第六讲-习题

姜帆

2019 年 7 月 24 日

目录

1 证明题

寻找最小二乘解:

$$min_y = \|Dy\|_2^2 \qquad s.t. \|y\| = 1 \tag{1}$$

对 D^TD 做 SVD 分解得到下式:

$$D^TD = \sum_{i=1}^{4} \sigma_i^2 u_i u_j^T \tag{2}$$

其中 σ^2 为 D^TD 的奇异值, u_i 为 D^TD 的奇异向量。D^TD 为对称矩阵, 因此奇异值分解与特征值分解得到的结果相同。根据 SVD 分解推导证明过程可以知道, 对 D 进行 SVD 分解, 奇异值为对 D^TD 做特征值分解得到的特征值的平方根, 特征向量 V 为右奇异向量, 其中大于 $k = rank(D)$ 维度的分量都在 D 的零空间中, 即 $AV_i = 0(i > k)$, 也就是说 u_4 是 $Ay = 0$ 的解。

当取 $y = u_4$ 时, 即取奇异值 σ_4^2 最小对应的奇异向量作为最小二乘的解, 此时有:

$$\begin{aligned}
&\|Dy\|_2^2 \\
&= y^TD^TDy \\
&= u_4^TD^TDu_4 \\
&= \sigma_4^2 u_4^T u_4 \\
&= \sigma_4^2
\end{aligned} \tag{3}$$

此时目标函数取得最小。因此 $y = u_4$ 为 $min_y = \|Dy\|_2^2$ 的解。

2 代码实践题

对 D^TD 做 SVD 分解，提取奇异值最小对应的奇异向量（由于 D^TD 为对称矩阵，左右奇异向量相同，且对 D^TD 做特征值分解得到的特征值、特征向量和做奇异值分解得到的奇异值和奇异向量相同）即为最小二乘的解，约掉第四维后便得到路标点的坐标。

```cpp
gulate.cpp ▸ ...
    Eigen::Vector3d P_est;           // 结果保存到这个变量
    P_est.setZero();
    /* your code begin */
    const int dim = 2*(end_frame_id - start_frame_id);
    Eigen::MatrixXd D(Eigen::MatrixXd::Zero(dim, 4));
    for(int i = start_frame_id;i < poseNums; i++)
    {
        Eigen::Matrix4d T = Eigen::Matrix4d::Identity();
        T.block(0, 0, 3, 3) = camera_pose[i].Rwc.transpose();
        T.block(0, 3, 3, 1) = -camera_pose[i].Rwc.transpose()*camera_pose[i].twc;

        int start_row = i - start_frame_id;
        D.block(2*start_row, 0, 1, 4) = camera_pose[i].uv(0)*T.row(2) - T.row(0);
        D.block(2*start_row+1, 0, 1, 4) = camera_pose[i].uv(1)*T.row(2) - T.row(1);
    }
    //D^TD特征值分解
    Eigen::Matrix4d T= D.transpose()*D;
    Eigen::EigenSolver<Eigen::Matrix4d> es(T);
    Eigen::Matrix4cd evector = es.eigenvectors();
    Eigen::Vector4cd evalue = es.eigenvalues();
    std::cout<<"------特征值分解---------"<<std::endl<<"evector:"<<evector<<std::endl<<"evalue"<<evalue<<std::endl;

    //D^TD SVD 分解
    Eigen::MatrixXd A = D.transpose()*D;
    Eigen::JacobiSVD<Eigen::MatrixXd> svd(A, Eigen::ComputeFullU | Eigen::ComputeFullV);
    Eigen::MatrixXd U,V,sigma;
    U = svd.matrixU();
    V = svd.matrixV();
    sigma = svd.singularValues();
    std::cout <<"------奇异值分解---------"<< "U:"<< U <<std::endl<<"V:"<< V<<std::endl<< "singularValues:"<< sigma<<std::endl;
    if(sigma(3)/sigma(2) > 1e-3)
        std::cout<<"三角化失败! "<<std::endl;

    Eigen::Vector4d u4 = V.col(3);
    P_est = u4.head(3)/u4(3);
    /* your code end */

    std::cout <<"ground truth: \n"<< Pw.transpose() <<std::endl;
    std::cout <<"your result: \n"<< P_est.transpose() <<std::endl;
```

