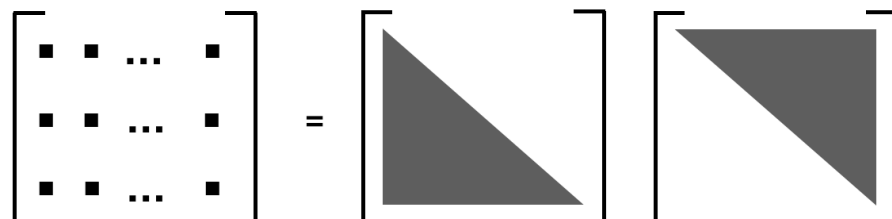
4. Cholesky 分解的原理是什么?

Cholesky分解定理:

若 $A \in \mathbb{R}^{n \times n}$ 是对称正定矩阵, 则存在一个对角元均为正数的下三角矩阵 $L \in \mathbb{R}^{n \times n}$, 使得

$$A = LL^T$$

Cholesky 分解是将一个实对称正定矩阵分解成一个下三角矩阵与其共轭转置的乘积。



A

L

L^T

$$A = A^T$$

$$x^T A x > 0 \quad (x \neq 0)$$

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。

代码说明:

若使用Cholesky分解, 矩阵需要满足为正定矩阵, 也就是

$$A = LL^T$$

所以先随机化生成一个矩阵Random_Matrix, 并根据此矩阵来获得矩阵 A 。

核心代码:

```
1 int main( int argc, char** argv )
2 {
3     //构造动态矩阵, 类型为double型
4     MatrixXd Random_Matrix = MatrixXd::Random( MATRIX_SIZE,
5     MATRIX_SIZE );
6     MatrixXd A = Random_Matrix * Random_Matrix.transpose();
7     //使得A为正定对称矩阵, 才能使得Cholesky分解成功
8     VectorXd B = VectorXd::Random( MATRIX_SIZE );
9     VectorXd x = A.colPivHouseholderQr().solve(B); //调用QR分
10    解求解
11    cout << "调用QR分解求得Ax=b方程的解为\n" << x << endl;
12
13    VectorXd y = A.llt().solve(B); //调用Cholesky分解求
14    解
15    cout << "调用cholesky分解求得Ax=b方程的解为\n" << y << endl;
16
17    return 0;
18 }
```

✓ 3.矩阵论基础

1. 什么是正定矩阵和半正定矩阵?

参考博客:<https://zhuanlan.zhihu.com/p/44860862>

在学习矩阵论或者线性代数等数学课程时, 我们都会学习到正定矩阵和半正定矩阵的知识。半正定矩阵的定义其实是包括正定矩阵的。在不考虑由复数构成的矩阵的话, 正定矩阵和半正定矩阵的定义是很简单的:

正定矩阵：给定一个大小为 $n \times n$ 的实对称矩阵 A ，若对于任意长度为 n 的非零向量 x ，有 $x^T A x > 0$ 恒成立，则矩阵 A 是一个正定矩阵。

半正定矩阵：给定一个大小为 $n \times n$ 的实对称矩阵 A ，若对于任意长度为 n 的向量 x ，有 $x^T A x \geq 0$ 恒成立，则矩阵 A 是一个正定矩阵。

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？

参考资料：

- <https://zhuanlan.zhihu.com/p/104980382>
- <https://zhuanlan.zhihu.com/p/95836870>

对于做算法的人来说，相比于使用code来计算矩阵的特征值和特征向量，深入理解这两个概念和几何意义更为重要。因为在机器学习领域经常会用到这两者进行降维处理。矩阵的特征值和特征向量的定义如下：

