CMSC 25400 Homework 2

Ruolin Zheng

January 21, 2019

1 Problem 1

1.1 (a)

WTS $J_{IC} = 2J_{avg^2}$. To simplify our calculation, we fix j and consider $J_{IC_j} = 2J_{avg_j^2}$.

$$J_{IC_j} = \frac{1}{|C_j|}\sum_{\mathbf{x}\in C_j}\sum_{\mathbf{x}'\in C_j} d(\mathbf{x},\mathbf{x}')^2 \qquad J_{IC} = \sum_{j=1}^{k}\frac{1}{|C_j|}\sum_{\mathbf{x}\in C_j}\sum_{\mathbf{x}'\in C_j} d(\mathbf{x},\mathbf{x}')^2$$

$$J_{avg_j^2} = \sum_{x\in C_j} d(\mathbf{x},\mathbf{x}')^2 \qquad J_{avg^2} = \sum_{j=1}^{k}\sum_{x\in C_j} d(\mathbf{x},\mathbf{x}')^2$$

First consider the LHS,

$$\begin{aligned}
J_{IC_j} &= \frac{1}{|C_j|}\sum_{\mathbf{x}\in C_j}\sum_{\mathbf{x}'\in C_j} d(\mathbf{x},\mathbf{x}')^2\\
&= \frac{1}{|C_j|}\sum_{i=1}^{|C_j|}\sum_{k=1}^{|C_j|} d(\mathbf{x}_i,\mathbf{x}_k)^2\\
&= \frac{1}{|C_j|}\sum_{i=1}^{|C_j|}\sum_{k=1}^{|C_j|} ||\mathbf{x}_i-\mathbf{x}_k||^2\\
&= \frac{1}{|C_j|}\sum_{i=1}^{|C_j|}\sum_{k=1}^{|C_j|} ||\mathbf{x}_i||^2+||\mathbf{x}_k||^2-||\mathbf{x}_i||||\mathbf{x}_k||\\
&= \frac{1}{|C_j|}\sum_{i=1}^{|C_j|}|C_j|||\mathbf{x}_i||^2+\frac{1}{|C_i|}\sum_{k=1}^{|C_j|}|C_j|||\mathbf{x}_k||^2-\frac{2}{|C_j|}\sum_{i=1}^{|C_j|}\sum_{k=1}^{|C_j|}||\mathbf{x}_i||||\mathbf{x}_j||\\
&= 2\sum_{i=1}^{|C_j|}||\mathbf{x}_i||^2-\frac{2}{|C_j|}\sum_{i=1}^{|C_j|}\sum_{k=1}^{|C_j|}||\mathbf{x}_i||\cdot||\mathbf{x}_k||
\end{aligned}$$

Now consider the RHS,

$$\begin{aligned}
J_{avg_j^2} &= \sum_{\mathbf{x}\in C_j} d(\mathbf{x}, \mathbf{m}_j)^2 \\
&= \sum_{i=1}^{|C_j|} d(\mathbf{x}, \mathbf{m}_j)^2 \\
&= \sum_{i=1}^{|C_j|} d(\mathbf{x}, \frac{1}{|C_j|} \sum_{i=1}^{|C_j|} \mathbf{x}_i)^2 \\
&= \sum_{i=1}^{|C_j|} \left|\left|\mathbf{x} - \frac{1}{|C_j|} \sum_{i=1}^{|C_j|} \mathbf{x}_i\right|\right|^2 \\
&= \sum_{i=1}^{|C_j|} ||\mathbf{x}_i||^2 - \frac{2}{|C_j|}||\mathbf{x}_i|| \cdot \sum_{i=1}^{|C_j|} ||\mathbf{x}_i|| + \frac{1}{|C_j|} \sum_{i=1}^{|C_j|} ||\mathbf{x}_i||^2 \\
&= \sum_{i=1}^{|C_j|} ||\mathbf{x}_i||^2 - \frac{2}{|C_j|} \sum_{i=1}^{|C_j|} \sum_{k=1}^{|C_j|} ||\mathbf{x}_i|| \cdot ||\mathbf{x}_k|| + \frac{1}{|C_j|} \sum_{i=1}^{|C_j|} \sum_{k=1}^{|C_j|} ||\mathbf{x}_i|| \cdot ||\mathbf{x}_k|| \\
&= \sum_{i=1}^{|C_j|} ||\mathbf{x}_i||^2 - \frac{1}{|C_j|} \sum_{i=1}^{|C_j|} \sum_{k=1}^{|C_j|} ||\mathbf{x}_i|| \cdot ||\mathbf{x}_k|| \\
&= \frac{1}{2} J_{IC_j}
\end{aligned}$$

