

1 scipy.spatial.transform

1.1 Rotation

这个类提供了一个接口来初始化和表示旋转

1.1.1 as_rotvec

旋转向量是与旋转轴同向的 3 维向量，其范数给出旋转角度

给定旋转角

```
1 from scipy.stats import special_ortho_group
2 from scipy.spatial.transform import Rotation
3
4 rand_rot = special_ortho_group.rvs(3)      #随机的旋转矩阵
5 axis_rot= Rotation.from_matrix(rand_rot)    #初始化生成的旋转矩阵
6 axis_angle = Rotation.as_rotvec(axis_rot)    #旋转角度
7
8 print(axis_angle)
9
10 #OUTPUT:
11 #      [-0.01449134  0.30742885  1.34225116]
```

1.1.2 as_matrix

可以使用旋转矩阵表示 3D 旋转，旋转矩阵是 3x3 实正交矩阵，行列式等于 +1

整体例子

```
1 from scipy.spatial.transform import Rotation
2 from scipy.stats import special_ortho_group
3
4 rand_rot = special_ortho_group.rvs(3)
5 axis_angle = Rotation.as_rotvec(Rotation.from_matrix(rand_rot))
6 print(axis_angle)
7 rot_mag = 45
8 axis_angle *= rot_mag / 180.0
9 print(axis_angle)
10 rand_rot = Rotation.from_rotvec(axis_angle)
11 rand_rot1 = rand_rot.as_matrix()    #表示单次旋转
12 print(rand_rot1)
13
14 #OUTPUT:
15 #      [-0.94779163  2.85219495  0.34512507]
16 #
17 #      [-0.23694791  0.71304874  0.08628127]
```

```

18 #
19 #      [[ 0.75412225 -0.15881423  0.63724224]
20 #      [-0.00223926  0.96969261  0.24431786]
21 #      [-0.65673025 -0.18567249  0.73091115]]

```

1.1.3 from_matrix

3 维旋转矩阵可以用 3 x 3 适当的正交矩阵表示。

如果输入不是适当的正交，则使用中描述的方法创建近似值。

生成正交矩阵

```

1 from scipy.stats import special_ortho_group
2 from scipy.spatial.transform import Rotation
3
4 rand_rot = special_ortho_group.rvs(3)      #随机的旋转矩阵
5 axis_rot = Rotation.from_matrix(rand_rot)  #初始化生成的旋转矩阵

```

1.1.4 from_rotvec

2 scipy.stats

2.1 special_ortho_group

返回从 haar 分布 ($SO(n)$ 上唯一的均匀分布) 绘制的随机旋转矩阵

绘制随机旋转矩阵

```

1 from scipy.stats import special_ortho_group
2
3 x = special_ortho_group.rvs(3)      #指定矩阵大小,随机生成矩阵
4 print(x)
5
6 #OUTPUT:
7 #      [[-0.48455797  0.03507638  0.87405562]
8 #      [-0.2471805   0.95298494 -0.17527551]
9 #      [-0.83910987 -0.30098065 -0.45310625]]

```