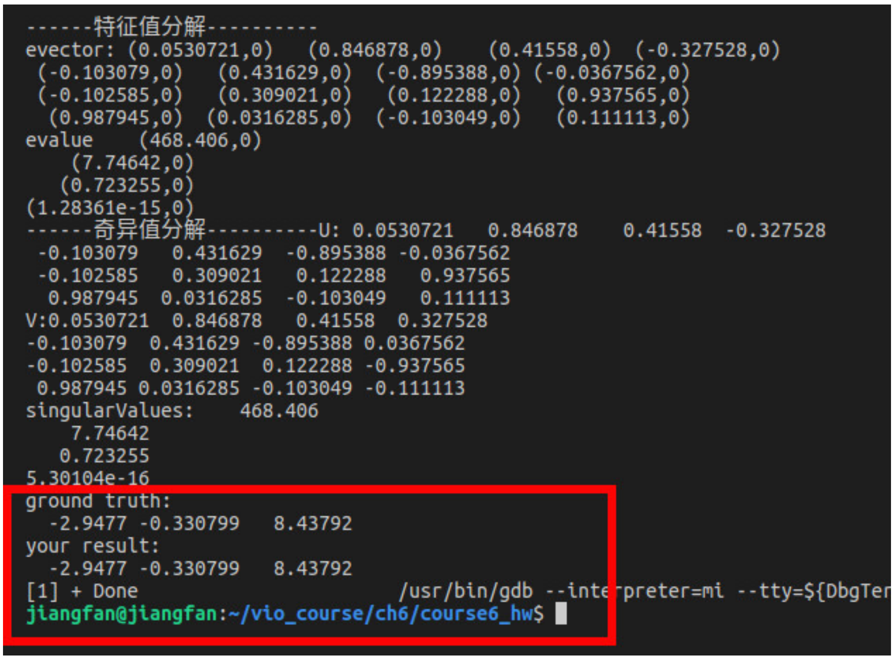
图 1: 代码补充

对 D^TD 做特征值分解以及做奇异值分解结果，可以看出奇异值与特征值，奇异向量与特征向量相等：

```
问题   输出   调试控制台   终端

------特征值分解----------
evector: (0.0530721,0)   (0.846878,0)    (0.41558,0)  (-0.327528,0)
 (-0.103079,0)   (0.431629,0)  (-0.895388,0) (-0.0367562,0)
 (-0.102585,0)   (0.309021,0)    (0.122288,0)    (0.937565,0)
  (0.987945,0)  (0.0316285,0)  (-0.103049,0)    (0.111113,0)
evalue    (468.406,0)
    (7.74642,0)
   (0.723255,0)
(1.28361e-15,0)
------奇异值分解---------U: 0.0530721   0.846878    0.41558  -0.327528
 -0.103079   0.431629  -0.895388 -0.0367562
 -0.102585   0.309021   0.122288   0.937565
  0.987945  0.0316285  -0.103049   0.111113
V:0.0530721  0.846878   0.41558  0.327528
-0.103079  0.431629 -0.895388 0.0367562
-0.102585  0.309021  0.122288 -0.937565
 0.987945 0.0316285 -0.103049 -0.111113
singularValues:    468.406
    7.74642
   0.723255
5.30104e-16
```

图 2: 特征值分解和奇异值分解结果



```
------特征值分解----------
evector: (0.0530721,0)   (0.846878,0)    (0.41558,0)  (-0.327528,0)
 (-0.103079,0)   (0.431629,0)  (-0.895388,0) (-0.0367562,0)
 (-0.102585,0)   (0.309021,0)   (0.122288,0)   (0.937565,0)
  (0.987945,0)  (0.0316285,0)  (-0.103049,0)   (0.111113,0)
evalue    (468.406,0)
    (7.74642,0)
   (0.723255,0)
(1.28361e-15,0)
------奇异值分解---------U: 0.0530721   0.846878    0.41558  -0.327528
 -0.103079   0.431629  -0.895388 -0.0367562
 -0.102585   0.309021   0.122288   0.937565
  0.987945  0.0316285  -0.103049   0.111113
V:0.0530721  0.846878   0.41558  0.327528
-0.103079  0.431629 -0.895388 0.0367562
-0.102585  0.309021  0.122288 -0.937565
 0.987945 0.0316285 -0.103049 -0.111113
singularValues:    468.406
    7.74642
   0.723255
5.30104e-16
ground truth:
  -2.9477 -0.330799   8.43792
your result:
  -2.9477 -0.330799   8.43792
[1] + Done                       /usr/bin/gdb --interpreter=mi --tty=${DbgTer
jiangfan@jiangfan:~/vio_course/ch6/course6_hw$
```

图 3: 路标点坐标计算结果