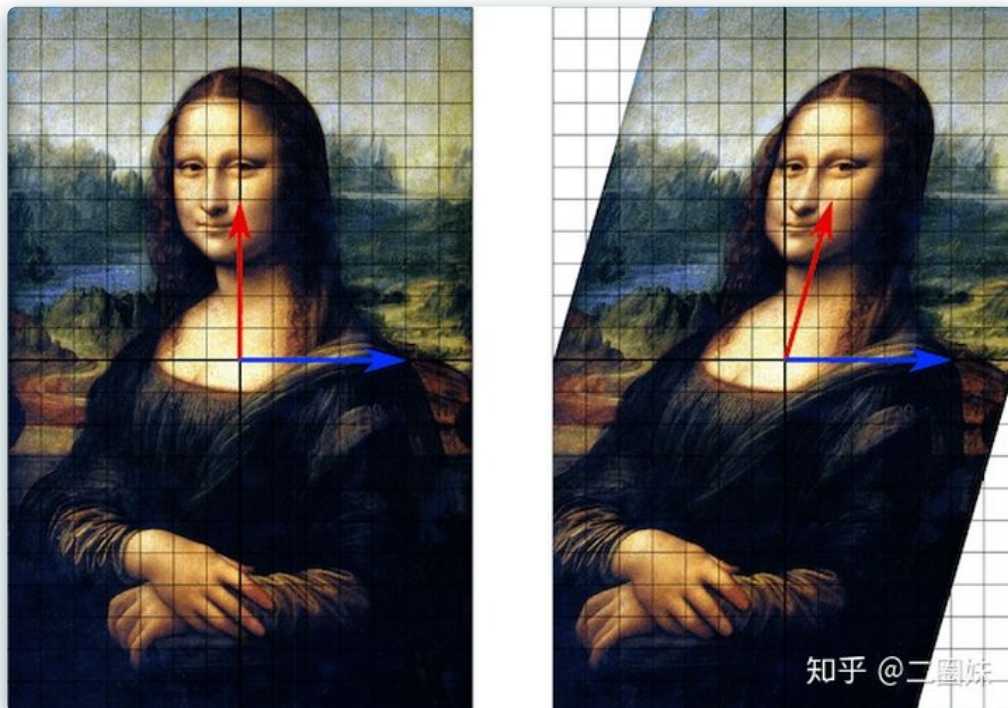
假设我们有一个 n 阶的矩阵 A 以及一个实数 λ ，使得我们可以找到一个非零向量 x ，满足：

$$Ax = \lambda x$$

如果能找到这样的值的话，我们就称 λ 是矩阵 A 的特征值，非零向量 x 是矩阵的特征向量。

几何意义：单从定义来看很难直观的理解这两个概念，下面我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**。



以上图中经过错切变换的蒙娜丽莎为例，可以将蒙娜丽莎原图简化成一个 2×2 的方阵，经过变换这幅图在水平方向上的相对位置是没有改变的（比如俩眼睛之间的距离没变化吧）， $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ 就是它的一个特征向量，对应的特征值 λ 为 1。

特征值一定是实数吗？

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

计算矩阵的特征值

由定义可做如下变形：

$$Ax = \lambda x \longrightarrow (x - \lambda I)x = 0$$

如果上述式子有非零解，那么 $A - \lambda I$ 是奇异的，也就是行列式为零。因此我们可以通过下式求出特征值：

$$\det(A - \lambda I) = 0$$

然后，针对每个特征值，再通过求解 $(A - \lambda I)x = 0$ 来找到特征向量。

3. 什么是矩阵的相似性？相似性反映了什么几何意义？

相似矩阵定义：

设 A, B 为 n 阶矩阵，如果有 n 阶可逆矩阵 P 存在，使得

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

则称矩阵 A 与 B 相似，记为 $A \sim B$ 。

这里推荐先看一下这篇解说就更容易理解下边这句话了：<https://zhuanlan.zhihu.com/p/151231495>

相似矩阵的本质就是，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。同一个运动过程（线性变换）在不同坐标系（不同基）中的表示矩阵（相似矩阵）虽然不一样，但实质上是指的同一个线性变换。

从机器人坐标变换的角度探究相似矩阵的几何意义：

- 坐标和变换的定义：

三维空间中某一点的位置在坐标系 $OX_0Y_0Z_0$ 和坐标系 $OX_1Y_1Z_1$ 中的坐标分别是 P^0 和 P^1 ，从坐标系 $OX_0Y_0Z_0$ 和坐标系 $OX_1Y_1Z_1$ 的旋转变换为 $R = R_{10}$

通过某一旋转变换，该点到达新的位置，点的坐标在坐标系 $OX_0Y_0Z_0$ 和坐标系 $OX_1Y_1Z_1$ 中分别为 p'^0 和 p'^1 。该旋转变换，在两个坐标系下分别为 A 和 B 。

- 坐标变换关系

$$\begin{aligned} P'^0 &= AP^0 \\ P'^1 &= BP^1 \\ P^0 &= R_{01}P^1 \\ P'^0 &= R_{01}P'^1 \end{aligned}$$

