4741, HW 3, Question 1

Submitted by: Aman Jain (aj644)

Part (a)

Probability that k people (at a polling station) vote for warren = ${}_n^k\mathrm{C} * \mu^k * (1-\mu)^(n-k)$

Given n = 10, $/mu$ = 0.05
p = no. of polling stations,

Lets P = Probability that at least one polling location will have v_j = 0
or P = 1 - (No polling location will have vj = $0)^p$
or P = 1 - (P(At least one per person voted in favour)$)^p$
or P = 1 - (1 - P(k = 0)$)^p$
or P = 1 - (1 - (1-$\mu)^n)^p$
or P = 1 - (1 - $0.95^n)^p$
or P = 1 - (1 - 0.95^10)^p

μ = 0.05
for p = 1,
P = 1 - (1 - 0.95^10)^1
P = 0.5987

for p = 1000,
P = 1 - (1 - 0.95^10)^1000
P = 1

for p = 1,000,000
P = 1 - (1 - 0.95^10)^1000000
P = 1

μ = 0.8
for p = 1, and
P = 1 - (1 - 0.2^10)^1
P = 1.024e-07

for p = 1000,
P = 1 - (1 - 0.2^10)^1000
P = 0.0001023

for p = 1000000
P = 1 - (1 - 0.2^10)^1000000
P = 0.09733

In [9]:
```
p = 1-pow((1 - pow(0.2,10)),1000000)
p
```

Out[9]: 0.09733159268316072

Part (b)

In [38]:
```
from scipy.stats import binom
import matplotlib.pyplot as plt
import numpy as np
```

In [12]:
```
X = [0,1,2,3,4,5,6]
Y = binom.pmf(X,6,0.5)
Y
```

Out[12]:
```
array([0.015625, 0.09375 , 0.234375, 0.3125  , 0.234375, 0.09375 ,
       0.015625])
```

lets consider, P = $P[[|\nu_1 - \mu| > \varepsilon]$
P(ε) = 2 $P[|\nu_1 - 0.5| > \varepsilon]$ *for ε > 0, (as for μ = 0.5, probability function will be constant around the mean)*
P(ε) = 2 $P[|\nu_1 > 0.5 + \varepsilon]$ for ε > 0
P(ε) = 2 *$P[|K >n(0.5 + ε)]$ $for\varepsilon > 0 < br > P(\varepsilon) = 2*$P[|K > 6*(0.5 + ε)]$ for ε > 0*
P(ε) = 2 $P[|K > 3 + 6\varepsilon)]$ for ε > 0

Coming back to P(ε):
P(ε) = 2 $P[|K > 3 + 6\varepsilon)]$ *for ε > 0*
*P(ε = 0) = 2*$P[|K > 3)]$ = 2 *P[K = 4,5,6] = 0.6875*
*P(ε = 0.05) = 2*$P[|K > 3 + 0.3)]$ = 2$P[|K > 3.3)]$ = 2 P[K = 4,5,6] = 0.6875
P(ε = 0.1) = 2$P[|K > 3 + 0.6)]$ = 2$P[|K > 3.6)]$ = 2 *P[K = 4,5,6] = 0.6875*
*P(ε = 0.15) = 2*$P[|K > 3 + 0.9)]$ = 2$P[|K > 3.9)]$ = 2 P[K = 4,5,6] = 0.6875
P(ε = 0.2) = 2$P[|K > 3 + 1.2)]$ = 2$P[|K > 4.2)]$ = 2 *P[K = 5,6] = 0.21875*
*P(ε = 0.25) = 2*$P[|K > 3 + 1.5)]$ = 2$P[|K > 4.5)]$ = 2 P[K = 5,6] = 0.21875
P(ε = 0.3) = 2$P[|K > 3 + 1.8)]$ = 2$P[|K > 4.8)]$ = 2 *P[K = 5,6] = 0.21875*
*P(ε = 0.35) = 2*$P[|K > 3 + 2.1)]$ = 2$P[|K > 5.1)]$ = 2 P[K = 6] = 0.03125
P(ε = 0.4) = 2$P[|K > 3 + 2.4)]$ = 2$P[|K > 5.4)]$ = 2 *P[K = 6] = 0.03125*
*P(ε = 0.45) = 2*$P[|K > 3 + 2.7)]$ = 2$P[|K > 5.7)]$ = 2 P[K = 6] = 0.03125
P(ε = 0.5) = 2$P[|K > 3 + 3)]$ = 2$P[|K > 6)]$ = 2* 0 = 0
P(ε > 0.5) = 0

