CSDS 391: W4

John Mays, `jkm100`

Due 11/02/21, Professor Lewicki

Q1

a)

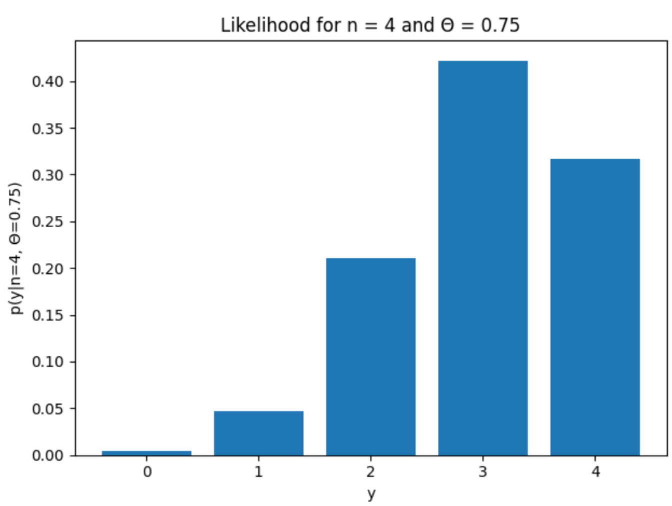
Normalizing constant:

$$p(y|n) = \int_0^1 p(y|\Theta, n)p(\Theta, n)d\Theta = \int_0^1 p(y|\Theta, n)d\Theta = \frac{1}{1+n}$$

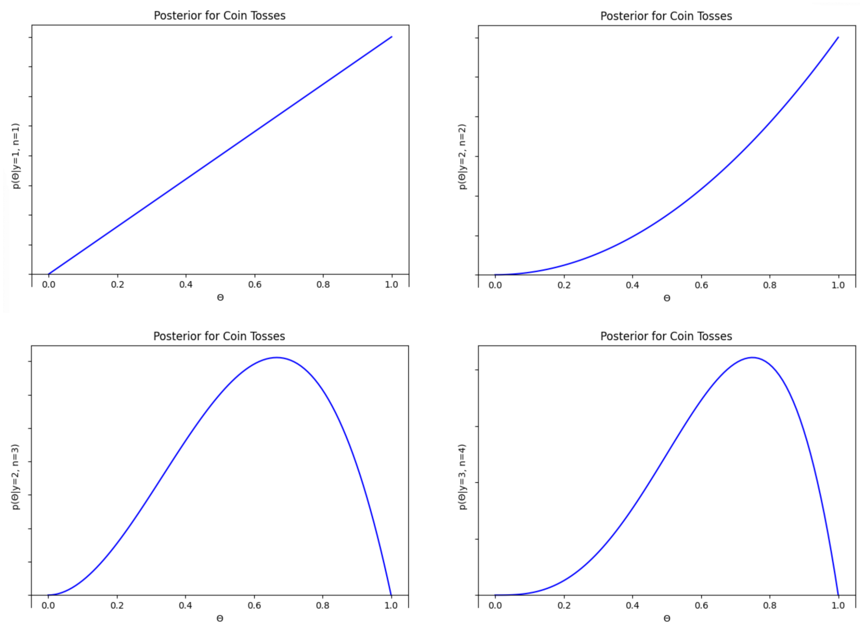
Posterior:

$$p(\Theta|y, n) = \frac{p(y|\Theta, n)p(\Theta|n)}{p(y|n)} = p(y|\Theta, n)p(\Theta|n)(1+n)$$

b)



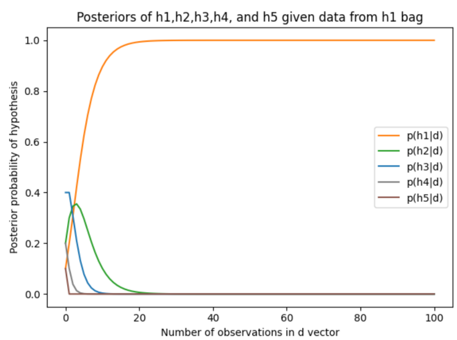
c)


