Chapter 2：

运行时能够产生变换后的四边形的四个坐标以及直线的坐标，但是在图形上显示不出来：



图1 Chapter 2 运行结果

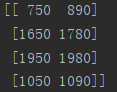
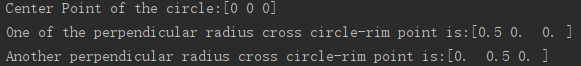


图2 四边形经过变换后的四个坐标



图3 直线的两个坐标



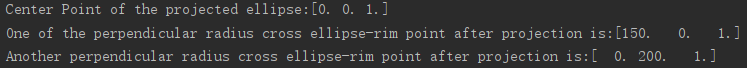


图4 将圆形投射后成为椭圆（取长轴和短轴与椭圆的交点）

做圆形投影成椭圆时，只是简单地将原来圆的半径上，取垂直的两个线与圆的两个交点，将这两个点与中心分别进行变换，得出的结果要计算出中心点与投影后两个椭圆上的点的距离作为长轴和短轴（只取x, y轴）这一做法我自己没有证明过是否正确，只是直观地感觉到可以这样。

疑惑：1、在类中编写class …. （object）时，（object）的作用具体是什么？想要知道一个类中的属性或者方法引用的参数是什么类型时，都是主函数中找到应用的语句来确定，再一层一层套回去，感觉与以前接触的类定义，简单地实例创造的那种方法不一样。

2、在这个相机模型里面使用的旋转rotation，在进行运算时都是使用默认的旋转矩阵作为参数（np.eye(3)），能不能有可以让自己输入的方式来进行？（担心破坏程序结构我没有去尝试）

3、Geometry.py里的函数我都自己测试过，能够实现对应的功能。老师的代码里面很多return返回值都是固定的，我将一些改了，但也担心这是在todo代码的外面。

Chapter 4:

从结果上看并没有与两个原来的轨迹拟合出一条特别吻合的轨迹，这里面我不太懂的是权重矩阵的选取的问题。由于只能用到一个参数arc\_lengths，亦即曲线的长度，联系之前老师讲过的协方差和Fisher Information Matrix等信息，我其实没有弄懂，主要因为这个单一参数的问题，不知道是否应该引入其他参数，为了使近点可信度高，远点可信度低，只采用了两种很简单的方法，一种是线性的变换，一种是非线性的。效果如下图：

