Computer Science M146, Homework 5

Michael Wu
UID: 404751542

March 15th, 2018

Problem 1

a) We lose the ordering of the documents, as a document $D_1 = \{a, b, c\}$ is treated as equivalent to the document $D_2 = \{a, c, b\}$. We only care about the number of words in the document, not the order they appear in.

b)

$$\begin{aligned}
\log \mathrm{P}(D_i, y_i) &= \log \left((\mathrm{P}(D_i | y_i = 1) \, \mathrm{P}(y_i = 1))^{y_i} \, (\mathrm{P}(D_i | y_i = 0) \, \mathrm{P}(y_i = 0))^{1-y_i} \right) \\
&= \log \left(\mathrm{P}(D_i | y_i = 1)^{y_i} \theta^{y_i} \, \mathrm{P}(D_i | y_i = 0)^{1-y_i} (1-\theta)^{1-y_i} \right) \\
&= y_i (\log \theta + \log \mathrm{P}(D_i | y_i = 1)) \\
&\qquad + (1 - y_i)(\log(1-\theta) + \log \mathrm{P}(D_i | y_i = 0)) \\
&= y_i \left(\log \theta + \log \frac{n!}{a_i! b_i! c_i!} \alpha_1^{a_i} \beta_1^{b_i} \gamma_1^{c_i} \right) \\
&\qquad + (1 - y_i) \left(\log(1-\theta) + \log \frac{n!}{a_i! b_i! c_i!} \alpha_0^{a_i} \beta_0^{b_i} \gamma_0^{c_i} \right) \\
&= y_i \left(\log \theta + \log \frac{n!}{a_i! b_i! c_i!} + a_i \log \alpha_1 + b_i \log \beta_1 + c_i \log \gamma_1 \right) \\
&\qquad + (1 - y_i) \left(\log(1-\theta) + \log \frac{n!}{a_i! b_i! c_i!} + a_i \log \alpha_0 + b_i \log \beta_0 \right. \\
&\qquad\quad \left. + \, c_i \log \gamma_0 \right)
\end{aligned}$$

$$\begin{aligned}\log \mathrm{P}(D_i, y_i) = \log \frac{n!}{a_i! b_i! c_i!} &+ y_i(\log \theta + a_i \log \alpha_1 + b_i \log \beta_1 + c_i \log \gamma_1) \\ &+ (1 - y_i)(\log(1-\theta) + a_i \log \alpha_0 + b_i \log \beta_0 + c_i \log \gamma_0)\end{aligned}$$

c) We can find the maximum likelihood estimate for each of our parameters by finding

$$\max_{\alpha_1,\beta_1,\gamma_1,\alpha_0,\beta_0,\gamma_0} \sum_{i=1}^{m} \log \mathrm{P}(D_i, y_i)$$

Our maximum likelihood estimate occurs when

$$\frac{\partial}{\partial \alpha_1} \sum_{i=1}^{m} \log \mathrm{P}(D_i, y_i) = 0$$

Because we are given that $\alpha_1 + \beta_1 + \gamma_1 = 1$, we know that β_1 is a function of α_1 and γ_1. So we can let $\beta_1 = 1 - \alpha_1 - \gamma_1$. Taking the partial derivative yields

$$\begin{aligned}\frac{\partial}{\partial \alpha_1} \sum_{i=1}^{m} \log \mathrm{P}(D_i, y_i) &= \frac{\partial}{\partial \alpha_1} \sum_{i=1}^{m} y_i(a_i \log \alpha_1 + b_i \log \beta_1) \\ &= \frac{\partial}{\partial \alpha_1} \sum_{i=1}^{m} y_i(a_i \log \alpha_1 + b_i \log(1 - \alpha_1 - \gamma_1)) \\ &= \sum_{i=1}^{m} y_i \left(\frac{a_i}{\alpha_1} - \frac{b_i}{1 - \alpha_1 - \gamma_1} \right) \\ &= \sum_{i=1}^{m} y_i(a_i(1 - \alpha_1 - \gamma_1) - b_i \alpha_1) \\ &= (1 - \gamma_1) \sum_{i=1}^{m} y_i a_i - \alpha_1 \sum_{i=1}^{m} y_i(a_i + b_i)\end{aligned}$$