- 推导

$$\begin{aligned} P'^1 &= BP^1 \\ (R_{01})^{-1}P'^0 &= B(R_{01})^{-1}P^0 \\ (R_{01})^{-1}AP^0 &= B(R_{01})^{-1}P^0 \\ (R_{01})^{-1}A &= B(R_{01})^{-1} \\ (R_{01})^{-1}AR_{01} &= B \end{aligned}$$

- 结论

$$B = R^{-1}AR$$

其中 R 为从坐标系 $OX_0Y_0Z_0$ 到坐标系 $OX_1Y_1Z_1$ 的旋转变换。

这块在网上搜到了下边这篇文章，从定义到推导，简单明了的说明了在机器人坐标变换角度中矩阵相似的几何意义。<https://blog.csdn.net/zhpfeng10/article/details/108977526>

4. 矩阵一定能对角化吗？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)？

定义：对角矩阵是指只有主对角线上含有非零元素的矩阵，即，已知一个 $n \times n$ 矩阵 M ，如果对于 $i \neq j, M_{ij} = 0$ ，则该矩阵为对角矩阵。

如果存在一个矩阵 A ，使 $A^{-1}MA$ 的结果为对角矩阵，则称矩阵 A 将矩阵 M 对角化。

对于一个矩阵来说，不一定存在将其对角化的矩，但是任意一个 $n \times n$ 矩阵如果存在 n 个线性不相关的特征向量，则该矩阵可被对角化。具体原因在这篇文章中说的很明白：<https://www.zhihu.com/question/323578684/answer/753474442>

若实在不能对角化，可化成Jordan标准型

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。具体形式参考《矩阵分析与应用》张贤达这本书p237页。

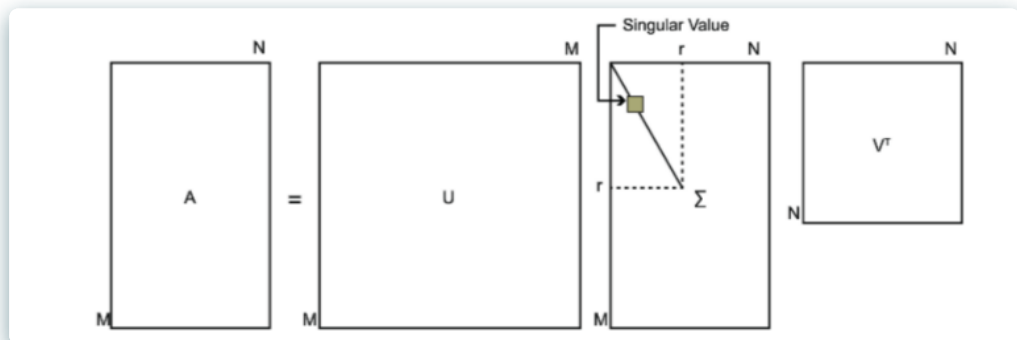
5. 奇异值分解 (SVD) 是什么意思？

定义：

SVD的基本公式：

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

其中， $A \in \mathbb{R}^{m \times n}$, $U \in \mathbb{R}^{m \times m}$, $\Sigma \in \mathbb{R}^{m \times n}$ 且除了主对角线上的元素以外全为0，主对角线上的每个元素都称为奇异值，且已按大小排好序， $V \in \mathbb{R}^{n \times n}$ 。其中， U 的列向量即是 AA^T 的特征向量，一般我们将 U 中的每个特征向量叫做 A 的左奇异向量； V 的列向量即是 $A^T A$ 的特征向量，一般我们将 V 中的每个特征向量叫做 A 的右奇异值。



上图可以很形象的看出上面SVD的定义。

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, 由于奇异矩阵或者非方阵的矩阵不存在逆矩阵, 但可以有伪逆矩阵 (或者说是广义逆矩阵)。

满足 $A^L A = E$, 但不满足 $AA^L = E$ 的矩阵 A^L 称为矩阵 A 的左逆矩阵。同理, 满足 $AA^R = E$, 但不满足 $A^R A = E$ 的矩阵 A^R 称为矩阵 A 的右逆矩阵。仅当 $m \geq n$ 时, 列满秩, 矩阵 $A_{m \times n}$ 有左逆矩阵, $A^L = (A^T A)^{-1} A^T$; 仅当 $m \leq n$ 时, 行满秩, 矩阵 $A_{m \times n}$ 有右逆矩阵, $A^R = A^T (A^T A)^{-1}$; 当 $m = n$, $A_{m \times n}$ 的秩为 $r \leq m = n$, 对 A 进行奇异值分解 $A = U \Sigma V^T$, A 的伪逆矩阵为 $A^+ = V \Sigma^+ U^T$

