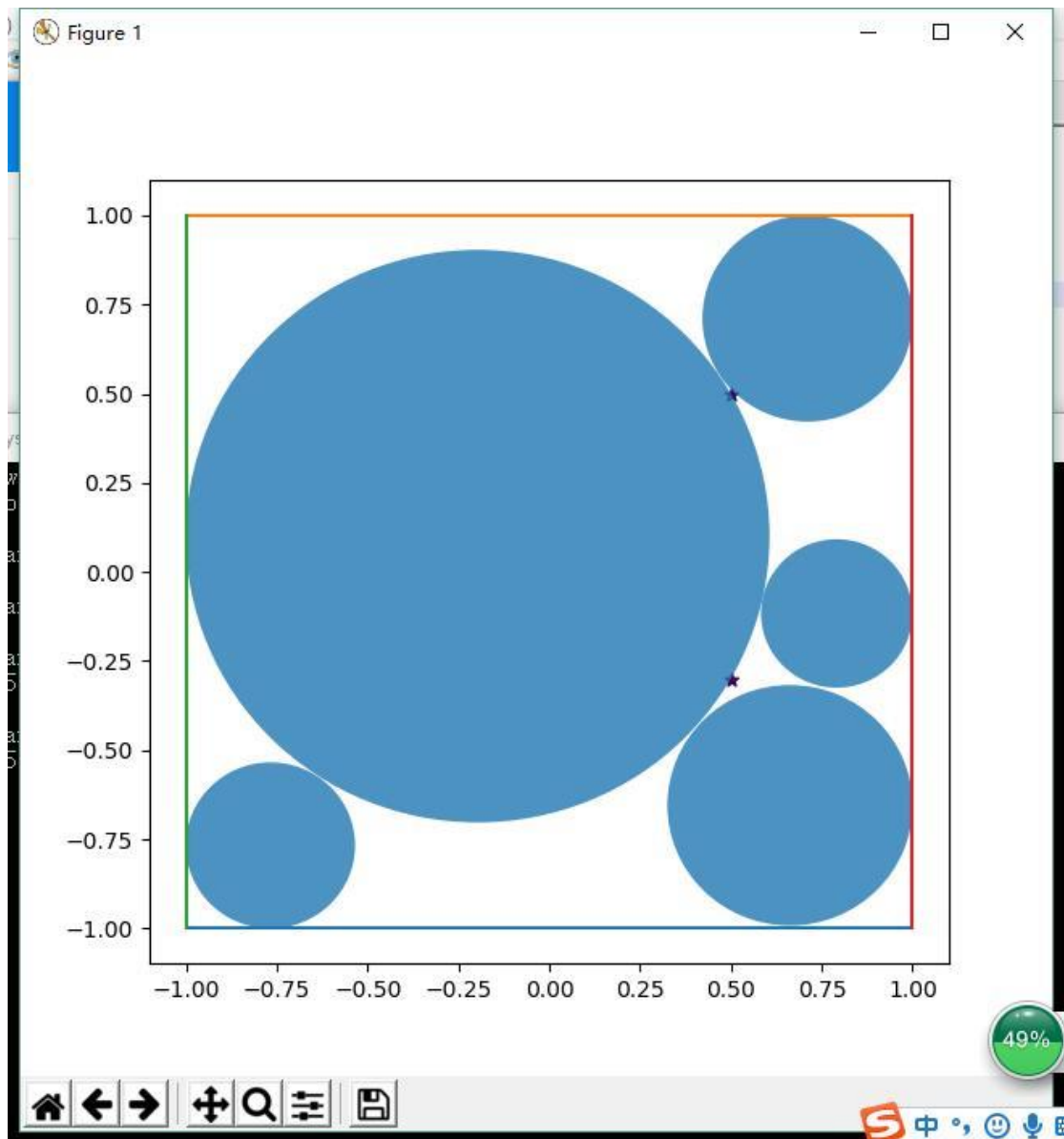


1. user story ----warm_up project 找寻最大圆面积和
作为程序的使用者，我想根据我输入的气球个数和随机障碍物个数，得到无覆盖的最大圆面积和；
作为程序的使用者，我想能够在视觉上看到气球填充方框的可视化窗口；
作为程序的使用者，我知道方框在 $(-1, 1)$ 之间且气球半径 r 和位置 μ 可变；
作为程序的使用者，我希望可以有良好的交互功能方便我操作。
2. tasks :
 - 1) 做一个 $(-1, 1)$ 的框
 - 2) 在框里面放一些随机障碍物
 - 3) 气球不能与其他气球或障碍物重叠，也不能越过边框
 - 4) 找出使所有气球半径 r^2 和最大的圆
 - 5) 要用 python 编写程序
3. 算法：
将 $[-1, 1]$ 平均分成 250×250 (或其他精度) 个点，用 list 进行枚举，将 n 个障碍物当作已选择的圆 (半径为 0)。把 list 里的这些点分别作为圆心去遍历，让每个点都找到以它为圆心的最大半径，在所有点中取最大半径最大的那一个，确定该点为已选择的圆并进行下一次遍历。直至找不出下一个圆为止，此时 r^2 的和就是我们找到的最大和。
4. test cases:
 - 1) $m = 5$

```
num_of_circle = 5
blockers = [(0.5, 0.5)