Then we can solve for zero which gives

$$\alpha_1 = \frac{\sum_{i=1}^{m} y_i a_i}{\sum_{i=1}^{m} y_i(a_i + b_i)}(1 - \gamma_1) = C_1(1 - \gamma_1)$$

Using the same process with the partial derivative for γ_1 yields

$$\begin{aligned}\frac{\partial}{\partial \gamma_1}\sum_{i=1}^{m}\log \mathrm{P}(D_i, y_i) &= \frac{\partial}{\partial \gamma_1}\sum_{i=1}^{m} y_i(c_i \log \gamma_1 + b_i \log \beta_1)\\ &= \sum_{i=1}^{m} y_i\left(\frac{c_i}{\gamma_1} - \frac{b_i}{1-\alpha_1-\gamma_1}\right)\\ &= \sum_{i=1}^{m} y_i(c_i(1-\alpha_1-\gamma_1) - b_i\gamma_1)\\ &= (1-\alpha_1)\sum_{i=1}^{m} y_i c_i - \gamma_1 \sum_{i=1}^{m} y_i(b_i + c_i)\end{aligned}$$

Setting this equal to zero gives us

$$\gamma_1 = \frac{\sum_{i=1}^{m} y_i c_i}{\sum_{i=1}^{m} y_i(b_i + c_i)}(1-\alpha_1) = C_2(1-\alpha_1)$$

and thus

$$\begin{aligned}\alpha_1 &= \frac{C_1(1-C_2)}{1-C_1C_2}\\ \beta_1 &= \frac{(1-C_1)(1-C_2)}{1-C_1C_2}\\ \gamma_1 &= \frac{C_2(1-C_1)}{1-C_1C_2}\end{aligned}$$

where

$$\begin{aligned}C_1 &= \frac{\sum_{i=1}^{m} y_i a_i}{\sum_{i=1}^{m} y_i(a_i + b_i)}\\ C_2 &= \frac{\sum_{i=1}^{m} y_i c_i}{\sum_{i=1}^{m} y_i(b_i + c_i)}\end{aligned}$$

Similarly for α_0, β_0, and γ_0, taking partial derivatives and setting them to zero yields

$$\begin{aligned}\alpha_0 &= \frac{C_3(1-C_4)}{1-C_3C_4}\\ \beta_0 &= \frac{(1-C_3)(1-C_4)}{1-C_3C_4}\\ \gamma_0 &= \frac{C_4(1-C_3)}{1-C_3C_4}\end{aligned}$$

where

$$C_3 = \frac{\sum_{i=1}^{m}(1-y_i)a_i}{\sum_{i=1}^{m}(1-y_i)(a_i+b_i)}$$
$$C_4 = \frac{\sum_{i=1}^{m}(1-y_i)c_i}{\sum_{i=1}^{m}(1-y_i)(b_i+c_i)}$$

because of the symmetry of our log likelihood function $\log \mathrm{P}(D_i, y_i)$. After some algebraic simplification these values become

$$\alpha_1 = \frac{\sum_{i=1}^{m} y_i a_i}{\sum_{i=1}^{m} y_i n}$$
$$\beta_1 = \frac{\sum_{i=1}^{m} y_i b_i}{\sum_{i=1}^{m} y_i n}$$
$$\gamma_1 = \frac{\sum_{i=1}^{m} y_i c_i}{\sum_{i=1}^{m} y_i n}$$
$$\alpha_0 = \frac{\sum_{i=1}^{m}(1-y_i)a_i}{\sum_{i=1}^{m}(1-y_i)n}$$
$$\beta_0 = \frac{\sum_{i=1}^{m}(1-y_i)b_i}{\sum_{i=1}^{m}(1-y_i)n}$$
$$\gamma_0 = \frac{\sum_{i=1}^{m}(1-y_i)c_i}{\sum_{i=1}^{m}(1-y_i)n}$$