莫尔-彭多斯逆 (Moore-Penrose 广义逆矩阵)

设 $A \in C^{m \times n}$, 如果 $G \in C^{n \times m}$ 满足,

- (1). $AGA = A$,
- (2). $GAG = G$,
- (3). $(AG)^H = AG$,
- (4). $(GA)^H = GA$.

则 G 为 A 的 Moore - Penrose 广义逆矩阵。 (A^T 为 A 的转置共轭矩阵)

计算矩阵的伪逆

算法核心步骤主要有两步:

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 $A^- = G^T (F^T A G^T)^{-1} F^T$ 。

更具体计算递推公式在《矩阵分析与应用》张贤达这本书中p95页有详细的介绍。

对于 Moore-Penrose 逆矩阵的计算方法有两种, 这里就不详细说明了, 在《矩阵分析与应用》张贤达这本书中p106有完整介绍。

7. a) 当 $b \neq 0$ 时, 也即是求解非齐次线性方程组, 有下面三种情况:

- 如果增广矩阵的阶梯型出现 $0 = \text{常数}$, 则方程组无解;
- 如果非零行个数 $r < \text{方程组未知数的个数 } n$, 则方程组有无穷多个解
- 如果非零行个数 $r = \text{方程组未知量的个数 } n$, 则方程组有唯一解, 解就是化简的增广矩阵最后一列。

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

当 $(r < n)$, 原方程有非零解, 然后将系数矩阵 A 化为行最简形矩阵, 并写出同解方程组;

选取合适的未知自由量, 并取出相应的基本向量组, 代入同解方程组, 得到原方程组的基础解系, 进而写出通解。

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，将 $A^T A$ 得到一个方阵，再求特征值。

c)上述解法的几何意义

之前曾经了解过，齐次线性方程的非零解求解可以用来判断三点是否在同一直线上，或者三点是否均经过某一点的判断。脑子不好使，具体的几何意义我还能抽象的想象出来。

✓ 4. 几何运算练习

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。

计算如下公式可得它的世界坐标系，其中，

- 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；

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

2. 相机传感器观察到自身坐标系下的点 p_C [0.3, 0.2, 1.2]，请计算某一点的激光系下的坐标和世界系下的坐标；

a) 激光系下的坐标，根据下列公式求解：

$$\begin{aligned} p_B &= T_{B,C} p_C \\ p_B &= T_{B,L} p_L \\ p_L &= (T_{B,L})^{-1} T_{B,C} p_C \end{aligned}$$

坐标点为：[0.35, 1.2, -0.02]

b) 世界系下的坐标，根据下列公式求解：

$$p_W = T_{W,R} T_{R,B} T_{B,C} p_C$$

坐标点为：[2.38, -0.13, 0.87]

代码运行结果截图：

```
ximing@ubuntu:~/code/slam_code/ch2/task4/build$ make
Scanning dependencies of target useEigen
[ 50%] Building CXX object CMakeFiles/useEigen.dir/useEigen.cpp.o
[100%] Linking CXX executable useEigen
[100%] Built target useEigen
ximing@ubuntu:~/code/slam_code/ch2/task4/build$ ./useEigen
激光系下坐标: 0.354286 1.23714 -0.0285714
世界系下坐标: 2.37806 0.134779 0.873485
ximing@ubuntu:~/code/slam_code/ch2/task4/build$
```

具体代码可查看相关代码文件。

✓ 5. 旋转的表达

1. 设有旋转矩阵 R , 证明 $R^T R = I$ 且 $\det R = +1$ 。

证明: 根据《视觉SLAM十四讲》p44页公式

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

两边同时左乘 $\begin{bmatrix} e_1^T \\ e_2^T \\ e_3^T \end{bmatrix}$ 得:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = R a' = \begin{bmatrix} e_1^T e'_1 & e_1^T e'_2 & e_1^T e'_3 \\ e_2^T e'_1 & e_2^T e'_2 & e_2^T e'_3 \\ e_3^T e'_1 & e_3^T e'_2 & e_3^T e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \quad (1)$$