            , (0.5, -0.3)]

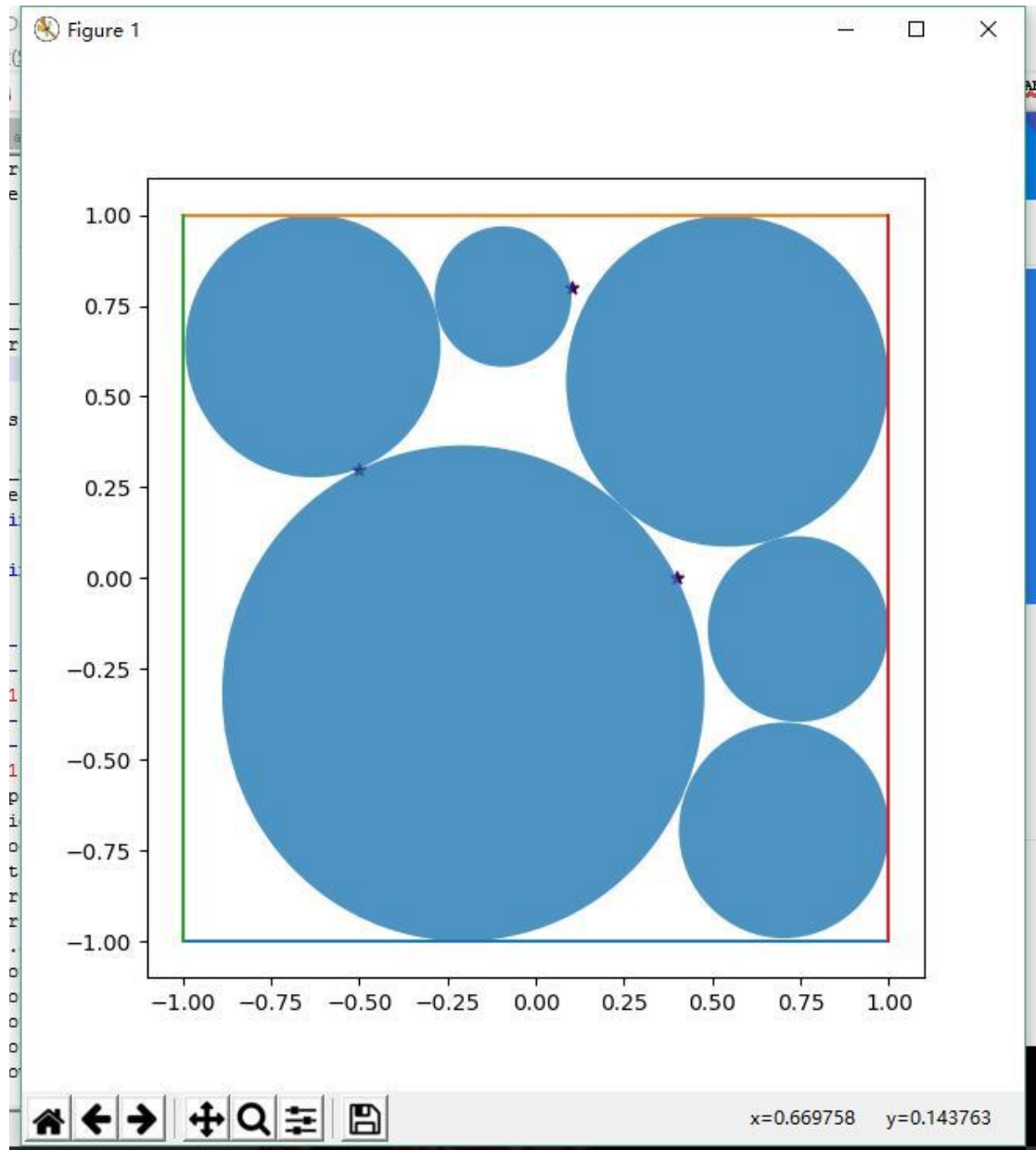
C:\Users\qianqian\Desktop\ball_in_box\ball_in_box>python area_sum.py
Total area: 2.953716234702045
```



2) $m = 6$

```
num_of_circle = 6
blockers = [(0.1, 0.8)
            , (-0.5, 0.3), (0.4, 0)]
```

```
C:\Users\qianqian\Desktop\ball_in_box\ball_in_box>python area_sum.py
Total area: 3.128730608255651
```



3) $m = 10$

```
num_of_circle = 10
blockers = [(0, 0)
            , (-0.5, 0.3), (0.4, 0.7), (0.6, 0.9), (0.8, 0.1)]
```

```
C:\Users\qianqian\Desktop\ball_in_box\ball_in_box>python area_sum.py
Total area: 3.2549211721506923
```