In [39]:
```
a = 0.015625
2*a
```

Out[39]: 0.03125

In [19]:
```
b = 0.03125
2*b - pow(b,2)
```

Out[19]: 0.0615234375

In [22]:
```
h = np.linspace(0,1,21)
h
```

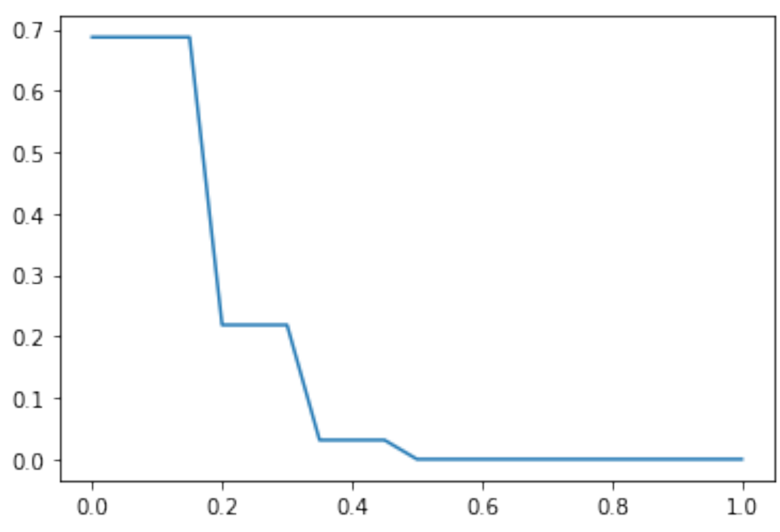
Out[22]:
```
array([0.  , 0.05, 0.1 , 0.15, 0.2 , 0.25, 0.3 , 0.35, 0.4 , 0.45, 0
.5 ,
       0.55, 0.6 , 0.65, 0.7 , 0.75, 0.8 , 0.85, 0.9 , 0.95, 1.  ])
```

In [35]:
```
P = [0.6875, 0.6875 ,0.6875 ,0.6875 ,0.21875, 0.21875, 0.21875, 0.0312
5 ,0.03125 ,0.03125 , 0,0,0,0,0,0,0,0,0,0,0]
np.size(P)
```

Out[35]: 21

In [40]:
```
plt.plot(h,P)
```

Out[40]: [<matplotlib.lines.Line2D at 0x1a1e59a7b8>]



