MT18052_A4_Q1

November 8, 2019

In [5]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
# reference:
# https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3059453/
# http://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Survival/BS704_Survival5.html
```

In [6]:
```
def getsurvival(risk,death,initialval = 1.0):
    risk = np.array(risk)
    death = np.array(death)
    survival = []
    survival.append(initialval)
    index = 1
    for i in range(1,len(risk)):
        tmp1 = survival[i-1] * ((risk[i] - death[i])/risk[i])
        survival.append(tmp1)
    return survival
def getexpectedevents(n1,o,n):
    events = []
    for i in range(len(n1)):
        events.append(n1[i] * (o[i]/n[i]))
    return events
```

In [7]:
```
table1 = pd.read_csv('paperdata/table1.csv')
table2 = pd.read_csv('paperdata/table2.csv')
table3d = pd.read_csv('paperdata/table3.csv')
```

In [8]:
```
table1.head(15)
```

Out[8]:

	Time	DG1	NG1
0	6	1	23
1	12	1	22
2	21	1	21
3	27	1	20
4	32	1	19
5	39	1	18
6	43	2	17
7	89	1	14

```
8    261    1    8
9    263    1   78
10   270    1    6
11   311    1    4
```

```
In [9]: table2.head(20)
```

```
Out[9]:    Time  DG2  NG2
0     9    1   23
1    13    1   22
2    27    1   21
3    38    1   20
4    49    2   18
5    93    1   15
6   126    1   12
7   218    1    9
8   301    1    5
9   333    1    4
```

```
In [10]: table3d.head(50)
```

```
Out[10]:     Time  TD  Patient_Died_G2   N  N2
0       6   1                0  46  23
1       9   1                1  45  23
2      12   1                0  44  22
3      13   1                1  43  22
4      21   1                0  42  21
5      27   2                1  40  21
6      32   1                0  39  20
7      38   1                1  38  20
8      39   1                0  37  19
9      43   2                0  36  19
10     49   2                2  32  18
11     89   1                0  31  16
12     93   1                1  29  15
13    126   1                1  25  12
14    218   1                1  19   9
15    261   1                0  17   8
16    263   1                0  15   7
17    270   1                0  14   7
18    301   1                1  11   6
19    311   1                0  10   5
20    333   1                1   9   4
```

```
In [11]: table3 = {'Time':[],
                   'N1':[],
                    'N2':[],
                    'N':[],
                    'O1':[],
```

```
          'O2':[],
          'O':[]
         }
i = 0
j = 0
data1 = np.array(table1)
data2 = np.array(table2)
index = 0
while(i < len(data1) and j < len(data2)):
    if(data1[i][0] < data2[j][0]):
        table3['N1'].append(data1[i][2])
        table3['N2'].append(0)
        table3['O1'].append(data1[i][1])
        table3['O2'].append(0)
        table3['Time'].append(data1[i][0])
        i+=1
    elif(data1[i][0] > data2[j][0]):
        table3['N2'].append(data2[j][2])
        table3['N1'].append(0)
        table3['O2'].append(data2[j][1])
        table3['O1'].append(0)
        table3['Time'].append(data2[j][0])
        j+=1
    else:
        table3['N1'].append(data1[i][2])
        table3['O1'].append(data1[i][1])
        table3['Time'].append(data1[i][0])
        table3['N2'].append(data2[j][2])
        table3['O2'].append(data2[j][1])
        j+=1
        i+=1
    table3['O'].append(table3['O1'][index] + table3['O2'][index])
    table3['N'].append(table3['N1'][index] + table3['N2'][index])
    index +=1
while(i < len(data1)):
    table3['N1'].append(data1[i][2])
    table3['N2'].append(0)
    table3['O1'].append(data1[i][1])
    table3['O2'].append(0)
    table3['Time'].append(data1[i][0])
    table3['O'].append(table3['O1'][index] + table3['O2'][index])
    table3['N'].append(table3['N1'][index] + table3['N2'][index])
    index +=1
    i+=1
while(j < len(data2)):
    table3['N2'].append(data2[j][2])
    table3['N1'].append(0)
    table3['O2'].append(data2[j][1])
```