Posterior for Coin Tosses
p(Θ|y=1, n=1)
0.0
0.2
0.4
0.6
0.8
1.0
Θ
Posterior for Coin Tosses
p(Θ|y=2, n=2)
0.0
0.2
0.4
0.6
0.8
1.0
Θ
Posterior for Coin Tosses
p(Θ|y=2, n=3)
0.0
0.2
0.4
0.6
0.8
1.0
Θ
Posterior for Coin Tosses
p(Θ|y=3, n=4)
0.0
0.2
0.4
0.6
0.8
1.0
Θ

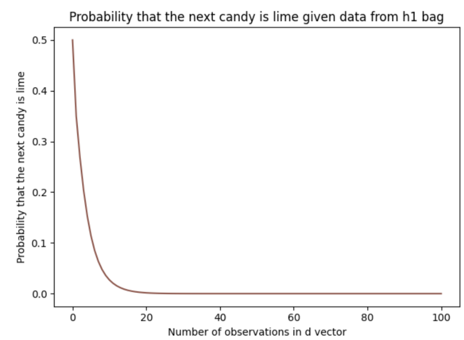
Q2

a)

h1:

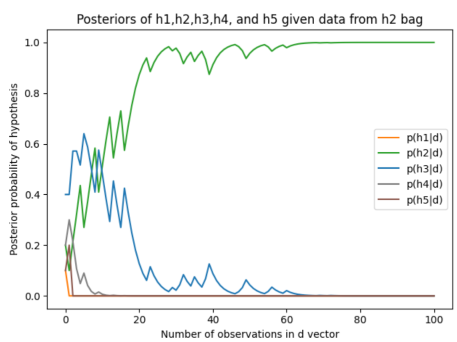


Posteriors of h1,h2,h3,h4, and h5 given data from h1 bag
Posterior probability of hypothesis
Number of observations in d vector
p(h1|d)
p(h2|d)
p(h3|d)
p(h4|d)
p(h5|d)

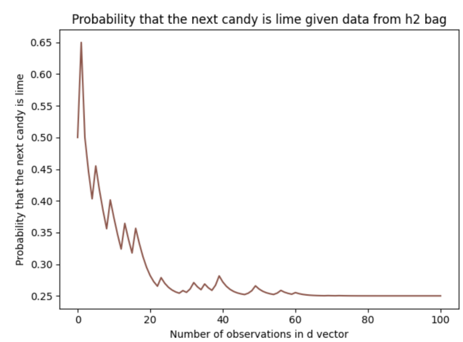


Probability that the next candy is lime given data from h1 bag
Probability that the next candy is lime
Number of observations in d vector

h2:

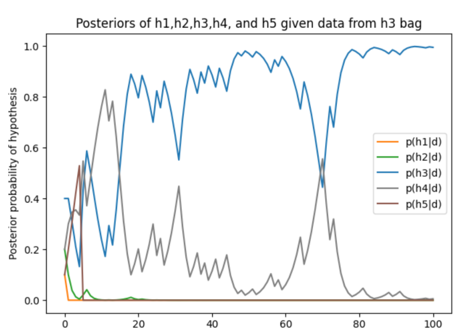


Posteriors of h1,h2,h3,h4, and h5 given data from h2 bag
Posterior probability of hypothesis
Number of observations in d vector
p(h1|d)
p(h2|d)
p(h3|d)
p(h4|d)
p(h5|d)

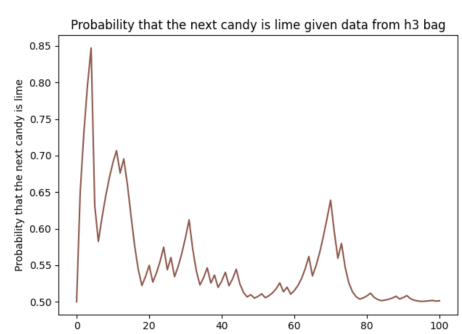


Probability that the next candy is lime given data from h2 bag
Probability that the next candy is lime
Number of observations in d vector

h3:

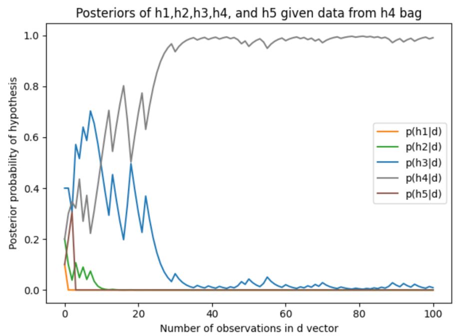


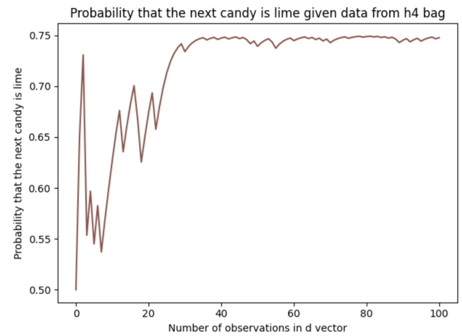
Posteriors of h1,h2,h3,h4, and h5 given data from h3 bag
Posterior probability of hypothesis
Number of observations in d vector
p(h1|d)
p(h2|d)
p(h3|d)
p(h4|d)
p(h5|d)



Probability that the next candy is lime given data from h3 bag
Probability that the next candy is lime
Number of observations in d vector

h4:





b)

Given a data vector $\mathbf{d}$ with N data points, and understanding that $\mathbf{d}_i$ is the data vector of length i with the first i observations from $\mathbf{d}$, $0 \leq i \leq N$, then

$$\min(i), \text{ s.t. } i \text{ satisfies } \max(\begin{bmatrix} P(h_1|\mathbf{d}_i) \\ P(h_2|\mathbf{d}_i) \\ P(h_3|\mathbf{d}_i) \\ P(h_4|\mathbf{d}_i) \\ P(h_5|\mathbf{d}_i) \end{bmatrix}) > 0.9$$

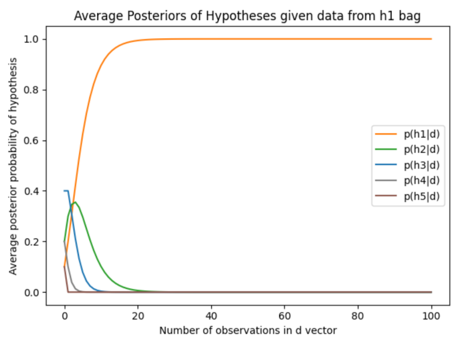
will return the minimum value of i for which any hypothesis, given $\mathbf{d}_i$, will return a probability of $> 90\%$. Then, i is the number of observations you have to make until you can be more than 90% sure of which bag you have.

For example, say you generated data vector $\mathbf{d}$ from a bag of candies satisfying one hypothesis. You would consider the set of values of i from 0 to N that satisfy the inequality. The inequality considers the most probable hypothesis, given data vector $\mathbf{d}_i$, and sees if it is greater than 0.9. The function returns the minimum value of i that satisfies the inequality, otherwise stated as the number of the first observation that leads to a posterior probability of greater than 90%.

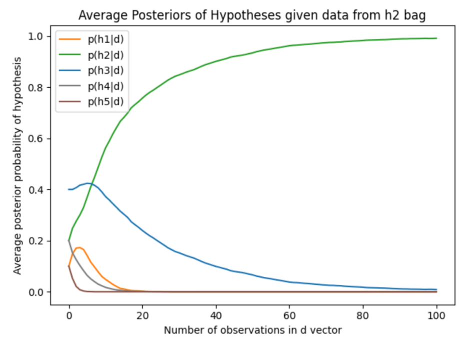
c)

For this question, I did essentially the same exact steps as part **2a**, only each data point is an average of 2500 runs of **2a**.

