Andrew Elysee (aelys2)
Discussion: Max Han (mhan36)
CS446

HW1

1.

a.

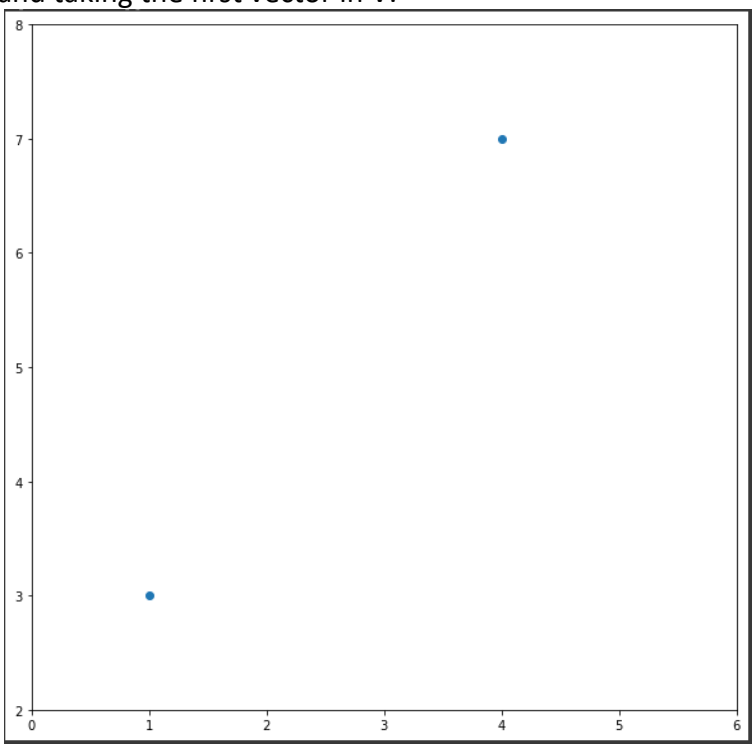
i. False, PCA seeks a subspace that reduces reconstruction error this means it seeks to minimize the orthogonal error (distance) to our model line for all points.

ii. False because least squares is trying to reduce the residuals while PCA is trying to reduce orthogonal distance from points to the component vectors.

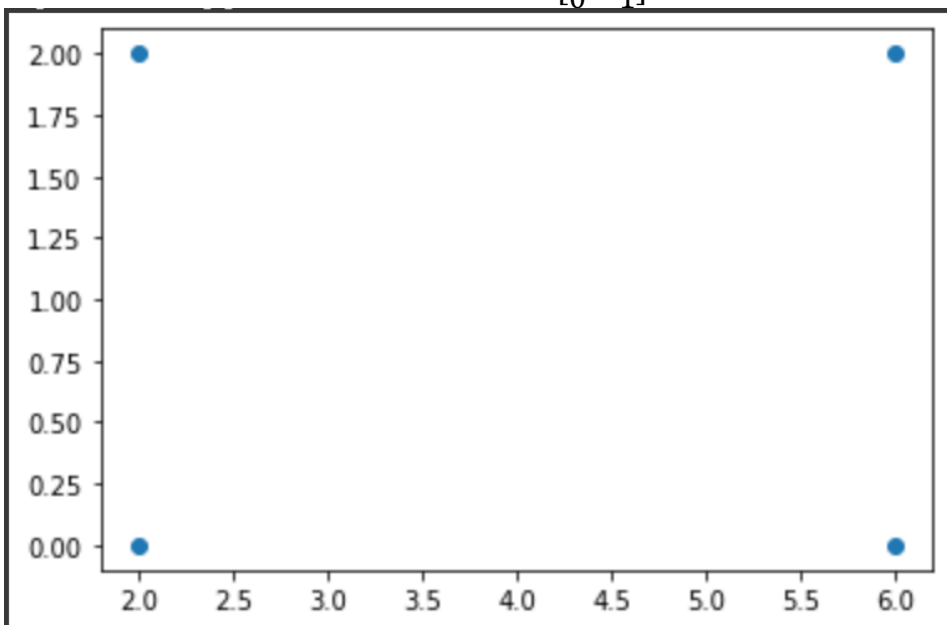
iii. True, the goal of PCA is to find the component vectors that maximize variance.

iv. False, the principal components must be orthogonal because they are derived from the eigenvectors of the covariance matrix which are all orthogonal to each other due to the properties of eigenvectors.

b. The first Principal Component $w = [0.6, 0.8]^T$. This makes sense because the slope between the 2 points is 4/3 and the most variance would be along the line connecting the 2 points. You could calculate this by finding the SVD of the matrix and taking the first vector in V.



c. Given this dataset the dimensions of the covariance matrix will still be 2x2 since there are only 2 features in the array. $\Sigma = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$



d. Given that Σ is a diagonal matrix, that means its eigen values are equal to the entries on the diagonal. So the 3rd diagonal entry is the largest which means that eigenvector corresponding to the 3rd diagonal is our optimal w. This means that $w = [0,\ 0, 1, 0]^T$ and our optimal value is 20.

2.

a. K-means is an unsupervised method because there are no given labels on the data.

b. To ensure hard assignment of one data point to one and only one cluster we need to add a constraint on r_{ik} s.t. $r_{ik} \in \{0,1\}\ \forall i \in D, k \in \{1,2, \dots, K\}$ and $\sum_{k \in \{1,2,\dots,K\}} r_{ik} = 1\ \forall i \in D$. This constraint is basically saying that for each centroid each data point will be assigned a value of 0 or 1 in the vector r_i and sum of the elements of this vector must equal 1. This means if more than one centroid is assigned to a data point the magnitude of the vector would be more than 1 and thus would not be valid, and if no centroid is assigned the magnitude would be 0 which would not be valid.

c. If we wanted to achieve soft assignment, we would just remove the constraint that says that says $r_{ik} \in \{0,1\}$ and replace it with a constraint that says $r_{ik} \in [0,1]$. Basically, instead of r_{ik} being either 0 or 1, r_{ik} must be a real number such that $0 \leq r_{ik} \leq 1$ such that the sum of the elements of $r_i = 1$. This would assign a probability that a point belongs to each centroid in the vector.

d. 5 clusters because at this point the squared distance of all points from their cluster center no longer decreases

e. K means would not be an efficient algorithm to cluster this data. K-means is not effective at separating clusters that surround other clusters as K-means only does linear division between the clusters. The orange points do not form a traditional

cluster. In other words, it separates clusters using lines or planes, not curves. It could be possible to cluster this data using K-means if you used a complex kernel for determining distance, but normal Euclidean distance K-means would not be effective. As it would divide the points straight down the middle leaving both blue and orange points on both sides.

3.

a. $\mu_k \in \mathbb{R}^f$ where f is the dimensionality of x.
b. The optimal $r_{x,k}$ for the program in equation 2 is given by:

$$r_{x,k} = \begin{cases} 1 & if\ k = \arg\min_{k \in \{1,\dots,K\}} \left|\left|x - \mu_k\right|\right|_2^2 \\ 0 & otherwise \end{cases}$$

This is the optimal $r_{x,k}$ because $r_{x,k}$ should only be 1 if centroid k is the closest centroid to the data point x, and we can only have one centroid for which this is true.

NOTE: Source class slides for lecture 5

c. Given a fixed $r_{x,k}$ $\forall x \in D, k \in \{1, \dots, K\}$ if we take the derivative with respect to μ_k, we get:

$$\sum_{i \in D} r_{x,k}\left(x^{(i)} - \mu_k\right) = 0$$

Thus, the optimal centers μ_k are:

$$\mu_k = \frac{\sum_{i \in D} r_{x,k} x^{(i)}}{\sum_{i \in D} r_{x,k}}$$

NOTE: Source is class slides for lecture 5

d. This algorithm is Guaranteed to converge because with every update the sum of distances to the center is reduced since the centers are updated with every iteration. It is not guaranteed to find the global optimum because if you start with different starting centers, it is possible to end at different centroids, so it clearly does not always reach the global optimum if this is the case.

Pseudo code

K_means(inputs: X, c):

p_r = an array of dimension n x m where m = # of centroids and n = # of data points

r = an array of dimension n x m where m = # of centroids and n = # of data points

While first iteration or p_r is not equal to previous r:

$p_r = r$

For data point x_i in X such that $r_{i,j} = 1$:

$dist$ = a vector of size n where n is number of centroids

For each centroid μ_k where k starts at 0:

$dist_k = ||x_i - \mu_k||_2^2$

$r_{i,k} = 1$ such that k is the index of smallest element in $dist$

while j is less than # of centroids, with j starting at 0:

c_j= an vector of 0's

$count = 0$

For data point x_i in X such that $r_{i,j} = 1$:

c_j += x_i

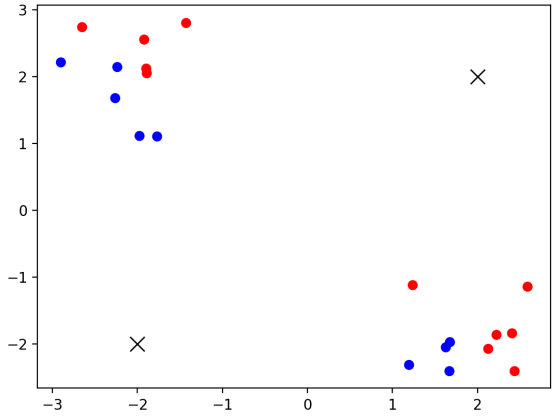
$count += 1$

$c_j = \frac{c_j}{count}$

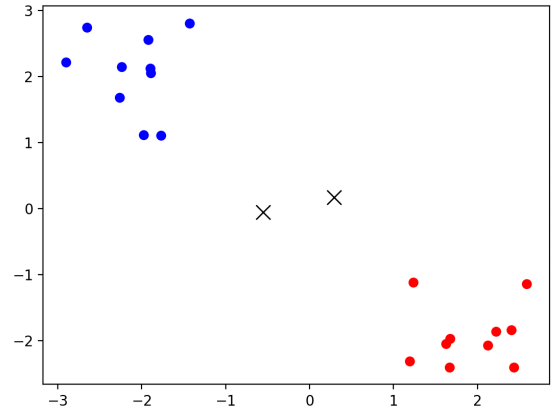
return c and finish

e. The cost function value was 243.039505 and the centroids are at $(1.9163, -1.9143)$ and $(-2.0952, 2.0540)$. For the given dataset, it took only 2 iterations to converge.

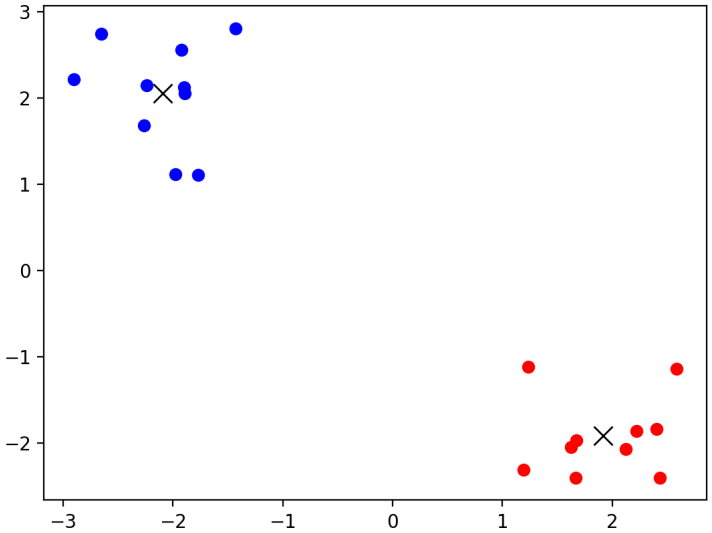
First Clustering



Second clustering After first update



Final Clustering After Second Update



4.

a. The log-likelihood is

$$\log \prod_{i \in D} P\left(x^{(i)} \middle| \pi_k, \mu_k, \sigma_k\right) = \sum_{i \in D} \log \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mathrm{x}^{(\mathrm{i})} \middle| \mu_\mathrm{k}, \sigma_\mathrm{k}\right)$$

$$= \sum_{i \in D} \log \sum_{k=1}^{K} \frac{\pi_k}{\sigma_k \sqrt{2\pi}} e^{\frac{-\left(x^{(i)} - \mu_k\right)^2}{2\sigma_k^2}}$$

NOTE: source is the slides for lecture 6

b. If $K = 1$, then $\pi_1 = 1$. Thus, our log-likelihood is:

$$\log \prod_{i \in D} P\left(x^{(i)} \middle| \pi_1, \mu_1, \sigma_1\right) = \sum_{i \in D} \log \frac{\pi_k}{\sigma_1 \sqrt{2\pi}} e^{\frac{-\left(x^{(i)} - \mu_1\right)^2}{2\sigma_1^2}}$$

$$= \sum_{i \in D} \frac{-\left(x^{(i)} - \mu_1\right)^2}{2\sigma_1^2} - \frac{|D|}{2} \log(2\pi\sigma^2)$$

$$\frac{\partial}{\partial \mu_1} : \frac{1}{\sigma_1^2} \sum_{i \in D} \left(x^{(i)} - \mu_1\right) = 0 \qquad \Longrightarrow \mu_1 = \frac{1}{|D|} \sum_{i \in D} x^{(i)}$$

$$\frac{\partial}{\partial \sigma} : -\frac{1}{\sigma^3} \sum_{i \in D} (x^{(i)} - \mu_1)^2 + \frac{|D|}{\sigma} \quad \Longrightarrow \sigma_1 = \frac{1}{|D|} \sum_{i \in D} \left(x^{(i)} - \mu_1\right)^2 \Longrightarrow$$

$$\sigma_1^2 = \left(\frac{1}{|D|} \sum_{i \in D} \left(x^{(i)} - \mu_1\right)^2\right)^2$$

NOTE: Source is the slides for lecture 6

c. If latent variable $z_{i,k} = 1$, then it uses the gaussian distribution of the *K*th gaussian. The underlying distribution of $p(z_i)$ is:

$$p(z_i) = \prod_{k=1}^{K} \pi_k^{z_{i,k}}$$

d. NOTE: Source is slides from class

$$p\left(z_{ik} = 1 \middle| x^{(i)}\right) = \frac{p(z_{ik} = 1) p\left(x^{(i)} \middle| z_{ik} = 1\right)}{\sum_{\hat{k}=1}^{K} p(z_{i\hat{k}} = 1) p\left(x^{(i)} \middle| z_{i\hat{k}} = 1\right)} = \frac{\pi_k \mathcal{N}\left(\mathrm{x}^{(\mathrm{i})} \middle| \mu_\mathrm{k}, \sigma_\mathrm{k}\right)}{\sum_{\hat{k}=1}^{K} \pi_{\hat{k}} \mathcal{N}\left(\mathrm{x}^{(\mathrm{i})} \middle| \mu_{\hat{k}}, \sigma_{\hat{k}}\right)}$$

e. Both K-means and Gaussian Mixture model use μ_k as the cluster centers and r_{ik} to assign a sample I to a cluster k. Gaussian Mixture Model just uses soft assignment but if we fix $\sigma_k^2 = \epsilon \; \forall k$, then as $\epsilon \to 0$, r_{ik} will act like a hard assignment, like k-means.

f. $\lim_{\epsilon \to 0} -\epsilon \log \sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right) = \lim_{\epsilon \to 0} \frac{\log \sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right)}{-\epsilon^{-1}} = \lim_{\epsilon \to 0} \frac{\frac{\sum_{k=1}^{K} F_k \exp\left(-\frac{F_k}{\epsilon}\right)}{\epsilon^2}}{\frac{\sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right)}{\epsilon^{-2}}} =$

$\lim_{\epsilon \to 0} \frac{\frac{\sum_{k=1}^{K} F_k \exp\left(-\frac{F_k}{\epsilon}\right)}{\epsilon^2 \sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right)}}{\epsilon^{-2}} = \lim_{\epsilon \to 0} \frac{\epsilon^2 \sum_{k=1}^{K} F_k \exp\left(-\frac{F_k}{\epsilon}\right)}{\epsilon^2 \sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right)} = \lim_{\epsilon \to 0} \frac{e^2}{e^2} * \lim_{\epsilon \to 0} \frac{\sum_{k=1}^{K} F_k \exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right)} = 1 *$

$\lim_{\epsilon \to 0} \frac{\sum_{k=1}^{K} F_k \exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K} \exp\left(-\frac{F_k}{\epsilon}\right)}$

We use L'Hopital's rule to get $\lim_{\epsilon\to 0}\frac{\frac{\sum_{k=1}^{K}F_k\exp\left(-\frac{F_k}{\epsilon}\right)}{\epsilon^2}}{\frac{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}{\epsilon^{-2}}}$ and $\lim_{\epsilon\to 0}\frac{e^2}{e^2}=1$. From here we can see that:

$$\lim_{\epsilon\to 0}\frac{\sum_{k=1}^{K}F_k\exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}=\lim_{\epsilon\to 0}\sum_{k=1}^{K}\frac{F_k\exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}$$
$$=\lim_{\epsilon\to 0}\sum_{k=1}^{K}F_k\frac{\exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}$$

WLOG suppose there is some F_i such that $F_i < F_k\ \forall k\ s.t.\ k\neq i$. As $\epsilon\to 0, -\frac{1}{\epsilon}\to -\infty$ so because in our exponents we have $-\frac{F}{\epsilon}$ the exponent with the smallest F will approach $-\infty$ the slowest. And we know that as $x\to\ -\infty,\ e^x\to 0$. Thus, $\frac{\exp\left(-\frac{F_i}{\epsilon}\right)}{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}$ will converge to 1 for the smallest F (in this case F_i), but $\frac{\exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}$ will converge to 0 for all the F that are not the smallest. So, this means that:

$$\lim_{\epsilon\to 0}-\epsilon\log\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)=\lim_{\epsilon\to 0}\sum_{k=1}^{K}F_k\frac{\exp\left(-\frac{F_k}{\epsilon}\right)}{\sum_{k=1}^{K}\exp\left(-\frac{F_k}{\epsilon}\right)}=\min_k F_k$$

g. Using what we showed in 4f if we make $F_k=(x-\mu_k)^2$ as $\epsilon\to 0$:

$$\min_{\mu}-\sum_{x_i\in D}\epsilon\log\sum_{k=1}^{K}\exp\left(-\frac{(x_i-\mu_k)^2}{\epsilon}\right)$$
$$=\min_{\mu}\sum_{x_i\in D}-\epsilon\log\sum_{k=1}^{K}\exp\left(-\frac{(x_i-\mu_k)^2}{\epsilon}\right)$$
$$=\min_{\mu}\sum_{x_i\in D}\min_k\ (x_i-\mu_k)^2$$

This is the same as the cost function for K-means because since we are trying to minimize along k that means that only one of the centroids 1 through k can be chosen to optimize for each point, in other words, we are performing the hard assignment that occurs in the k-means algorithm. So, the objective for k-Means is the 0-temperature limit of Gaussian Mixture Model.