两边同时左乘 $\begin{bmatrix} (e'_1)^T \\ (e'_2)^T \\ (e'_3)^T \end{bmatrix}$ 得:

$$R^{-1} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = a' = \begin{bmatrix} e_1^T e'_1 & e_2^T e'_1 & e_3^T e'_1 \\ e_1^T e'_2 & e_2^T e'_2 & e_3^T e'_2 \\ e_1^T e'_3 & e_2^T e'_3 & e_3^T e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \quad (2)$$

由以上式1和式2可以看出 $R^{-1} = R^T$

所以 $R \cdot R^T = R \cdot R^{-1} = E$ 得证。

证明 $\det(R) = +1$

旋转矩阵 R 把 n 维空间的一组标准化正交基转化为另一组标准正交基，这两组标准正交基分别用矩阵 I , I' 来表示,

所以, $RI = I'$, 因为 $\det(I) = 1$, $\det(I') = 1$

所以, $\det(R) = 1$ 得证。

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。

一个四元数 q 拥有一个实部和三个虚部, 由三个虚部来表示三维的旋转。所以 ε 是三维的, 实部 η 是一维的。

3. 证明对任意单位四元数 q_1, q_2 , 四元数乘法可以写成矩阵乘法

$$q^+ = \begin{bmatrix} \eta I + \varepsilon^\times & \varepsilon \\ -\varepsilon^T & \eta \end{bmatrix}, q^\oplus = \begin{bmatrix} \eta I - \varepsilon^\times & \varepsilon \\ -\varepsilon^T & \eta \end{bmatrix},$$

两个四元数 q_1, q_2 , 它们的向量表示为 $[\varepsilon_1, \eta_1]^T, [\varepsilon_2, \eta_2]^T$, 其原始四元数表示为:

$$q_1 = x_1 i + y_1 j + z_1 k + \eta_1$$

$$q_2 = x_2 i + y_2 j + z_2 k + \eta_2$$

q_1 与 q_2 相乘, 整理可得,

$$\begin{aligned} q_1 q_2 = & \eta_1 \eta_2 - x_1 x_2 - y_1 y_2 - z_1 z_2 + (\eta_1 x_2 + x_1 \eta_2 + y_1 z_2 - z_1 y_2) i \\ & + (\eta_1 y_2 - x_1 z_2 + y_1 \eta_2 + z_1 x_2) j + (\eta_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 \eta_2) k \end{aligned}$$

将上式写成向量形式并利用内外积运算, 该表达式为:

$$q_1 q_2 = \begin{bmatrix} \eta_1 \varepsilon_2 + \varepsilon_1 \eta_2 + \varepsilon_1 \times \varepsilon_2 \\ \eta_1 \eta_2 - \varepsilon_1^T \varepsilon_2 \end{bmatrix}$$

四元数乘法写成矩阵形式:

$$q_1^+ q_2 = \begin{bmatrix} \eta_1 I + \varepsilon_1^\times & \varepsilon_1 \\ -\varepsilon_1^T & \eta_1 \end{bmatrix} \begin{bmatrix} \varepsilon_2 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} \eta_1 \varepsilon_2 + \varepsilon_1 \eta_2 + \varepsilon_1^\times \varepsilon_2 \\ \eta_1 \eta_2 - \varepsilon_1^T \varepsilon_2 \end{bmatrix} = q_1 q_2$$

同理可以证明:

