newnb

April 13, 2017

IEOR 173 HW 10 Problem 4
Simulation Part b
Andrew Nichol

```
In [2]: from numpy import *
        from math import *

        gen_u = random.uniform

        def gen_exp(rate):
                def generate():
                        return -log(gen_u()) / rate
                return generate

        # arrival rate and server2 have rate 2 while server1 has rate 4
        arrival, leave_s1, leave_s2 = gen_exp(2), gen_exp(4), gen_exp(2)

        # sim initialization: (first entrance guaranteed)
        count_total = 1
        count_s1 = 1
        count_s2 = 0
        current_status = dict(s1 = True, s2 = 0)

        # 1 million loops
        for i in range(1000000):

            #generate rates
                arrival_result = arrival()
                leave_s1_result = leave_s1() if current_status['s1'] == True else i
                leave_s2_result = leave_s2() if current_status['s2'] == True else i
                get_min = min(arrival_result, leave_s1_result, leave_s2_result)

                if arrival_result == get_min:
                        count_total += 1
                        if current_status['s1'] == False:
                                count_s1 += 1
                                current_status['s1'] = True
                elif leave_s1_result == get_min:
                        current_status['s1'] = False
                        if current_status['s2'] == False:
                                count_s2 += 1
                                current_status['s2'] = True

                elif leave_s2_result == get_min:
                        current_status['s2'] = False

        print("Prop Enter: ", count_s1/count_total, '\n', 'Prop S2 :', count_s2/cou
```

```
Prop Enter:  0.6675649476131844
 Prop S2 : 0.44438507530142746
```

Thus, the proportion of customers who enter the second server converges to 4/9. The proportion who enter is as expected based on our solution to part a (2/3).