```
            table3['O1'].append(0)
            table3['Time'].append(data2[j][0])
            table3['O'].append(table3['O1'][index] + table3['O2'][index])
            table3['N'].append(table3['N1'][index] + table3['N2'][index])
            index +=1
            j+=1
```

In [12]:
```
# table3df = pd.DataFrame(data=table3)
```

In [13]:
```
# table3df.head(50)
```

In [14]:
```
table3d.head()
```

Out[14]:

	Time	TD	Patient_Died_G2	N	N2
0	6	1	0	46	23
1	9	1	1	45	23
2	12	1	0	44	22
3	13	1	1	43	22
4	21	1	0	42	21

In [15]:
```
N1 = []
O1 = []
for i in range(len(table3d)):
    N1.append(table3d['N'][i] - table3d['N2'][i])
    O1.append(table3d['TD'][i] - table3d['Patient_Died_G2'][i])
```

In [16]:
```
table3d['N1'] = N1
table3d['O1'] = O1
table3d['O2'] = table3d['Patient_Died_G2']
table3d['O'] = table3d['TD']
table3d = table3d.drop(columns=['Patient_Died_G2','TD'])
```

In [17]:
```
table3d.head(50)
```

Out[17]:

	Time	N	N2	N1	O1	O2	O
0	6	46	23	23	1	0	1
1	9	45	23	22	0	1	1
2	12	44	22	22	1	0	1
3	13	43	22	21	0	1	1
4	21	42	21	21	1	0	1
5	27	40	21	19	1	1	2
6	32	39	20	19	1	0	1
7	38	38	20	18	0	1	1
8	39	37	19	18	1	0	1
9	43	36	19	17	2	0	2
10	49	32	18	14	0	2	2
11	89	31	16	15	1	0	1
12	93	29	15	14	0	1	1
13	126	25	12	13	0	1	1

```
14   218  19   9  10   0   1  1
15   261  17   8   9   1   0  1
16   263  15   7   8   1   0  1
17   270  14   7   7   1   0  1
18   301  11   6   5   0   1  1
19   311  10   5   5   1   0  1
20   333   9   4   5   0   1  1
```

```
In [18]: table3d['Expected1'] = getexpectedevents(n=table3d['N'],o=table3d['O'],n1=table3d['N1
         table3d['Expected2'] = getexpectedevents(n=table3d['N'],o=table3d['O'],n1=table3d['N2
```