such that α_1 is the total proportion of a words in the set of D_i that have label $y_i = 1$, β_1 is the total proportion of b words in the set of D_i that have label $y_i = 1$, γ_1 is the total proportion of c words in the set of D_i that have label $y_i = 1$, α_0 is the total proportion of a words in the set of D_i that have label $y_i = 0$, β_0 is the total proportion of b words in the set of D_i that have label $y_i = 0$, and γ_0 is the total proportion of c words in the set of D_i that have label $y_i = 0$.

Problem 2

a) The two unspecified state transitions are

$$q_{21} = \mathrm{P}(q_{t+1} = 2 | q_t = 1) = 0$$
$$q_{22} = \mathrm{P}(q_{t+1} = 2 | q_t = 2) = 0$$

The two unspecified output probabilities are

$$\begin{aligned}
e_1(B) &= \mathrm{P}(O_t = B|q_t = 1) = 0.01 \\
e_2(A) &= \mathrm{P}(O_t = A|q_t = 2) = 0.49
\end{aligned}$$

b) We have

$$\mathrm{P}(A) = \pi_1 e_1(A) + \pi_2 e_2(A) = 0.49 \times 0.99 + 0.51 \times 0.49 = 0.735$$

We also have

$$\mathrm{P}(B) = \pi_1 e_1(B) + \pi_2 e_2(B) = 0.49 \times 0.01 + 0.51 \times 0.51 = 0.265$$

Thus A will be the most frequent output symbol to appear in the first position of sequences generated from this HMM.

c) Consider any output sequence that begins after q_1. Because the state transition probabilities are 1 for q_{11} and q_{12}, any output sequence that begins after $t = 1$ must begin with state $q_t = 1$. After reaching state 1, the state must remain 1. Then the three letter sequences have the following probabilities

$$\begin{aligned}
\mathrm{P}(AAA) &= e_1(A)^3 e_1(B)^0 = 0.970299 \\
\mathrm{P}(AAB) &= e_1(A)^2 e_1(B)^1 = 0.009801 \\
\mathrm{P}(ABA) &= e_1(A)^2 e_1(B)^1 = 0.009801 \\
\mathrm{P}(ABB) &= e_1(A)^1 e_1(B)^2 = 0.000099 \\
\mathrm{P}(BAA) &= e_1(A)^2 e_1(B)^1 = 0.009801 \\
\mathrm{P}(BAB) &= e_1(A)^1 e_1(B)^2 = 0.000099 \\
\mathrm{P}(BBA) &= e_1(A)^1 e_1(B)^2 = 0.000099 \\
\mathrm{P}(BBB) &= e_1(A)^0 e_1(B)^3 = 0.000001
\end{aligned}$$

Because the sequence that is generated may be of an arbitrary length, there can be an arbitrarily large number of sequences that begin after the first state. So the effect of the initial state probabilities goes to zero, and we can conclude that the most probable sequence of three output symbols that can be generated from this HMM model is AAA.

