STAT 4510J Bayesian Analysis

Fall 2022 – Homework4 Decision Theory

This homework is done by ***Wei Linda***

1 Problem1

1.

$$f(y \mid \theta) \propto \left(\frac{1}{2}+\frac{\theta}{4}\right)^{y_1}\left(\frac{1-\theta}{4}\right)^{y_2+y_3} \cdot\left(\frac{\theta}{4}\right)^{y_4}$$
$$\propto(2+\theta)^{y_1}(1-\theta)^{y_2+y_3} \theta^{y_4}$$

We think $P(\theta)$ is a uniform distribution thus $f(\theta|y) \propto p(\theta)\cdot f(y|\theta) \propto (2+\theta)^{y_1}(1-\theta)^{y_2+y_3}\theta^{y_4}$ which is not a good distribution kernel thus we can separate $y_1 = x_1 + x_2, y_2 = x_3, y_3 = x_4, y_4 = x_5$, the new posterior will be.

$$f(\theta \mid x) \propto \theta^{x_2}(1-\theta)^{x_3+x_4} \theta^{x_5}$$
$$\propto \theta^{x_2+x_5}(1-\theta)^{x_3+x_4}$$

which follows a Beta distribution Beta (x_2+x_5+1, x_3+x_4+1) the log-likely hood

$$l(\theta|x) = (x_2+x_5)\ln\theta + (x_3+x_4)\ln(1-\theta)$$

the E - step

$$Q\left(\theta \mid \theta^t\right) = E\left[(\theta \mid x) \mid y, \theta^t\right]$$
$$= E\left[\left[(x_2+x_5)\ln\theta+(x_3+x_4)\ln(1-\theta)\right] \mid y_1, \theta^t\right]$$
$$= (y_2+y_3)\ln(1-\theta)+\left\{y_4+E\left[x_2 \mid y_1, \theta^t\right]\right\}\ln\theta$$
$$\frac{\partial Q}{\partial\theta} = \frac{y_2+y_3}{1-\theta}\times(-1)+\frac{1}{\theta}\{y_4+E\left[x_2 \mid y_1, \theta^t\right]\} = 0$$
$$\frac{y_2+y_3}{\theta-1}+\frac{1}{\theta}\left(y_4+E\left[x_2 \mid x_1, \theta^t\right]\right) = 0,$$
$$\theta\left(y_2+y_3\right)+(\theta-1)\left[y_4+E\left[x_2 \mid y_1, \theta^t\right]\right] = 0$$
$$\theta\left(y_2+y_3+y_4+E\left[x_2 \mid y_1, \theta^t\right]\right) = y_4+E\left[x_2 \mid y_\theta t\right)$$
$$\theta^{t+1} = \frac{y_4+E\left[x_2 \mid y_1, \theta^t\right]}{y_2+y_3+y_4+E\left[x_2 \mid y_1, \theta^t\right]}$$

in which the $E\left[x_2 \mid x_1, \theta^t\right]$ will be calculated as:

$$
\begin{aligned}
& E\left[x_2 \mid y_1, \theta^t\right] \\
& = E\left[x_2 \mid x_1 + x_2, \theta^t\right] \\
& \therefore E\left[x_2 \mid x_1, \theta^t\right] = y_1 \cdot \frac{\frac{\theta^t}{4}}{\frac{\theta^t}{4} + \frac{1}{2}}
\end{aligned}
$$

Here are the python code to implement the E-M algorithm

```
#Your turn
#The samples are comming from 3 gaussian distributions,
#please find the source of each cluster & the corresponding parameter
#mu & sigma
import pandas as pd
import matplotlib.pyplot as plt
import os
import sys
import glob
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from scipy.stats import multivariate_normal as mvn

%matplotlib inline
plt.style.use('ggplot')
np.random.seed(1234)

np.set_printoptions(formatter={'all':lambda x: '%.3f' % x})

from IPython.display import Image
from numpy.core.umath_tests import matrix_multiply as mm
from scipy.optimize import minimize
from scipy.stats import bernoulli, binom
```

```
C:\Users\aa219040\AppData\Local\Temp\ipykernel_2912\862796165.py:22: DeprecationWarning: numpy.core.umath_tests is an internal NumPy module
and should not be imported. It will be removed in a future NumPy release.
  from numpy.core.umath_tests import matrix_multiply as mm
```

```
y1=125
y2=18
y3=20
y4=34
i=0
theta =0.5
theta_next=0.6
threshold=0.0001
```

```
while(abs(theta-theta_next)>threshold):
    i=i+1
    print("Iteration ",i)
    theta = theta_next
    x2t = 125*(theta/4)/(0.5+theta/4)
    theta_next = (y4+x2t)/(y2+y3+y4+x2t)
    print("Estimate theta is",theta_next)
theta_next
```

```
0.6268129668525383
```

图 1: Implementation Of EM algorithm

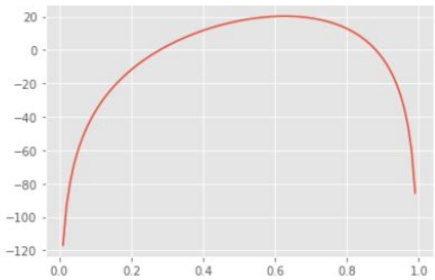
And the figure Below is the plot of the log likelihood function

From the plot you can see that the maximum of the function is near 0.63, which is the same as our calculation.

In [18]:
```
x = np.arange(0,1.0,0.01)
def log_likelyhood(x):
    return 125*np.log(2+x)+np.log(1-x)*38+np.log(x/4)*34
y=log_likelyhood(x)
plt.plot(x,y)
```

```
C:\Users\aa219040\AppData\Local\Temp\ipykernel_2912\1441262157.py:3: RuntimeWarning: divide by zero encountered in log
  return 125*np.log(2+x)+np.log(1-x)*38+np.log(x/4)*34
```

Out[18]: [<matplotlib.lines.Line2D at 0x26703ef2ee0>]



From the plot you can see the maximum of this function is near to 0.63, which is the same as our calculation.

图 2: Plot of log likelyhood

2 Problem 2

Please refer to the jupyter notebook.