```
In [19]: table3d.head(20)
```

Out[19]:

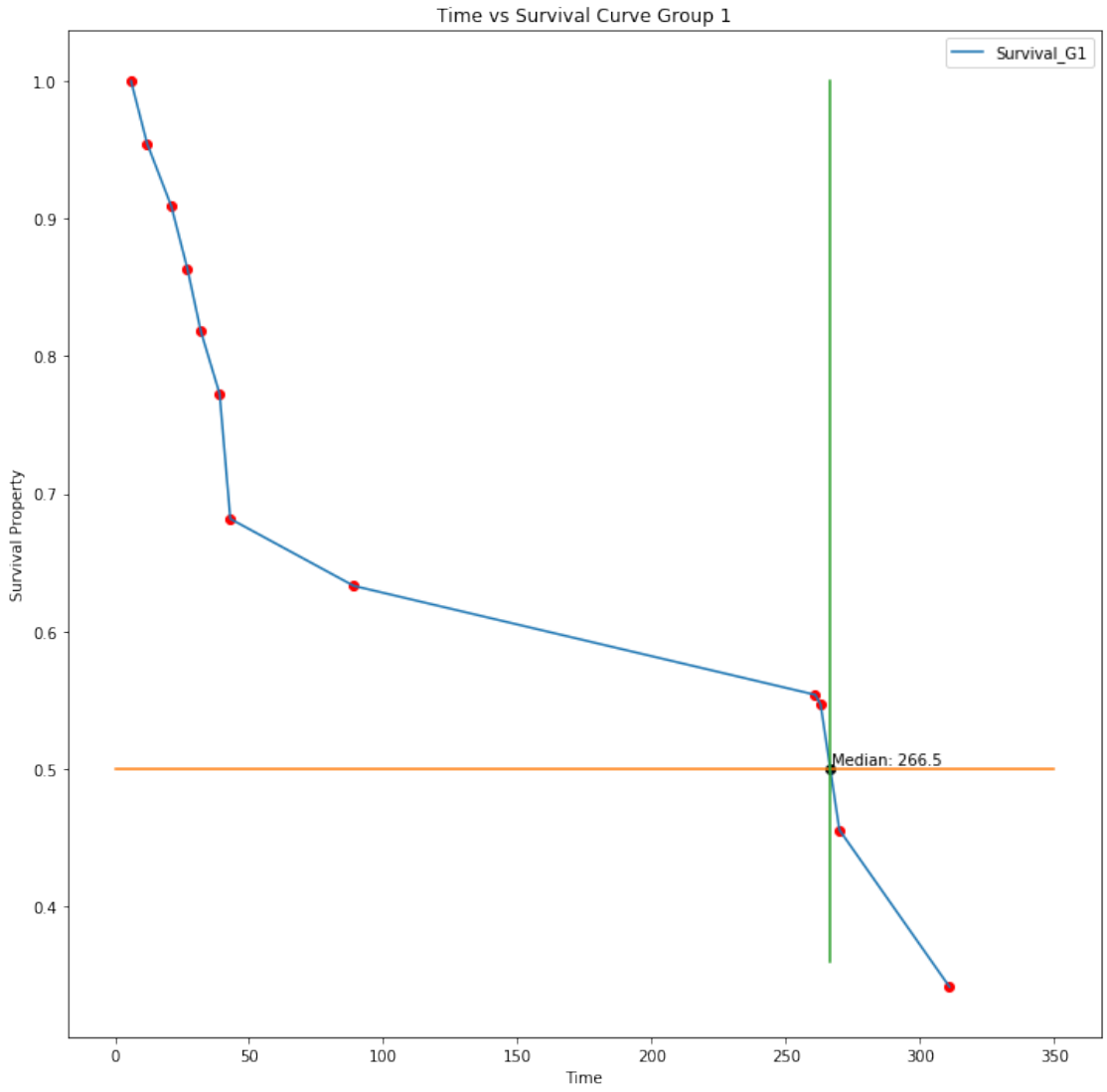
	Time	N	N2	N1	O1	O2	O	Expected1	Expected2
0	6	46	23	23	1	0	1	0.500000	0.500000
1	9	45	23	22	0	1	1	0.488889	0.511111
2	12	44	22	22	1	0	1	0.500000	0.500000
3	13	43	22	21	0	1	1	0.488372	0.511628
4	21	42	21	21	1	0	1	0.500000	0.500000
5	27	40	21	19	1	1	2	0.950000	1.050000
6	32	39	20	19	1	0	1	0.487179	0.512821
7	38	38	20	18	0	1	1	0.473684	0.526316
8	39	37	19	18	1	0	1	0.486486	0.513514
9	43	36	19	17	2	0	2	0.944444	1.055556
10	49	32	18	14	0	2	2	0.875000	1.125000
11	89	31	16	15	1	0	1	0.483871	0.516129
12	93	29	15	14	0	1	1	0.482759	0.517241
13	126	25	12	13	0	1	1	0.520000	0.480000
14	218	19	9	10	0	1	1	0.526316	0.473684
15	261	17	8	9	1	0	1	0.529412	0.470588
16	263	15	7	8	1	0	1	0.533333	0.466667
17	270	14	7	7	1	0	1	0.500000	0.500000
18	301	11	6	5	0	1	1	0.454545	0.545455
19	311	10	5	5	1	0	1	0.500000	0.500000