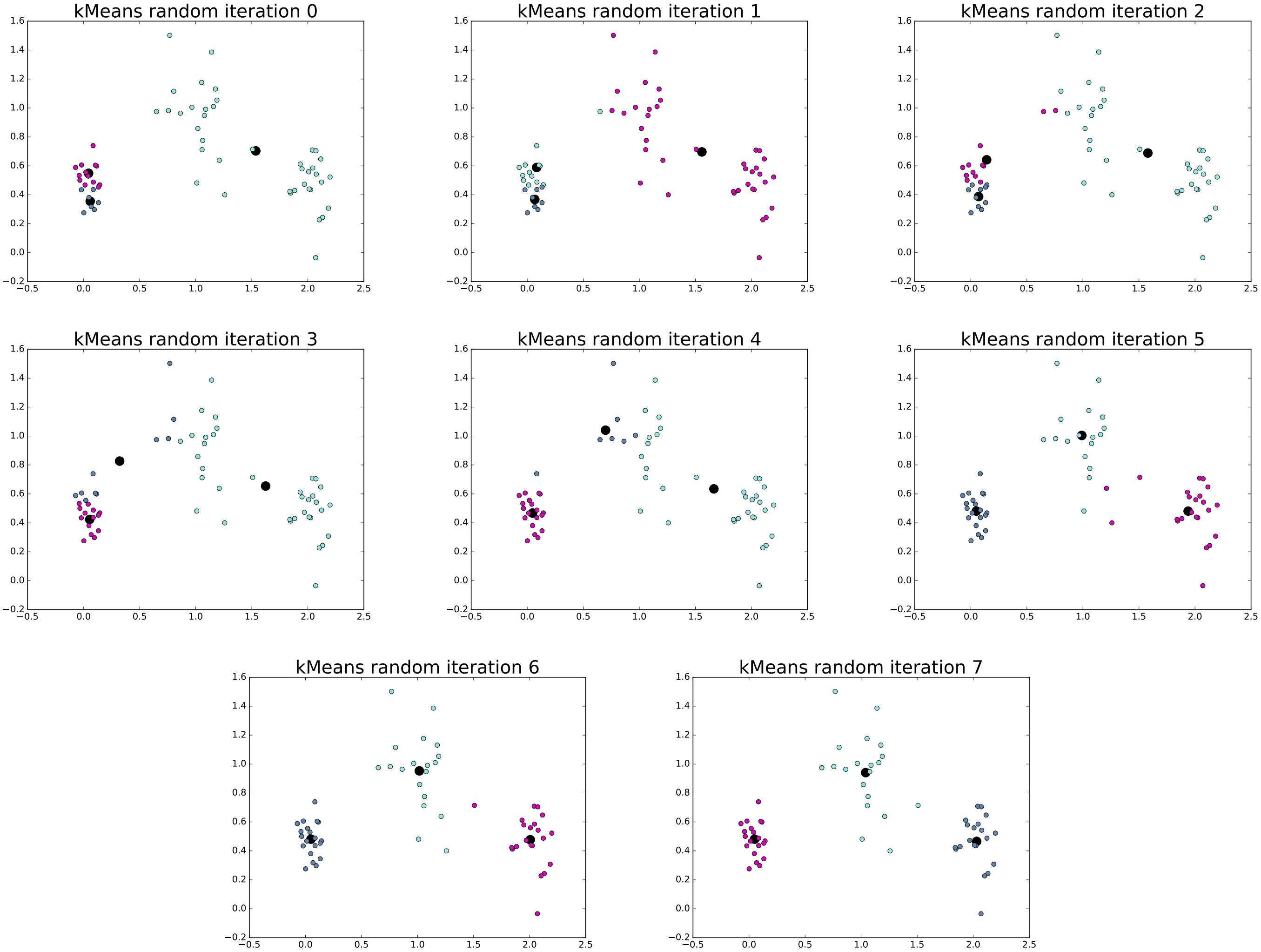
Problem 3

a) The minimum value of $J(c, \mu, k)$ is zero. This occurs when there are $k = n$ clusters such that each cluster is assigned to a single data point. So $c^{(i)} = i$, $\mu_i = x^i$, and $k = n$. This is a bad idea because it gives no information about the data, as it gives each point a unique label.

b) This implementation was pretty simple. In `Cluster` I simply took the average of all the features to find the centroid, and used the distance function to find the medoids. In `ClusterSet` I simply returned a list of all the centroids and medoids to implement the missing functions in the class.

c) For `random_init`, I used `np.random.choice` to select random starting points from the list of given points. I made sure to copy points and pop the selected points in order to avoid duplicates. In `kMeans` I implemented the initial point generation based on either random or cheat, then I found the initial clusters based on these initial points. I implemented plotting for each iteration of the algorithm, and made it calculate updated clusters by calculating the centroids and assigning points to the closest centroid. I terminate the algorithm when the centroids do not change after an iteration.

d) The results of running the `kMeans` algorithm with a random initialization are shown in the following figure.


kMeans random iteration 0

kMeans random iteration 1

kMeans random iteration 2

kMeans random iteration 3

kMeans random iteration 4

kMeans random iteration 5

kMeans random iteration 6

kMeans random iteration 7

Figure 1: All iterations of kMeans with random initialization.

e) The results of running the `kMedoids` algorithm with a random initialization are shown in the following figure.

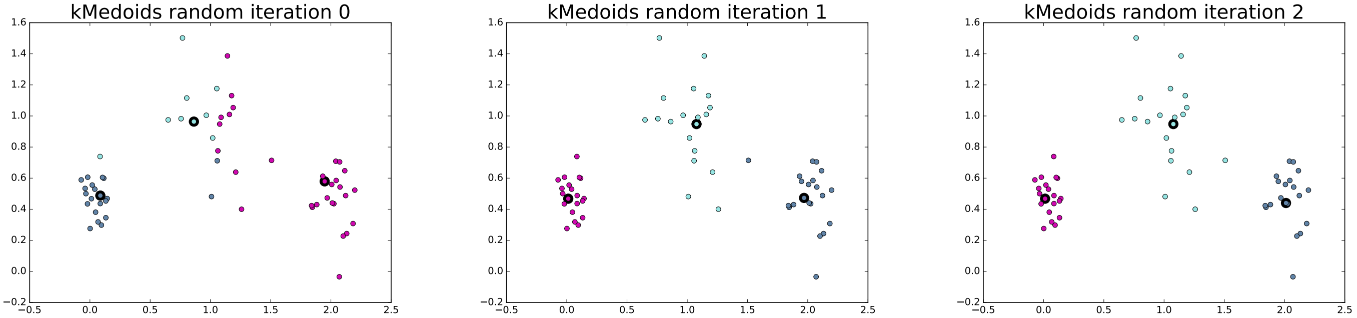


Figure 2: All iterations of kMedoids with random initialization.

f) The results of running the `kMeans` and `kMedoids` algorithm with a cheating initialization are shown in the following figure. Note that both algorithms terminated after one iteration as the cheating allowed the algorithms to begin with the optimal clusters.

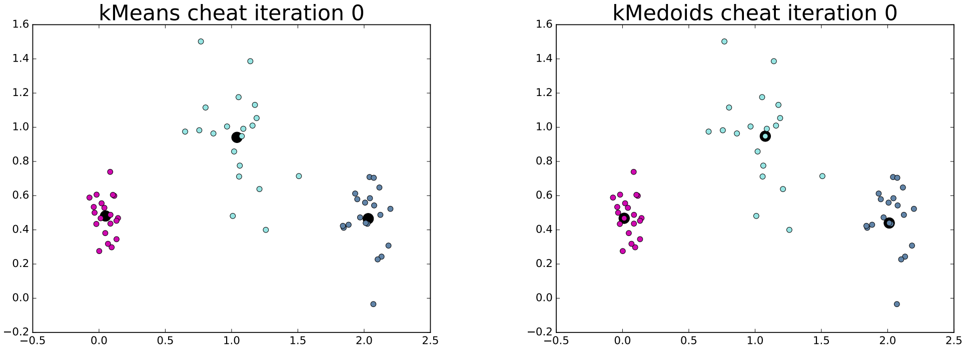


Figure 3: kMeans and kMedoids with cheating initialization.