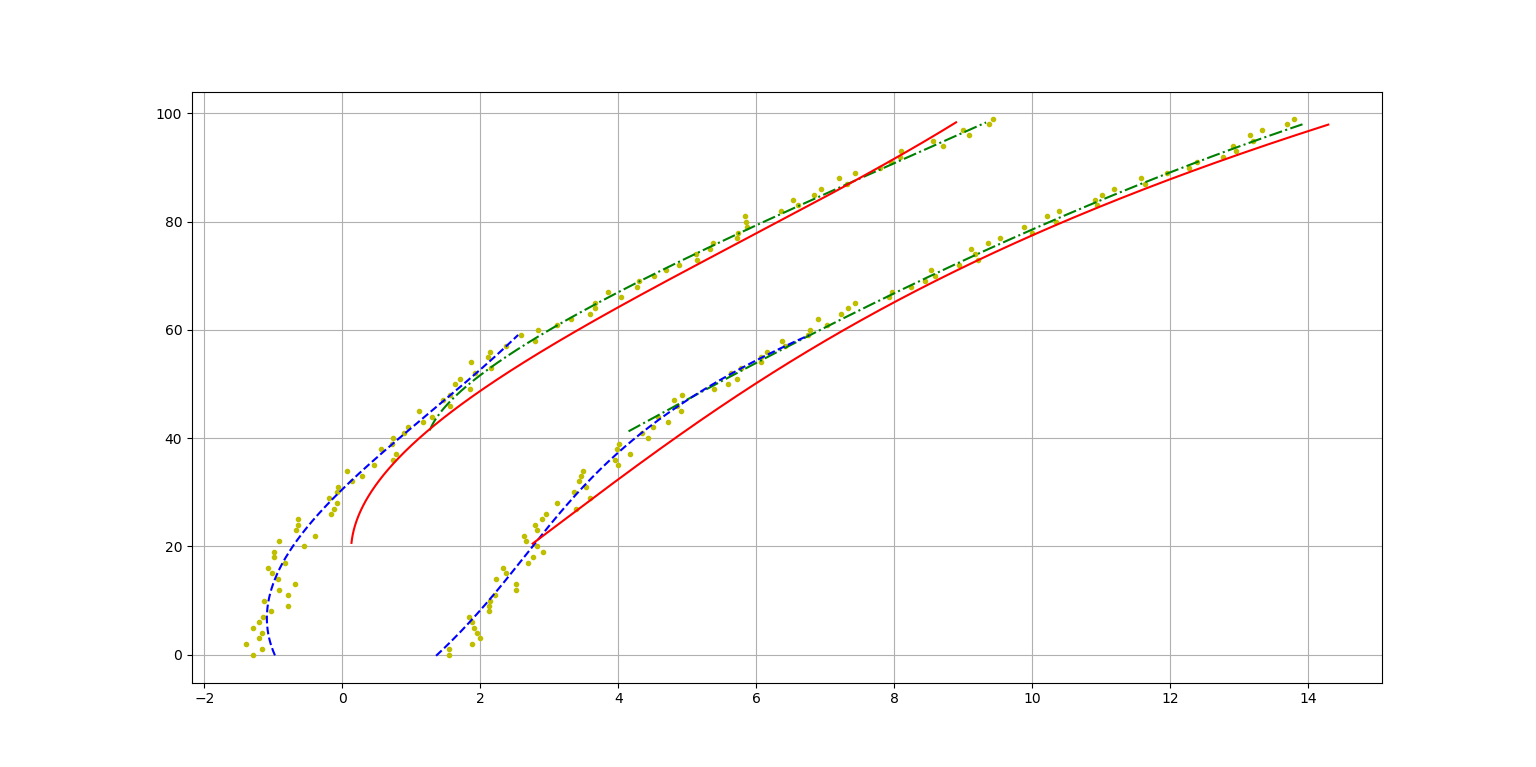


图1 采用与arc\_lengths非线性关系的权重值

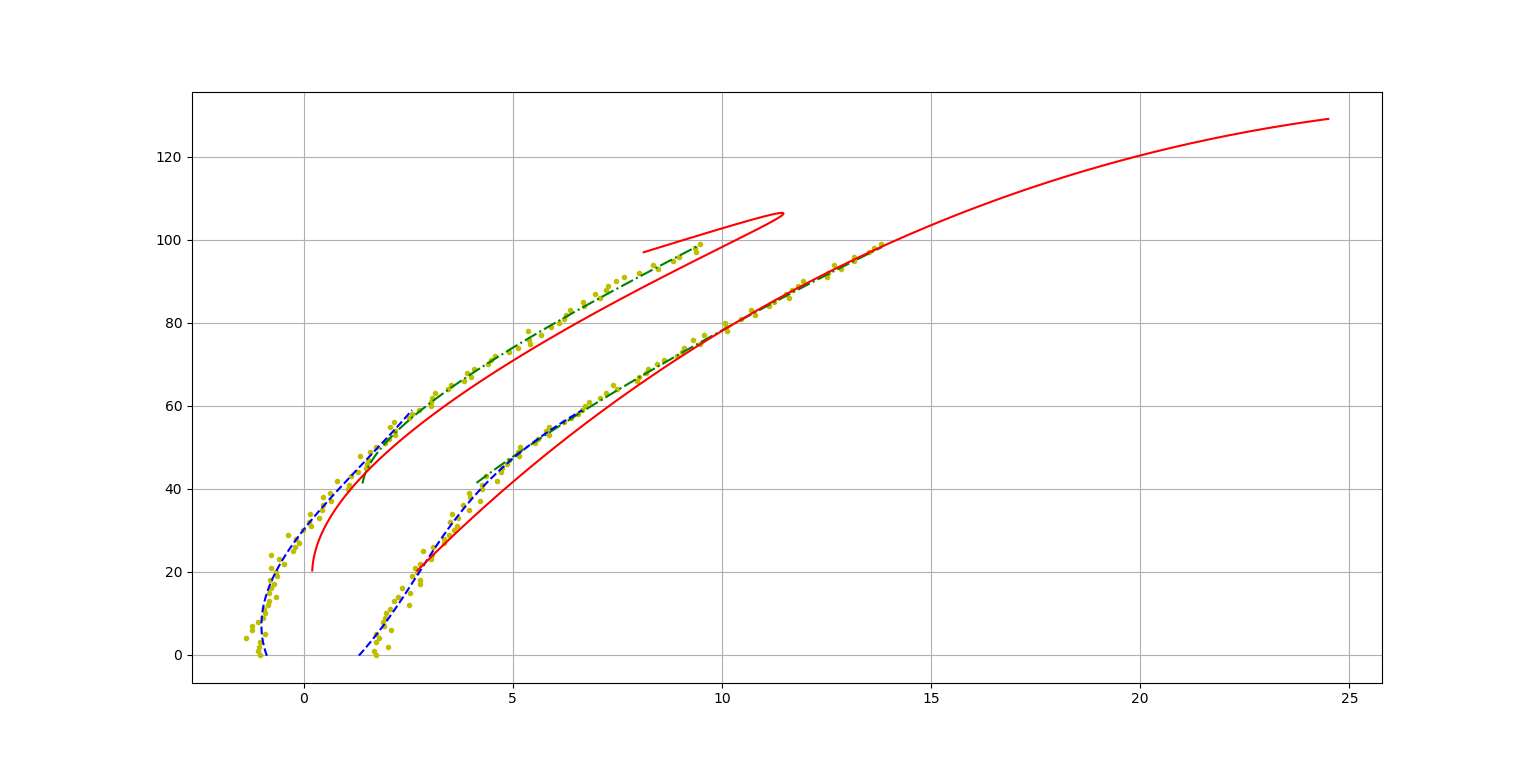


图2 采用与arc\_lengths线性关系的权重值

寻找起始点是用最短距离去完成的，将参数中的一个周期的lanemarking转成一系列散点，找到具有最短的欧式距离，完成s的选取。