using Hoeffding Inequality, P[|ν – μ| > ε] ≤ 2e–2nε2,

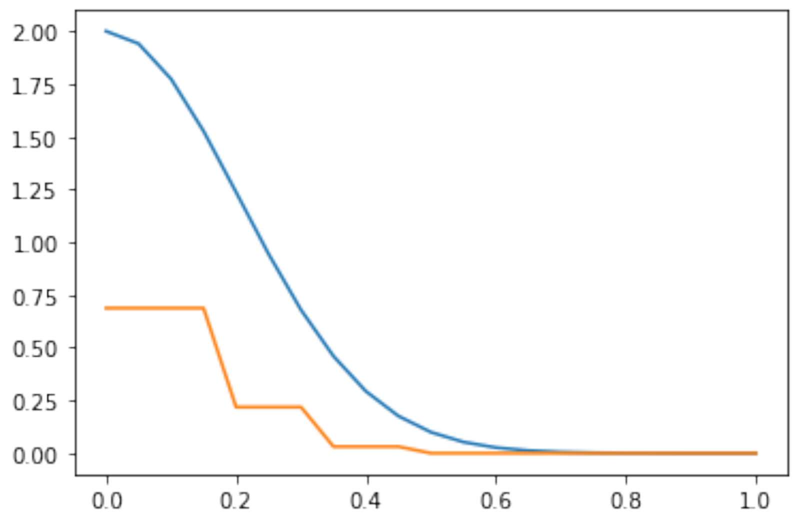
In [41]:
```
P_hoeff = 2*np.exp(-2*6*pow(h,2))
```

In [42]:
```
P_hoeff
```

Out[42]:
```
array([2.00000000e+00, 1.94089107e+00, 1.77384087e+00, 1.52675899e+0
0,
       1.23756678e+00, 9.44733105e-01, 6.79191051e-01, 4.59850970e-0
1,
       2.93213924e-01, 1.76073665e-01, 9.95741367e-02, 5.30323688e-0
2,
       2.65997671e-02, 1.25648403e-02, 5.58957055e-03, 2.34175924e-0
3,
       9.23949798e-04, 3.43318208e-04, 1.20140001e-04, 3.95932139e-0
5,
       1.22884247e-05])
```

In [43]:
```
plt.plot(h,P_hoeff)
plt.plot(h,P)
```

Out[43]:
```
[<matplotlib.lines.Line2D at 0x1a1e50ec88>]
```



Now lets consider two centers:

Required probability, G(ε) = $P[max|\nu_j - \mu| > \varepsilon]$

Required probability, G(ε) = $P((|\nu_1 - \mu| > \varepsilon$ OR $|\nu_2 - \mu|) > \varepsilon)$

Required probability, G(ε) = $P(|\nu_1 - \mu| > \varepsilon)$ + P$(|\nu_2 - \mu| > \varepsilon)$ - $P((|\nu_1 - \mu| > \varepsilon)$ AND $(|\nu_2 - \mu| > \varepsilon)) > 0)$

Required probability, G(ε) = $P(|\nu_1 - \mu| > \varepsilon)$ + P$(|\nu_2 - \mu| > \varepsilon)$ - $P((|\nu_1 - \mu| > \varepsilon) * (|\nu_2 - \mu| > \varepsilon)) > 0)$ (v1 and v2 are independent)

Required probability, G(ε) = P(ε) + P(ε) - P(ε)* P(ε) (referring to P(ε) calculated above)

Required probability, G(ε) = 2P(ε) - $(P(\varepsilon))^2$

Required probability, G(ε) = 2P(ε) - (P(ε))^2

G(ε = 0) = 2 * 0.6875 - (0.6875)^2 = 0.90234375

G(ε = 0.05) = 0.90234375

G(ε = 0.1) = 0.90234375

G(ε = 0.15) = 0.90234375

G(ε = 0.2) = 0.3896484375

G(ε = 0.25) = 0.3896484375

G(ε = 0.3) = 0.3896484375

G(ε = 0.35) = 0.0615234375

G(ε = 0.4) = 0.0615234375

G(ε = 0.45) = 0.0615234375

G(ε = 0.5) = 0

G(ε > 0.5) = 0

In [25]:
```
G = [0.90234375, 0.90234375 ,0.90234375 ,0.90234375 , 0.3896484375, 0.
3896484375, 0.3896484375, 0.0615234375 ,0.0615234375 ,0.0615234375 , 0
,0,0,0,0,0,0,0,0,0,0]
np.size(G)
```

Out[25]: 21

In [44]:
```
plt.plot(h,P_hoeff)
plt.plot(h,G)
```

Out[44]: [<matplotlib.lines.Line2D at 0x1a1e6b5b38>]