Lab10Report

April 9, 2018

```
In [35]: import numpy as np
         import matplotlib.pyplot as plt
         import matplotlib.cm as cm
         import pandas as pd
         import scipy.interpolate

         %matplotlib inline
```

0.1 Exercise 1

The output of Exercise 1, where the player does not change their initial choice: the probability of winning is 0.3333

0.2 Exercise 2

The output of Exercise 2, where the player does change their initial choice: the probability of winning is 0.6666

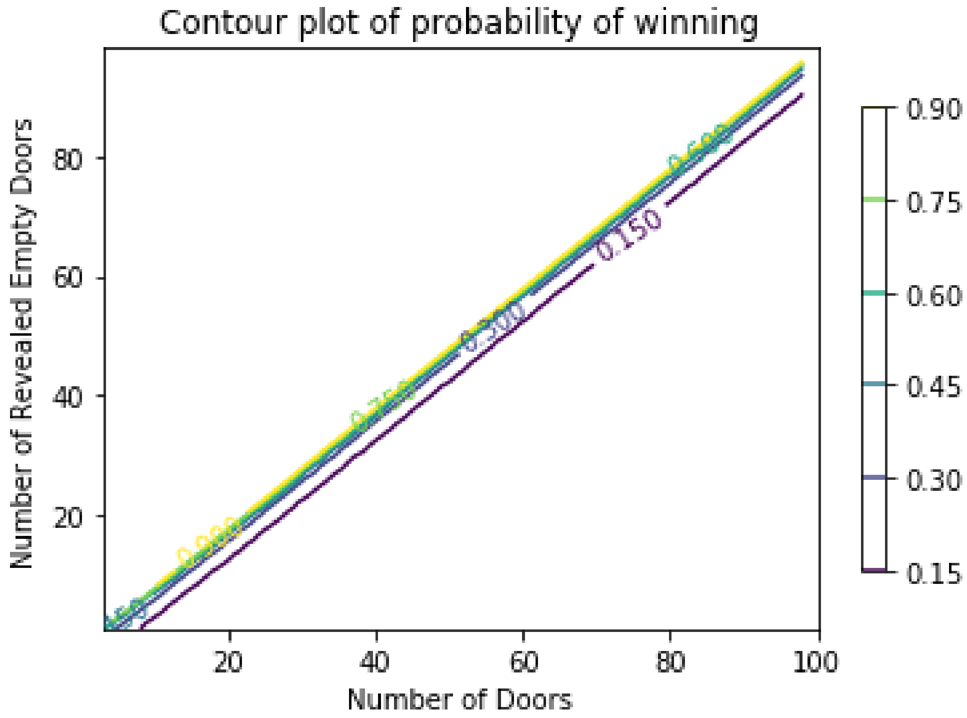
0.3 Exercise 3

```
In [7]: data = np.loadtxt("afile.txt")
```

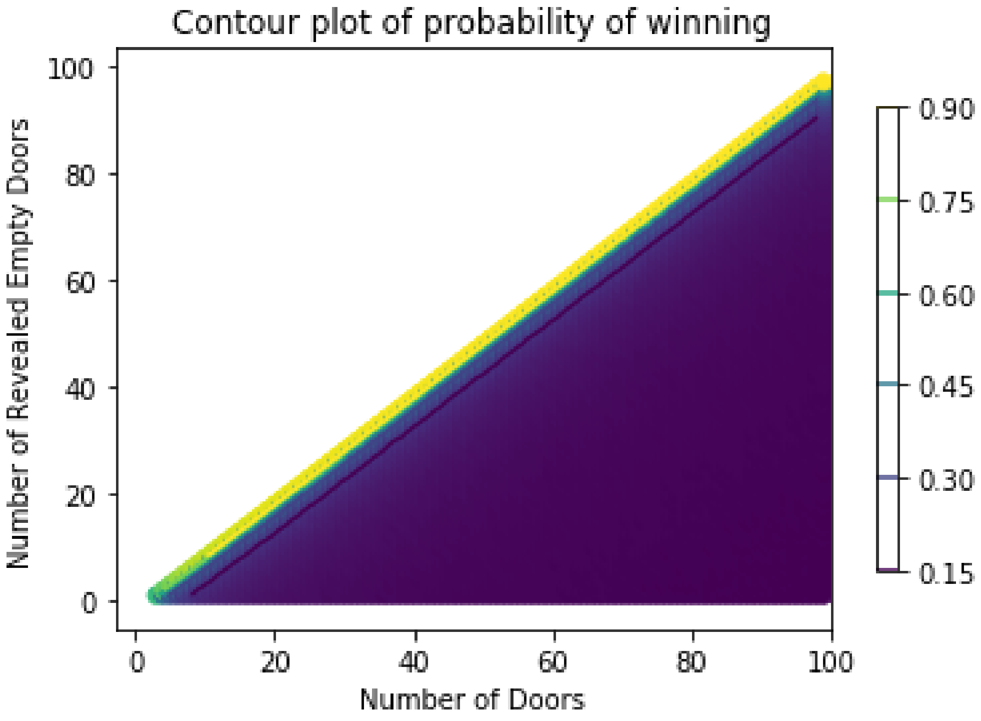
```
In [8]: m_mesh = np.linspace(3, 100, 100);
        n_mesh = np.linspace(1, 98, 100)

        m = data[:, 0]
        n = data[:, 1]
```

```
In [12]: z = data[:, 2]
         M, N = np.meshgrid(m_mesh, n_mesh)
         rbf = scipy.interpolate.griddata((m, n), z,(M, N), method='linear')
```

```
In [45]: fig=plt.contour(m_mesh, n_mesh, rbf)
         plt.clabel(fig, inline=1, fontsize=10)
         CB = plt.colorbar(fig, shrink=0.8, extend='both')
         plt.xlabel("Number of Doors")
         plt.ylabel("Number of Revealed Empty Doors")
         plt.title("Contour plot of probability of winning")
         plt.show()
```



```
In [44]: fig2=plt.contour(m_mesh, n_mesh, rbf)
         plt.scatter(m, n, c=z)
         CB = plt.colorbar(fig2, shrink=0.8, extend='both')
         plt.xlabel("Number of Doors")
         plt.ylabel("Number of Revealed Empty Doors")
         plt.title("Contour plot of probability of winning")
         plt.show()
```



Above are the two contour plots of the probability of winning, given the number of doors (m) and the number of empty doors revealed before the player switches their choice. Both graphs show that the closer the number of doors revealed to the amount of doors there are, there is a greater probability of winning, and even then, the chances are slim. The rest of the graph, besides those strong lines near slope =1, has a very low chance of winning, even if the player switches doors, which is logical.

0.4 Exercise 4

Unfortunately for Exercise 4, I wasn't able to output proper results, however, the amount of seconds still decreases slightly with additions of threads

Number of Threads	Time (seconds)
1	30.0576
2	30.0569
4	30.0561