Therefore,

$$J_{IC} = \sum_{j=1}^{k} J_{IC_j} = \sum_{j=1}^{k} 2J_{avg_j^2} = 2\sum_{j=1}^{k} J_{avg_j^2} = 2J_{avg^2}$$

1.2 (b)

1.2.1 (i)

Let γ_i' denote the cluster assignment in the previous iteration and γ_i the one in the current iteration. The update rule $\gamma_i \leftarrow \text{argmin}_{j\in 1,\ldots,k}\, d(\mathbf{x}_i, \mathbf{m}_j)$ requires that $d(\mathbf{x}_i, \mathbf{m}_{\gamma_i}) \leq d(\mathbf{x}_i, \mathbf{m}_{\gamma_i'})$. Therefore,

$$\begin{aligned}
J_{avg^2}(\gamma_1, \ldots, \gamma_n, \mathbf{m}_1, \ldots, \mathbf{m}_k) &= \sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{m}_{\gamma_i})^2 \\
&\leq \sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{m}_{\gamma_i'})^2 = J_{avg^2}(\gamma_1', \ldots, \gamma_n', \mathbf{m}_1, \ldots, \mathbf{m}_k)
\end{aligned}$$

1.2.2 (ii)

Find $\mathbf{m}'_j$ which minimizes J_{avg^2} by minimizing $\sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{m}'_j)^2$.

$$\begin{aligned}
\sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{m}'_j)^2 &= \sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{m}'_j||^2 \\
&= \sum_{i=1}^{n} ||\mathbf{x}_i||^2 - 2||\mathbf{m}'_j|| \cdot \sum_{i=1}^{n} ||\mathbf{x}_i|| + \sum_{i=1}^{n} ||\mathbf{m}'_j||^2 \\
&= \sum_{i=1}^{n} ||\mathbf{x}_i||^2 - 2||\mathbf{m}'_j|| \cdot \sum_{i=1}^{n} ||\mathbf{x}_i|| + |C_j|||\mathbf{m}'_j||^2 \\
\frac{dJ_{avg^2}}{dm'_j} &= -2\sum_{i=1}^{n} ||\mathbf{x}_i|| + 2|C_j|||\mathbf{m}'_j|| = 0 \\
||\mathbf{m}'_j|| &= \frac{1}{|C_j|} \sum_{i=1}^{n} ||\mathbf{x}_i|| \\
\mathbf{m}'_j &= \frac{1}{|C_j|} \sum_{i=1}^{n} \mathbf{x}_i
\end{aligned}$$

Therefore, $\mathbf{m}'_j = \mathbf{m}_j = \frac{1}{|C_j|} \sum_{i=1}^{n} \mathbf{x}_i$ minimizes J_{avg^2}.

1.3 (c)

The proof that J decreases monotonically iteration by iteration follows from (b), where both steps of update minimize J. The k-means algorithm halts when J no longer decreases.

1.4 (d)

The upper bound is k^n, where k is the number of clusters and n the number of data points. This follows from the fact that there are k^n ways to partition n data points into k clusters. Each $\mathbf{x}$ can be assigned to one of the k clusters, and the n assignments are independent.

2 Problem 2

2.1 (a)

2.1.1 (i)

Given $I = \int e^{-x^2/2}dx$, WTS $I^2 = 2\pi$ and $C = (2\pi)^{-1/2}$.

$$\begin{aligned}
I^2 &= \Big(\int e^{-x^2/2}dx\Big)\Big(\int e^{-y^2/2}dy\Big) \\
&= \iint e^{-(x^2+y^2)/2} r dx dy \\
&= \iint e^{-r^2/2} r dr d\theta
\end{aligned}$$

Let $u = -\frac{r^2}{2}$, $du = -rdr$.

$$\begin{aligned}
I^2 &= \iint -e^u du d\theta \\
&= \int -e^u d\theta \\
&= \int -e^{-r^2/2} d\theta \\
&= \int_0^{2\pi} \lim_{x\to\infty} -e^{-r^2/2}\Big|_0^x d\theta \\
&= \int_0^{2\pi} \lim_{x\to\infty} \Big(-\frac{1}{e^x} + e^0\Big) d\theta \\
&= \int_0^{2\pi} 1 d\theta \\
&= 2\pi
\end{aligned}$$

$$\begin{aligned}
I &= (2\pi)^{1/2} \\
C &= \frac{1}{I} = (2\pi)^{-1/2}
\end{aligned}$$

2.1.2 (ii)

Given $I = \int e^{-x^2/(2\sigma^2)}dx$, compute I^2 and C.
Let $u = -\frac{1}{2\sigma^2}r^2$, $du = -\frac{1}{\sigma^2}rdr$.

$$\begin{aligned}
I^2 &= \iint e^{-r^2/(2\sigma^2)}rdrd\theta \\
&= \int -\sigma^2 e^u d\theta \\
&= \int -\sigma^2 e^{-x^2/(2\sigma^2)}d\theta \\
&= \int_0^{2\pi} -\sigma^2 \lim_{x\to\infty} e^{-r^2/2}\Big|_0^x d\theta \\
&= \int_0^{2\pi} -\sigma^2 \lim_{x\to\infty} \Big(\frac{1}{e^x} - e^0\Big)d\theta \\
&= \int_0^{2\pi} \sigma^2 d\theta \\
&= 2\pi\sigma^2
\end{aligned}$$

Therefore,

$$\begin{aligned}
I &= (2\pi)^{1/2}\sigma \\
C &= \frac{1}{I} = (2\pi)^{-1/2}\sigma^{-1}
\end{aligned}$$

2.2 (b)

Given that $f(\mathbf{x}) = e^{-\mathbf{x}^T D\mathbf{x}/2}$, WTS $C = (2\pi)^{-d/2}|D|^{1/2}$.

$$\begin{aligned}
f(\mathbf{x}) &= exp\Big(-\frac{1}{2}[x_1 \dots x_d] \begin{bmatrix} \sigma_1^{-2} & & \\ & \ddots & \\ & & \sigma_d^{-2} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \Big) \\
&= exp(-\frac{1}{2}\sum_{i=1}^{d} x_i^2\sigma_i^2) \\
&= \prod_{i=1}^{d} e^{-x_i^2/(2\sigma_i^2)}
\end{aligned}$$

From (a)ii, we have $f(x_i) = e^{-x_i^2/(2\sigma_i^2)}$, and therefore $f(\mathbf{x}) = \prod_{i=1}^d f(x_i)$.

$$\begin{aligned}
\int f(x_i)dx_i &= (2\pi)^{1/2}\sigma_i \\
|D| &= \prod_{i=1}^{d}(\sigma_i)^{-2}\,.
\end{aligned}$$