```
In [20]: sums= table3d.sum()
```

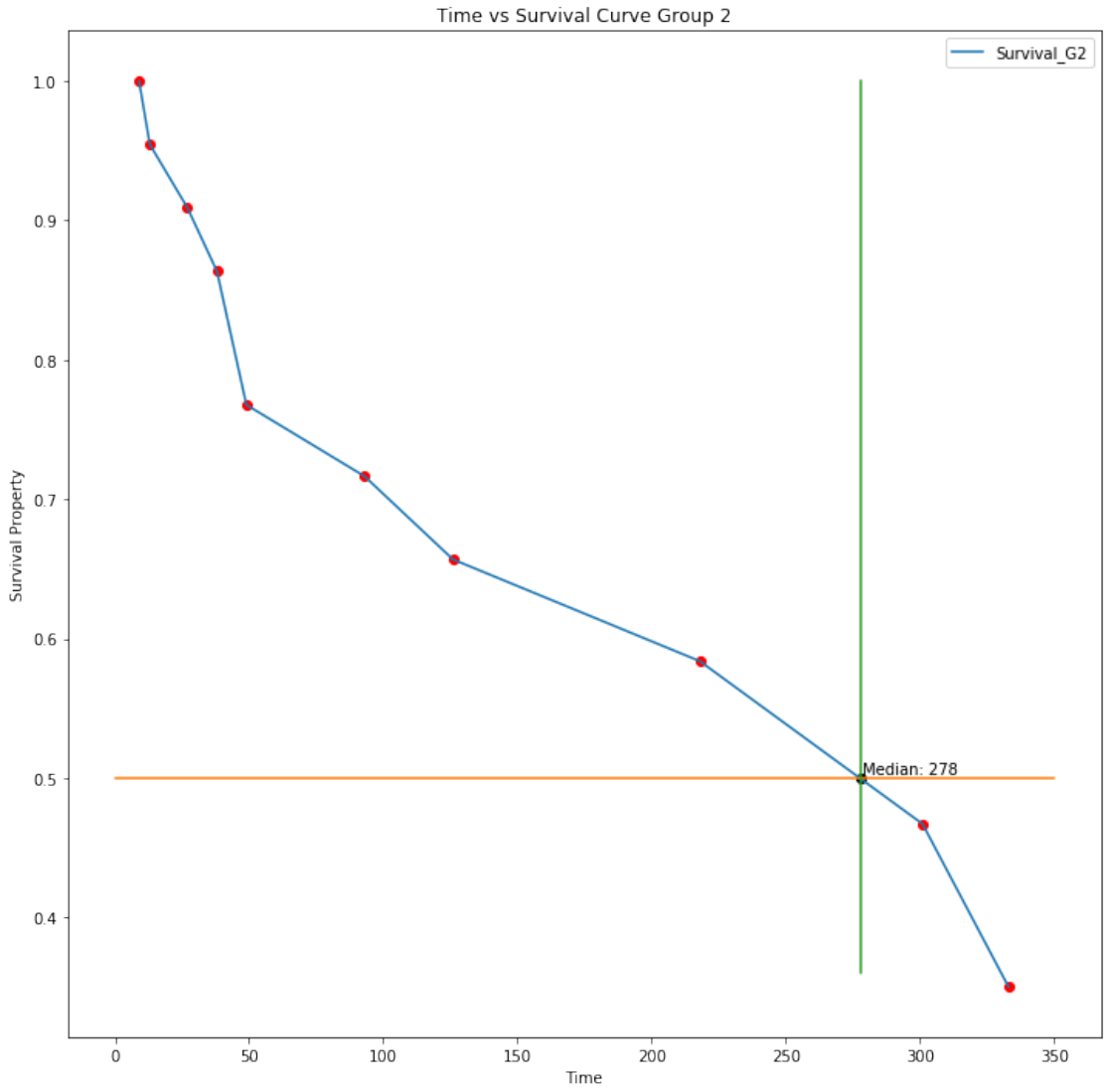
```
In [21]: sums
```

```
Out[21]: Time         2554.000000
         N             622.000000
         N2            317.000000
         N1            305.000000
         O1             13.000000
         O2             11.000000
         O              24.000000
         Expected1      11.779847
         Expected2      12.220153
         dtype: float64
```

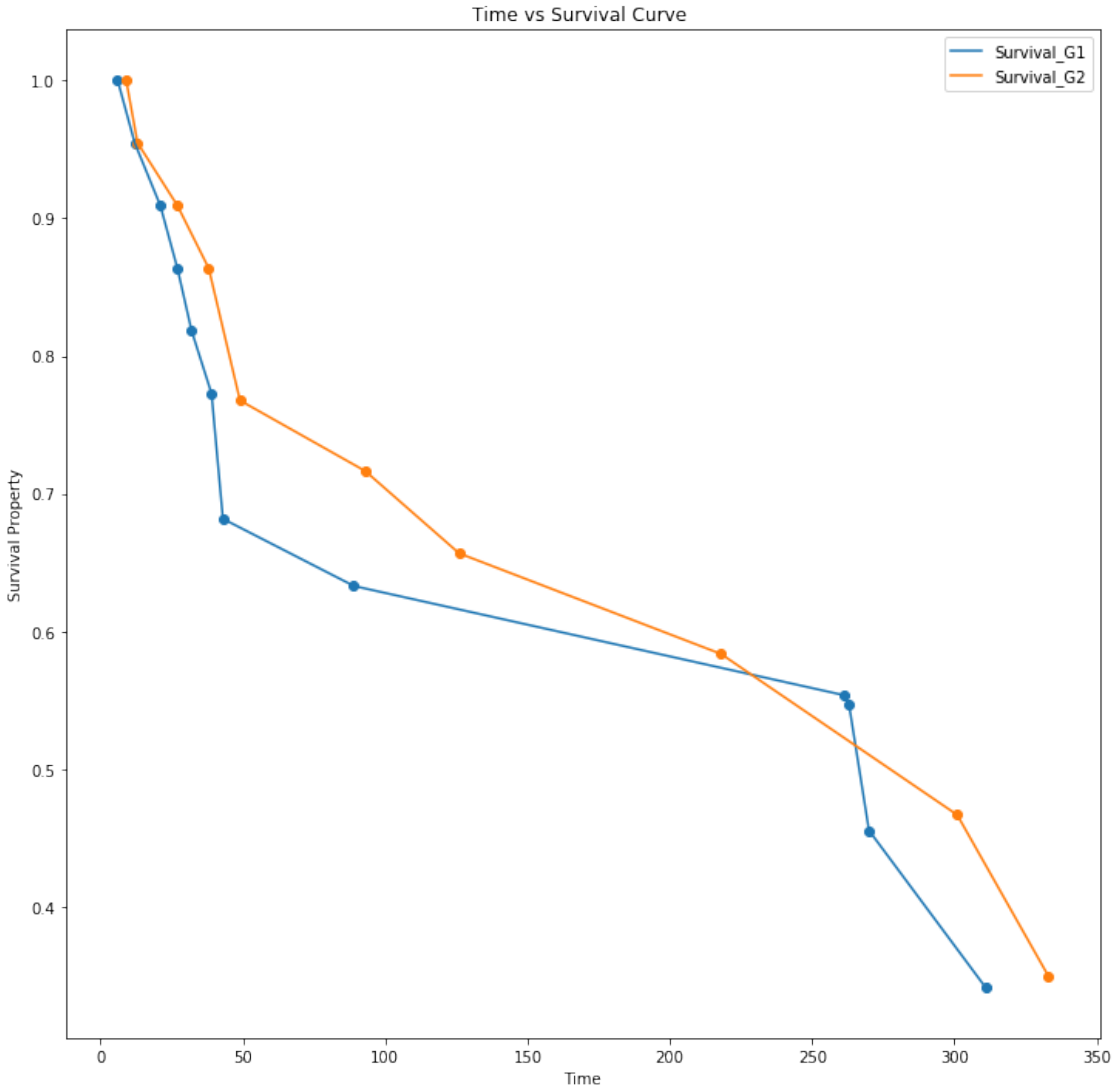
```
In [22]: X1 = ((sums['O1'] - sums['Expected1']) **2)/(sums['Expected1'])
         X2 = ((sums['O2'] - sums['Expected2']) **2)/(sums['Expected2'])
         X = X1 + X2