$$q_1 q_2 = q_2^\oplus q_1$$

✓ 6. 罗德里格斯公式的证明

由于数学功底有限, 所以参考了很多材料, 最后经过整理, 自我消化, 然后按照我的理解在下边记录该公式的推导过程。

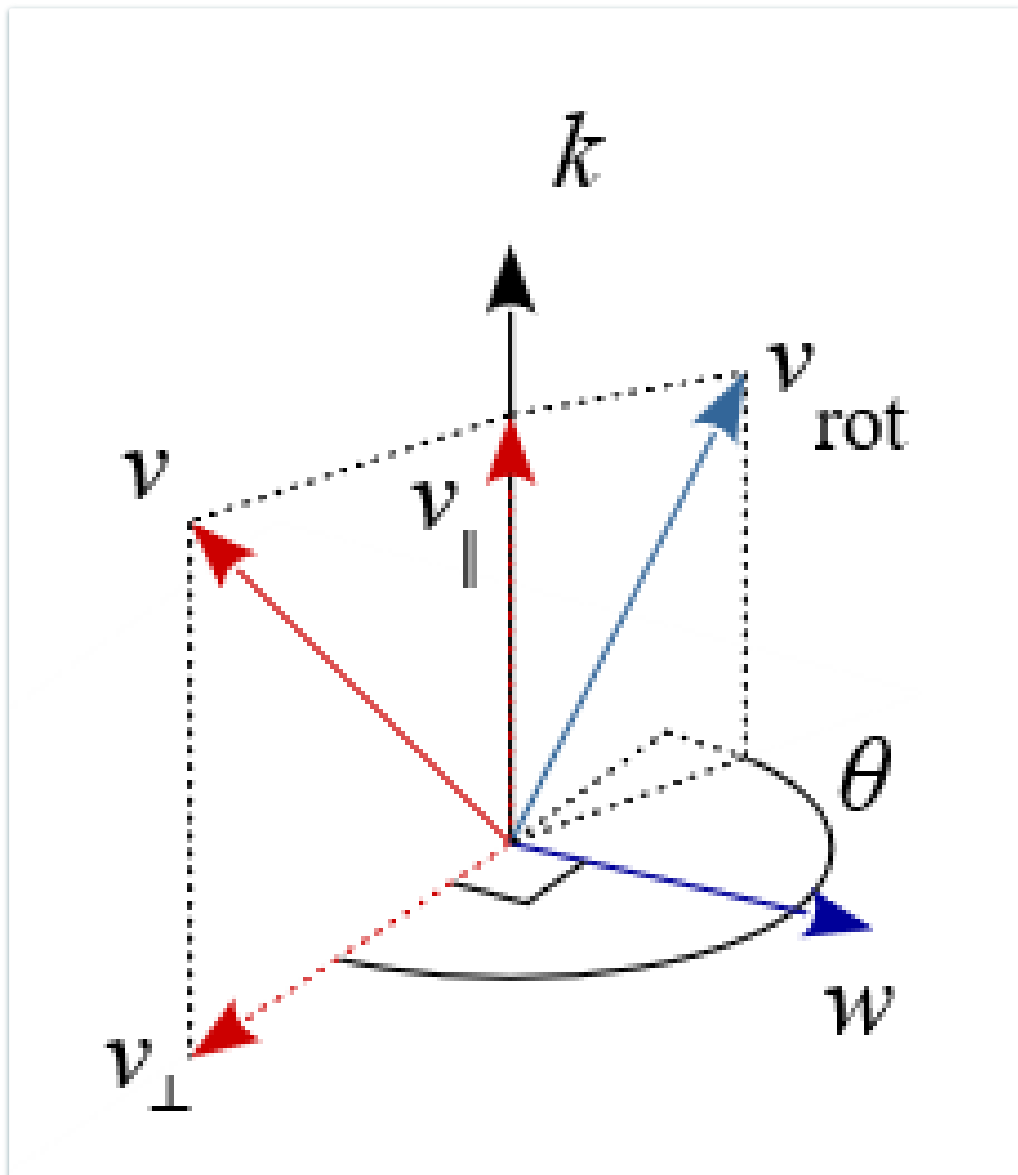
- <https://zhuanlan.zhihu.com/p/79061355>
- <https://blog.csdn.net/SKANK911/article/details/90056646>
- https://en.wikipedia.org/wiki/Rodrigues%27_rotation_formula

罗德里格斯公式描述了从旋转向量到旋转矩阵的转换关系。设旋转向量长度为 θ ，方向为 n ，那么旋转矩阵 R 为：

$$R = \cos \theta I + (1 - \cos \theta)nn^T + \sin \theta n^\wedge.$$

1. 罗德里格斯公式证明

下图来自维基百科中罗德里格斯公式词条。如图所示，假设存在一个单位向量 k ，某个向量 v 以 k 为旋转轴进行旋转，角度为 θ ，旋转后的向量为 v_{rot} 。



其中向量 v 可以分解成垂直于 k 轴和平行于 k 轴的两个向量：

$$v = v_{\perp} + v_{\parallel}$$

由点乘的投影几何意义可得：

$$v_{\parallel} = (v \cdot k)k$$

由向量减法可得：

$$v_{\perp} = v - v_{\parallel}$$

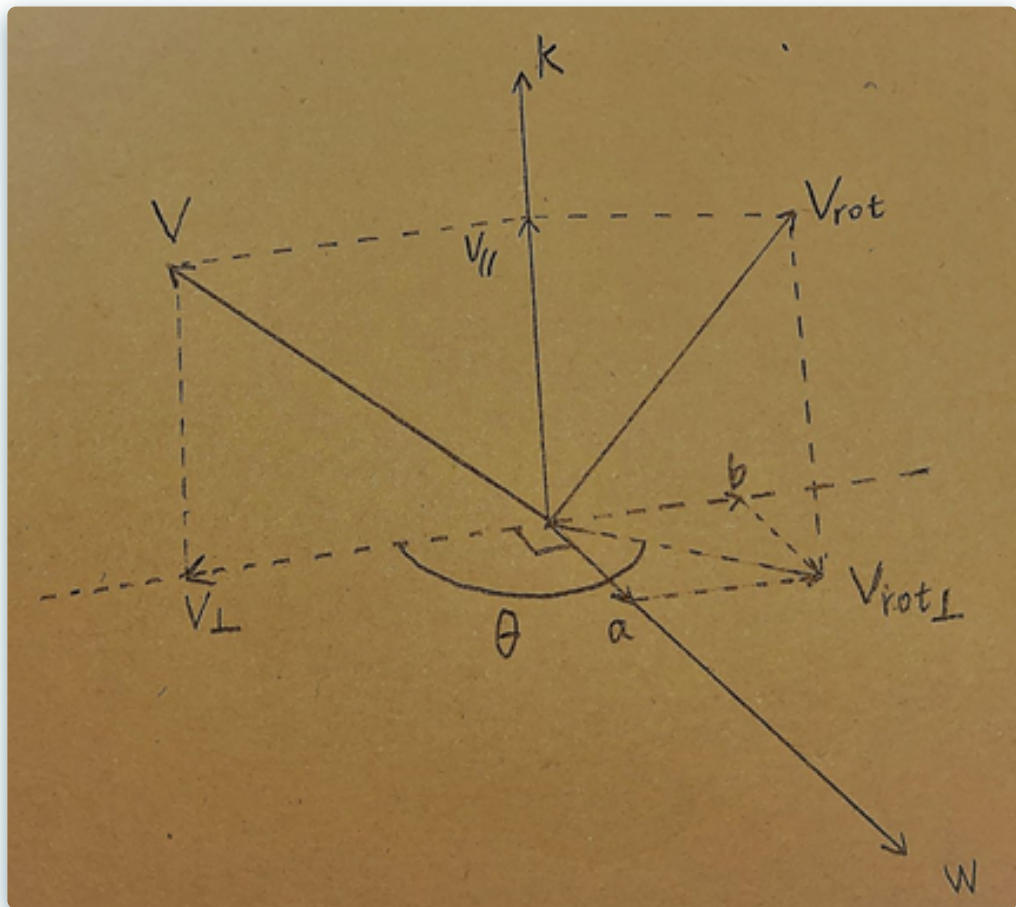
因为旋转过程中，跟旋转轴平行的向量是不改变的，所以：

$$v_{rot\parallel} = v_{\parallel}$$

为计算方便，将 $v_{rot\perp}$ 进行分解：

$$v_{rot\perp} = a + b$$

示意图如下图所示：



由图中的向量关系经过推导可得： $b = \cos \theta v_{\perp}$; $a = \sin \theta k \times v$

综上可得：

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

以上为罗德里格斯的矢量运算表达式，下面需要将其转化为矩阵表达式：

$k \times v$ 可用叉乘矩阵来化简为 Kv

$$\begin{bmatrix} (k \times v)_x \\ (k \times v)_y \\ (k \times v)_z \end{bmatrix} = \begin{bmatrix} k_y v_z - k_x v_y \\ k_z v_x - k_x v_z \\ k_x v_y - k_y v_x \end{bmatrix} = \begin{bmatrix} 0 & -k_z & k_y \\ k_z & 0 & -k_x \\ -k_y & k_x & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

所以,

$$\begin{aligned} & \cos \theta v + (1 - \cos \theta)(v \cdot k)k + \sin \theta k \times v \\ &= \cos \theta v + (1 - \cos \theta)kk^T v + \sin \theta Kv \\ &= (\cos \theta I + (1 - \cos \theta)kk^T + \sin \theta k^\times)v \\ &= Rv \end{aligned}$$