We calculate I as:

$$\begin{aligned} I &= \int f(\mathbf{x}) d\mathbf{x} \\ &= \int \prod_{i=1}^{d} f(x_i) d\mathbf{x} \\ &= \prod_{i=1}^{d} \int f(x_i) dx_i \\ &= \prod_{i=1}^{d} (2\pi)^{1/2} \sigma_i \\ &= (2\pi)^{d/2} \prod_{i=1}^{d} \sigma_i \\ &= (2\pi)^{d/2} \Big(\prod_{i=1}^{d} \sigma_i^{-2} \Big)^{-1/2} \\ &= (2\pi)^{d/2} |D|^{-1/2} \end{aligned}$$

Therefore,

$$C = \frac{1}{I} = (2\pi)^{-d/2} |D|^{1/2}$$

2.3 (c)

2.3.1 (i)

In the case of 2D Gaussian function, the area under the curve $\int e^{-x^2/2\sigma^2}$ remains 1 regardless of the choice of σ. For 3D Gaussian, the choice of covariance matrix will only change the shape or orientation of the resulting 'solid', but not its volume. Similarly for higher dimensions, integrating over the density function produces the same result for axis-aligned D or generic covariance matrix Σ^{-1}. (Also note that $\Sigma^{-1} = Q^T D Q$ implies that Σ^{-1} can be transformed into D by a change of basis.) Therefore, $\int e^{-\mathbf{x}^T \Sigma^{-1} \mathbf{x}/2} = \int e^{-\mathbf{x}^T D \mathbf{X}/2}$ always holds.

2.3.2 (ii)

Given $\Sigma^{-1} = Q^T DQ$, WTS $|\Sigma^{-1}| = |D|$ By theorem, the determinant of the product is equal to the product of the determinants. $|AB| = |A||B|$.

$$\begin{aligned}
|\Sigma^{-1}| &= |Q^T DQ| \\
&= |Q^T||D||Q| \\
&= |Q^T||Q||D| \\
&= |Q^T Q||D| \\
&= |I||D| \\
&= |D|
\end{aligned}$$

Therefore,

$$\begin{aligned}
C &= (2\pi)^{-d/2}|D|^{1/2} \\
&= (2\pi)^{-d/2}|\Sigma^{-1}|^{1/2} \\
&= (2\pi)^{-d/2}|\Sigma|^{-1/2} \\
&= \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}
\end{aligned}$$

3 Problem 3

3.1 (a)

$$\begin{aligned}
l(\theta) &= \sum_{i=1}^{n} \log(p(\mathbf{x}_i, z_i)) \\
&= \sum_{i=1}^{n} \log(\pi_{z_i}\mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_{z_i}, \boldsymbol{\Sigma}_{z_i})) \\
&= \sum_{i=1}^{n} \log(\pi_{z_i}(2\pi)^{-d/2}|\boldsymbol{\Sigma}_{z_i}|^{-1/2} exp(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_{z_i})^T \boldsymbol{\Sigma}_{z_i}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{z_i}))) \\
&= \sum_{i=1}^{n} \log((2\pi)^{-d/2} + \quad \log(\pi_{z_i}) + \quad \log(|\boldsymbol{\Sigma}_{z_i}|^{-1/2}) \\
&+ \quad \log(exp(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_{z_i})^T \boldsymbol{\Sigma}_{z_i}^{-1}(\mathbf{x} - \boldsymbol{\mu}_{z_i}))) \\
&= \sum_{i=1}^{n} \Big(\log(\pi_{z_i}) - \quad \frac{1}{2}\log(|\boldsymbol{\Sigma}_{z_i}|) - \quad \frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_{z_i})^T \boldsymbol{\Sigma}_{z_i}^{-1}(\mathbf{x} - \boldsymbol{\mu}_{z_i}) \Big)
\end{aligned}$$

Note: the constant $\log((2\pi)^{-d/2})$ is dropped.

3.2 (b)

By Bayes' rule,

$$\begin{aligned}
p(z_i = j|\mathbf{x}_i) &= \frac{p(\mathbf{x}_i|z_i = j)p(z_i = j)}{p(\mathbf{x}_i)} \\
&= \frac{p(\mathbf{x}_i, z_i = j)}{\sum_{l=1}^{k} p(\mathbf{x}_i, z_i = l)} \\
&= \frac{\pi_j \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{l=1}^{k} \pi_l \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} \\
&= \frac{\pi_j \quad \cdot \quad (2\pi)^{-d/2}|\boldsymbol{\Sigma}_j|^{-1/2} exp(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_j))}{\sum_{l=1}^{k} \Big(\pi_l \quad \cdot \quad (2\pi)^{-d/2}|\boldsymbol{\Sigma}_l|^{-1/2} exp(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_l)^T \boldsymbol{\Sigma}_l^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_l))\Big)}
\end{aligned}$$

Therefore,

$$p_{i,j} = \frac{\pi_j \quad \cdot \quad (2\pi)^{-d/2}|\boldsymbol{\Sigma}_j|^{-1/2} exp(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_j))}{\sum_{l=1}^{k} \Big(\pi_l \quad \cdot \quad (2\pi)^{-d/2}|\boldsymbol{\Sigma}_l|^{-1/2} exp(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_l)^T \boldsymbol{\Sigma}_l^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_l))\Big)}$$

3.3 (c)

Since z_i's are independent, by the linearity of expectation, we fix $z_i = j$ in calculating its log-likelihood, and then sum over all $z_1, \ldots, z_n$.

$$\begin{aligned}
\bar{l}_{\theta_{old}}(\theta) &= \sum_{i=1}^{n} \sum_{j=1}^{k} p_{i,j} l_{z_i=j}(\theta) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{k} p_{i,j} \Big(\log(\pi_j) - \quad \frac{1}{2}\log(|\boldsymbol{\Sigma}_j|) - \quad \frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_j)\Big)
\end{aligned}$$

3.4 (d)

$$\bar{l}_{\theta_{old}}(\theta) = \sum_{i=1}^{n} \sum_{j=1}^{k} p_{i,j} \log(\pi_j \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j))$$

$\bar{l}_{\theta_{old}}(\theta)$ is maximized when $\frac{d\bar{l}_{\theta_{old}}(\theta)}{d\pi_j} = 0$, and to account for the constraint $\sum_{j=1}^{k} \pi_j = 1$, we introduce a Lagrange multiplier.

$$\begin{aligned}
f(\theta) &= \sum_{i=1}^{n}\sum_{j=1}^{k} p_{i,j} \log(\pi_j \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)) + \lambda\Big(\sum_{i=1}^{k} \pi_j - 1\Big) \\
\frac{df(\theta)}{d\pi_j} &= \frac{1}{\pi_j \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \sum_{i=1}^{n} p_{i,j} + \lambda \\
&= \frac{1}{\pi_j} \sum_{i=1}^{n} p_{i,j} + \lambda = 0 \\
0 &= \sum_{i=1}^{n} p_{i,j} + \lambda \pi_j \\
\pi_j &= -\frac{1}{\lambda} \sum_{i=1}^{n} p_{i,j} \\
\sum_{j=1}^{k} \pi_j &= \sum_{j=1}^{k} -\frac{1}{\lambda} \sum_{i=1}^{n} p_{i,j} = -\frac{1}{\lambda} \sum_{i=1}^{n}\sum_{j=1}^{k} p_{i,j} = -\frac{1}{\lambda} \cdot n = 1 \\
\lambda &= -n \\
\pi_j &= \frac{1}{n} \sum_{i=1}^{n} p_{i,j}
\end{aligned}$$

Therefore, to maximize the expected log-likelihood,

$$\pi_j \leftarrow \frac{1}{n} \sum_{i=1}^{n} p_{i,j}$$

3.5 (e)

$$\boldsymbol{\mu}_j \leftarrow \frac{\sum_{i=1}^{n} p_{i,j} \mathbf{x}_i}{\sum_{i=1}^{n} p_{i,j}}$$

The calculation is as follows,

$$
\begin{aligned}
f(\theta) &= \sum_{i=1}^{n}\sum_{j=1}^{k} p_{i,j}\Big(\log(\pi_j) - \quad \frac{1}{2}\log(|\boldsymbol{\Sigma}_j|) - \quad \frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_j)^T\boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_j)\Big) \\
\frac{df(\theta)}{\boldsymbol{\mu}_j} &= \sum_{i=1}^{n} p_{i,j}\boldsymbol{\Sigma}_j(\mathbf{x}_i - \boldsymbol{\mu}_j) = \sum_{i=1}^{n} p_{i,j}\boldsymbol{\Sigma}_j\boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_j) = \sum_{i=1}^{n} p_{i,j}(\mathbf{x}_i - \boldsymbol{\mu}_j) = 0 \\
\sum_{i=1}^{n} p_{i,j}\boldsymbol{\mu}_j &= \sum_{i=1}^{n} p_{i,j}\mathbf{x}_i \\
\mu_j &= \frac{\sum_{i=1}^{n} p_{i,j}\mathbf{x}_i}{\sum_{i=1}^{n} p_{i,j}}
\end{aligned}
$$

3.6 (f)

These update rules correspond to the two update steps in k-means. Namely, $\pi_j \leftarrow \frac{1}{n}\sum_{i=1}^{n} p_{i,j}$ corresponds to the cluster assignment $\gamma_i \leftarrow \text{argmin}_{j\in\{1,2,\ldots,k\}}\, d(\mathbf{x}_i, \mathbf{m}_j)$. On the other hand, $\boldsymbol{\mu}_j \leftarrow \frac{\sum_{i=1}^{n} p_{i,j}\mathbf{x}_i}{\sum_{i=1}^{n} p_{i,j}}$ corresponds to the centroid location update $\mathbf{m}_j \leftarrow \frac{1}{|C_j|}\sum_{i:\gamma_i=j}\mathbf{x}_i$. Similar to how we minimize distortion J in k-means, in this generative model, we maximize our expected log-likelihood with these two update steps.

4 Problem 4

4.1 (a)

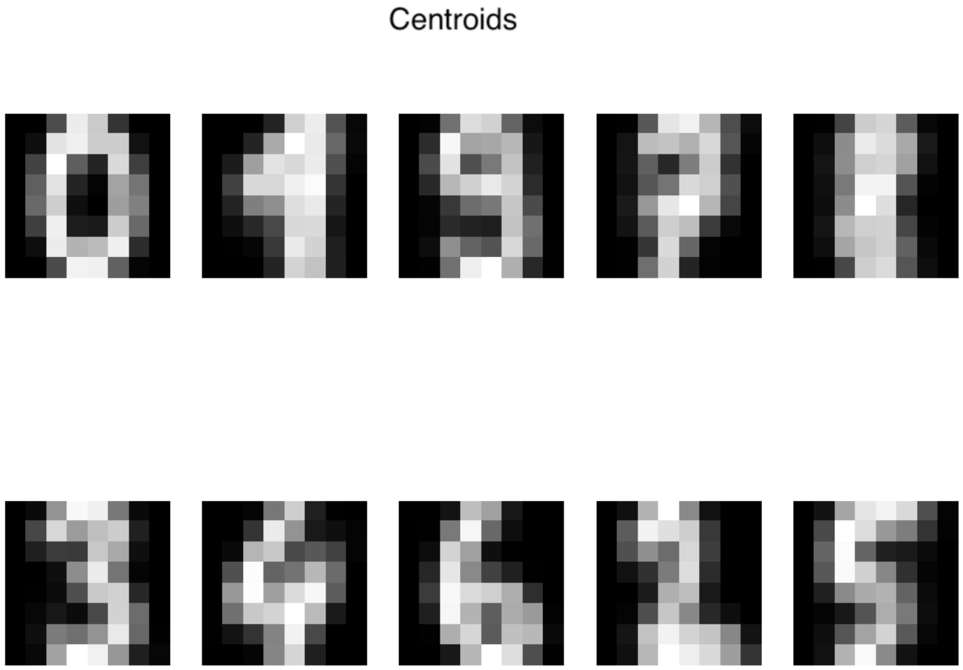

Figure 1: Resultant Centroids

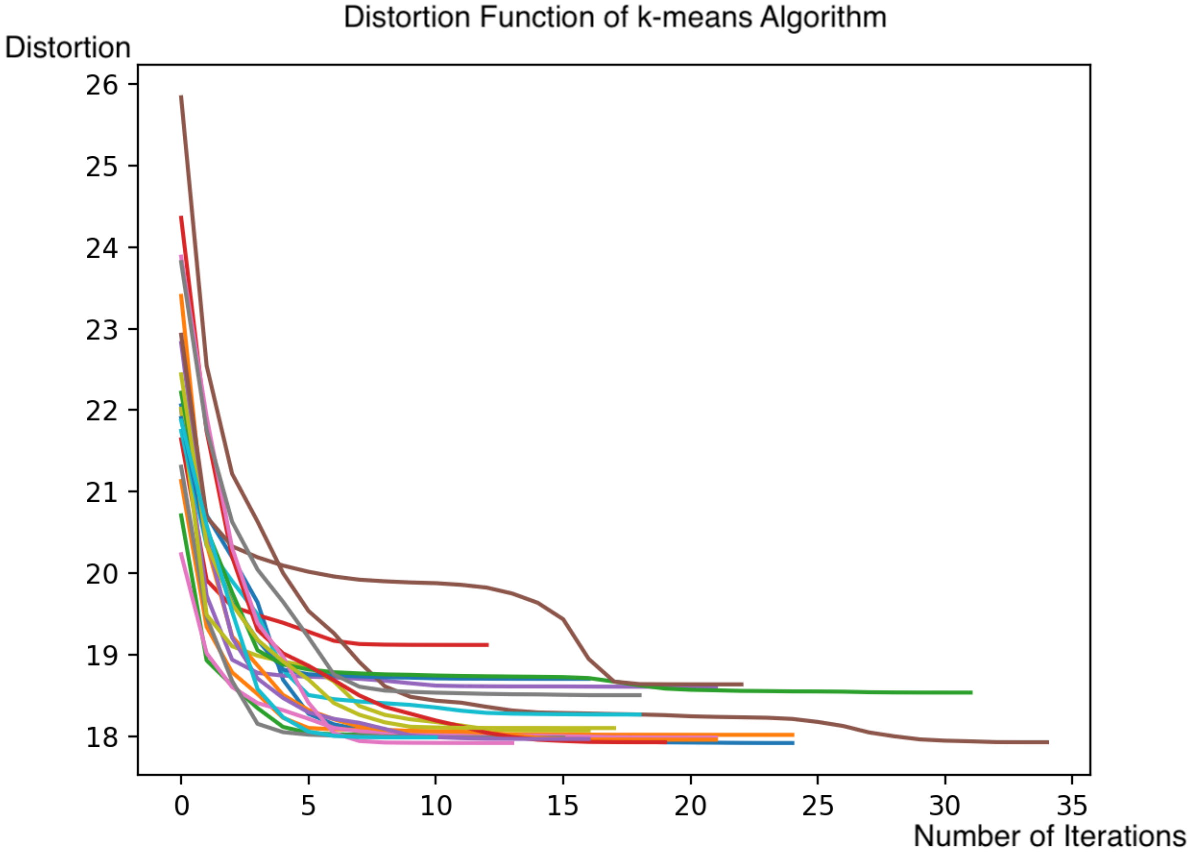

Figure 2: Distortion Function of k-means Algorithm

Comment: As shown in Figure 2, there is one global minimum as well as several local minima. Only the global minimum will generate the desired clustering result as shown in Figure 1, while the local minima generate erroneous clustering. Most trials of the k-means algorithm converge to the global minimum in fewer than 25 iterations.

4.2 (b)

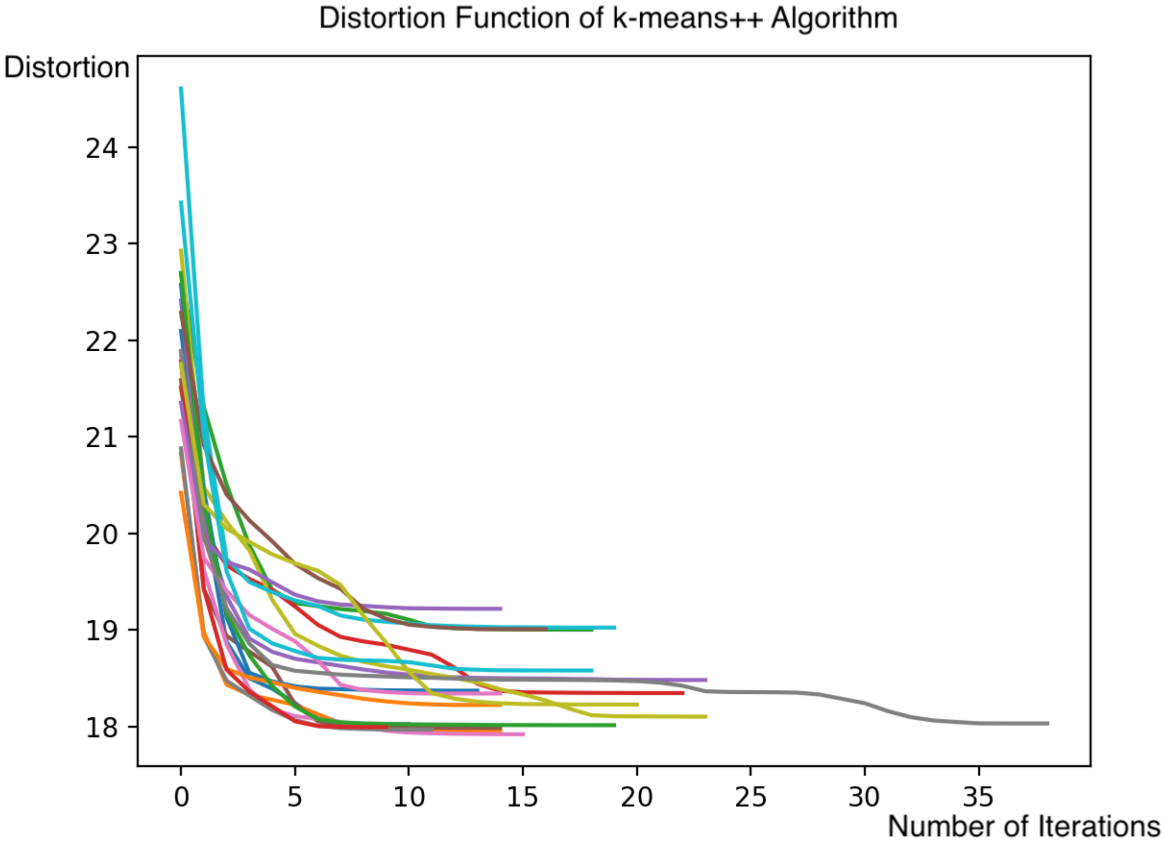

Figure 3: Distortion Function of k-means++ Algorithm

Comment: Similar to Figure 2, there is one global minimum as well as several local minima in Figure 3. With the k-means++ Algorithm, most of the trials converge to the global minimum in 15 - 25 iterations, which is fewer than the steps taken in k-means. This is expected, since k-means++ provides an optimized initialization of the centroids, which helps the algorithm to converge faster while decreasing the chance of getting caught in local minima.