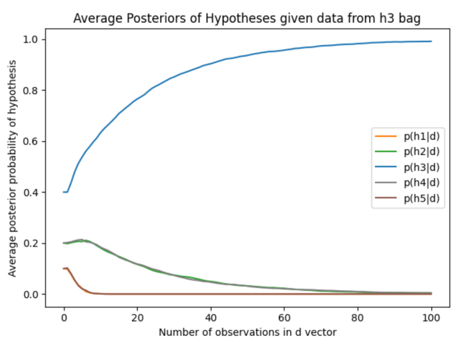
Bag satisfying **h1**:



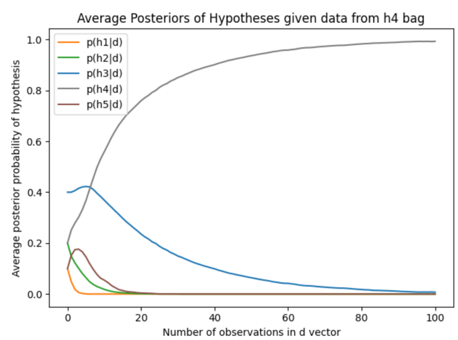
Bag satisfying **h2**:



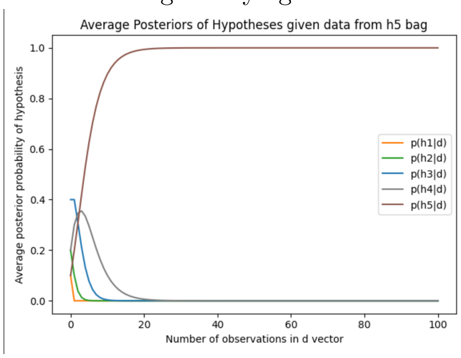
Bag satisfying **h3**:



Bag satisfying **h4**:



Bag satisfying **h5**:



Q3

The objective function, distortion, is given as

$$D = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} ||\mathbf{x}_n - \boldsymbol{\mu}_k||^2$$

The update rule is supposed to minimize this function, therefore we can take the gradient and set it equal to zero:

$$\frac{\partial D}{\partial \boldsymbol{\mu}_k} = 2 \sum_{n=1}^{N} r_{n,k} (\mathbf{x}_n - \boldsymbol{\mu}_k)$$

$$= 2 \sum_{n=1}^{N} r_{n,k} \mathbf{x}_n - 2 \sum_{n=1}^{N} r_{n,k} \boldsymbol{\mu}_k = 0 \implies$$

$$\sum_{n=1}^{N} r_{n,k} \boldsymbol{\mu}_k = \sum_{n=1}^{N} r_{n,k} \mathbf{x}_n$$

Since the $\boldsymbol{\mu}_k$ vector is the same across the entire sum on the left side, we can take it out as a constant:

$$\boldsymbol{\mu}_k \sum_{n=1}^{N} r_{n,k} = \sum_{n=1}^{N} r_{n,k} \mathbf{x}_n \implies$$

$$\boldsymbol{\mu}_k = \frac{\sum\limits_{n=1}^{N} r_{n,k} \mathbf{x}_n}{\sum\limits_{n=1}^{N} r_{n,k}}$$

In other words, the gradient with respect to $\boldsymbol{\mu}_k$ is equal to 0 when $\boldsymbol{\mu}_k$ is equal to the average value of the data vectors $\mathbf{x}_n$ belonging to class k.
The same can be said for the individual values of $\boldsymbol{\mu}_k$, $\mu_{k,i}$:

$$\mu_{k,i} = \frac{\sum\limits_{n=1}^{N} r_{n,k} x_{n,i}}{\sum\limits_{n=1}^{N} r_{n,k}}$$

Given these two equations, we can formulate our update rule as:

- update mean $\boldsymbol{\mu}_k$ to satisfy equations (listed below)
- repeat until convergence

scalar form equation:

$$\mu_{k,i} = \frac{\sum\limits_{n=1}^{N} r_{n,k} x_{n,i}}{\sum\limits_{n=1}^{N} r_{n,k}}$$

vector form equation:

$$\boldsymbol{\mu}_k = \frac{\sum\limits_{n=1}^{N} r_{n,k} \mathbf{x}_n}{\sum\limits_{n=1}^{N} r_{n,k}}$$