旋转矩阵 R 得证:

$$R = \cos \theta I + (1 - \cos \theta)nn^T + \sin \theta n^\wedge.$$

2. 使用罗德里格斯公式证明 $R^{-1} = R^T$

证明:

若要证明 $R^{-1} = R^T$, 只需证明 $R^T R = I$ 即可, 因为 $R^{-1} R = I$

$$\begin{aligned} R^T &= (\cos \theta I + (1 - \cos \theta)nn^T + \sin \theta n^\wedge)^T \\ &= \cos \theta I + (1 - \cos \theta)nn^T + \sin \theta (n^\wedge)^T \end{aligned}$$

所以可计算 $R^T R$:

$$\begin{aligned} R^T R &= (\cos \theta I + (1 - \cos \theta)nn^T + \sin \theta (n^\wedge)^T) \cdot (\cos \theta I + (1 - \cos \theta)nn^T + \sin \theta n^\wedge) \\ &= (\cos \theta)^2 I + \cos \theta (1 - \cos \theta) I nn^T + \cos \theta \sin \theta I n^\wedge \\ &\quad + \cos \theta (1 - \cos \theta) nn^T I + (1 - \cos \theta)^2 (nn^T)^2 + \sin \theta (1 - \cos \theta) nn^T n^\wedge \\ &\quad + \sin \theta \cos \theta n \times I + \sin \theta \cos \theta (n^\wedge)^T nn^T + (\sin \theta)^2 (n^\wedge)^T n^\wedge \\ &= \end{aligned}$$

暂时没有化简出来

✓ 7. 四元数运算性质的验证

验证 p' 必定为虚四元数 (实部为零)

证明:

$$\text{已知 } p = [0, x, y, z]^T = [0, \mathbf{v}]^T$$

$$\text{假定四元数 } q = [s, x, y, z]^T$$

使用《视觉SLAM十四讲》第二版课本P59页中的公式(跟本次习题第5题中的结果类似), 可得:

$$\begin{aligned} p' &= qpq' \\ &= q^+ p^+ q^{-1} \\ &= q^+ (q^{-1})^\oplus p \\ &= Qp \end{aligned}$$

带入两个符号所对应的矩阵, 得:

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

矩阵 Q 为:

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

因此,

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

可以看出, 旋转后的点 p' 的实部 s 为0, 故为纯虚四元数, 得证。

✓ 8. *熟悉C++11

代码中用到c++11标准的地方:

第15行中 `vector<avec{a1, a2, a3}>;` 这里使用了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。

第17行: 范围for语句, 这是c++11新标准引入的一种更简单的for语句, 其语法形式是:

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


第17行中auto : auto类型说明符; 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。

第16行中的 {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:

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

习题讲解后补充:

✓ 2. Eigen矩阵运算

如果矩阵A可逆, AA^T 才是正定矩阵。否则只能说明 AA^T 为半正定。这里Cholesky分解: `llt()`(正定矩阵), `ldlt()`(半正定/半负定)。所以代码里使用`ldlt`函数比较合理。

✓ 6. 使用罗德里格斯公式证明 $R^{-1} = R^T$

第二问: 一般方法:

①写出 R^T ,

$$\begin{aligned} R^T &= (\cos \theta I + (1 - \cos \theta) k k^T + \sin \theta k^\wedge)^T \\ &= \cos \theta I + (1 - \cos \theta) k k^T - \sin \theta k^\wedge \end{aligned}$$

($(k^\wedge)^T = -k^\wedge$ 这一步非常重要, 能够使下一步展开的时候少好几项)

②证明 $RR^T = I$

化简过程中注意使用 $k^\wedge k^\wedge = k k^T - I$ (书P. 78 式4. 20)

一种巧妙的方法

$$\begin{aligned} R^T &= (\cos \theta I + (1 - \cos \theta) k k^T + \sin \theta k^\wedge)^T \\ &= \cos \theta I + (1 - \cos \theta) k k^T - \sin \theta k^\wedge \end{aligned}$$

R^{-1} 表达的是绕同一根旋转轴反方向转动。

$$\begin{aligned} R^{-1} &= \cos -\theta I + (1 - \cos -\theta) k k^T + \sin -\theta k^\wedge \\ &= \cos \theta I + (1 - \cos \theta) k k^T - \sin \theta k^\wedge \end{aligned}$$