In [23]: print (X)

0.2482123942314774


In [24]: def kmplot(time,survival,title=""):
             plt.plot(time,survival)
             plt.xlabel("Time")

             plt.title("Time vs Survival Curve "+title)

             plt.ylabel("Survival Property")
         #     plt.scatter(time,survival)
             plt.legend()


In [25]: table1['Survival_G1'] = getsurvival(risk=table1['NG1'],death=table1['DG1'])
         table2['Survival_G2'] = getsurvival(risk=table2['NG2'],death=table2['DG2'])

In [26]: table1['Survival_G1']

Out[26]: 0     1.000000
         1     0.954545
         2     0.909091
         3     0.863636
         4     0.818182
         5     0.772727
         6     0.681818
         7     0.633117
         8     0.553977
         9     0.546875
         10    0.455729
         11    0.341797
         Name: Survival_G1, dtype: float64

In [40]: plt.figure(figsize=(12,12))
         kmplot(table1['Time'],table1['Survival_G1'],title="Group 1")
         plt.scatter(table1['Time'],table1['Survival_G1'],color='r')
         plt.plot([0,350],[0.5,0.5])
         plt.plot([266.5,266.5],[0.36,1])
         plt.scatter([266.5],[0.5],color='black')
         plt.text(266.8,0.503,"Median: 266.5")
         # plt.legend('Median')

Out[40]: Text(266.8, 0.503, 'Median: 266.5')
```




```
In [38]: plt.figure(figsize=(12,12))
         kmplot(table2['Time'],table2['Survival_G2'],title="Group 2")
         plt.scatter(table2['Time'],table2['Survival_G2'],color='r')
         plt.plot([0,350],[0.5,0.5])
         plt.plot([278,278],[0.36,1])
         plt.text(278.7,0.503,"Median: 278")
         plt.scatter(278,0.5,color='black')

Out[38]: <matplotlib.collections.PathCollection at 0x7fd75598f208>
```




```
In [29]: plt.figure(figsize=(12,12))
         kmplot(table1['Time'],table1['Survival_G1'])

         kmplot(table2['Time'],table2['Survival_G2'])
         plt.scatter(table1['Time'],table1['Survival_G1'])
         plt.scatter(table2['Time'],table2['Survival_G2'])

Out[29]: <matplotlib.collections.PathCollection at 0x7fd7560ac8d0>
```



The test statistic value is less than the critical value (using chi-square table) for degree of freedom equal to one. Hence, we can say that there is no significant difference between the two groups regarding the survival.

In [ ]: