

# CS 446 / ECE 449 — Homework 4

*junsukh2*

Version 1.0

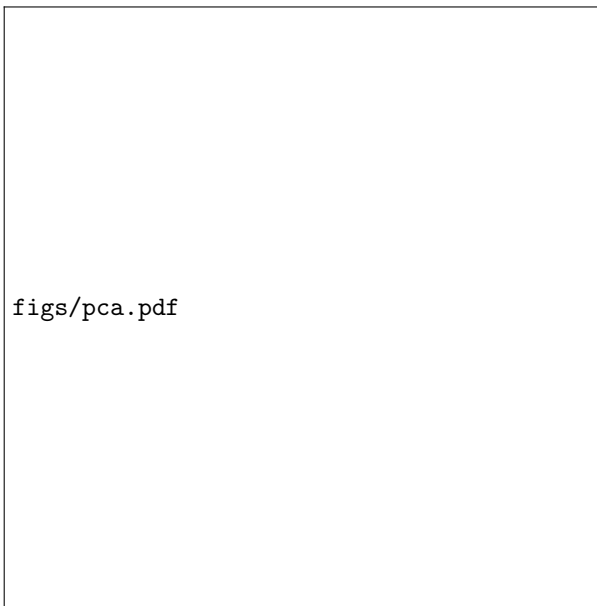
## Instructions.

- Homework is due **Tuesday, April 6th, at noon CST**; no late homework accepted.
- Everyone must submit individually at gradescope under **hw4** and **hw4code**.
- The “written” submission at **hw4** **must be typed**, and submitted in any format gradescope accepts (to be safe, submit a PDF). You may use L<sup>A</sup>T<sub>E</sub>X, markdown, google docs, MS word, whatever you like; but it must be typed!
- When submitting at **hw4**, gradescope will ask you to mark out boxes around each of your answers; please do this precisely!
- Please make sure your NetID is clear and large on the first page of the homework.
- Your solution **must** be written in your own words. Please see the course webpage for full academic integrity information. Briefly, you may have high-level discussions with at most 3 classmates, whose NetIDs you should place on the first page of your solutions, and you should cite any external reference you use; despite all this, your solution must be written in your own words.
- We reserve the right to reduce the auto-graded score for **hw4code** if we detect funny business (e.g., your solution lacks any algorithm and hard-codes answers you obtained from someone else, or simply via trial-and-error with the autograder).
- When submitting to **hw4code**, only upload **hw4.py** and **hw4\_utils.py**. Additional files will be ignored.

# 1. Principal Component Analysis

(a) For each of the following statements, specify whether the statement is true or false. If you think the statement is wrong, explain in 1 to 2 sentences why it is wrong.

- True or False: As shown in the figure below, PCA seeks a subspace such that the sum of all the vertical distance to the subspace (the dashed line) is minimized.



- True or False: PCA seeks a projection that best represents the data in a least-squares sense.
- True or False: PCA seeks a linear combination of variables such that the maximum variance is extracted from the variables.
- True or False: The principal components are not necessarily orthogonal to each other.
- True or False: Solving PCA using SVD might result in a solution which corresponds to a local minimum.

(b) Recall that PCA finds a direction  $w$  in which the projected data has highest variance by solving the following program:

$$\max_{w: ||w||^2=1} w^T \Sigma w. \quad (1)$$

Here,  $\Sigma$  is a covariance matrix. You are given a dataset of two 2-dimensional points  $(1, 3)$  and  $(4, 7)$ . Draw the two data points on the 2D plane. What is the first principal component  $w$  of this dataset?

(c) Now you are given a dataset of four points  $(2, 0)$ ,  $(2, 2)$ ,  $(6, 0)$  and  $(6, 2)$ . Draw the four data points on the 2D plane. Given this dataset, what is the dimension of the covariance matrix  $\Sigma$  in Eq. (1)? Also, explicitly write down the values of  $\Sigma$  given the dataset.

(d) What is the optimal  $w$  and the optimal value of the program in Eq. (1) given

$$\Sigma = \begin{bmatrix} 12 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 \\ 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}$$

Solution:

(a)

- False: Not vertical line, but perpendicular line to graph.
- True
- True
- False: We set that eigenvectors are orthogonal to each other.
- False: Singular values are related to eigenvalues. PCA objective has no non-optimal local minima.

(b)

$\bar{X} = \begin{bmatrix} -1.5 & 1.5 \\ -2 & 2 \end{bmatrix}$ ,  $\Sigma = \begin{bmatrix} 2.25 & 3 \\ 3 & 4 \end{bmatrix}$ . So an eigenvector corresponding to the largest eigenvalue is  $\begin{bmatrix} 3/4 \\ 1 \end{bmatrix}$  and the vector of norm = 1 is  $\begin{bmatrix} 3/5 \\ 4/5 \end{bmatrix}$

(c)

$\bar{X} = \begin{bmatrix} -2 & -2 & 2 & 2 \\ -1 & 1 & -1 & 1 \end{bmatrix}$ , so  $\Sigma = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$

$\text{Dim}(\Sigma) = 2 * 2$ .

(d)

$\lambda_{\max} = 20$  where  $\lambda$  is an eigenvalue. The corresponding eigenvector  $w = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$  and optimal value of the program is 20.

## 2. Gaussian Mixture Models

Consider a Gaussian mixture model with  $K$  components ( $k \in \{1, \dots, K\}$ ), each having mean  $\mu_k$ , variance  $\sigma_k^2$ , and mixture weight  $\pi_k$ . Further, we are given a dataset  $\mathcal{D} = \{x_i\}$ , where  $x_i \in \mathbb{R}$ . We use  $z_i = \{z_{ik}\}$  to denote the latent variables.

- What is the log-likelihood of the data according to the Gaussian Mixture Model? (use  $\mu_k$ ,  $\sigma_k$ ,  $\pi_k$ ,  $K$ ,  $x_i$ , and  $\mathcal{D}$ ).
- Assume  $K = 1$ , find the maximum likelihood estimate for the parameters  $(\mu_1, \sigma_1^2, \pi_1)$ .
- What is the probability distribution on the latent variables, i.e., what is the distribution  $p(z_{i,1}, z_{i,2}, \dots, z_{i,K})$  underlying Gaussian mixture models. Also give its name.
- For general  $K$ , what is the posterior probability  $p(z_{ik} = 1 | x_i)$ ? To simplify, wherever possible, use  $\mathcal{N}(x_i | \mu_k, \sigma_k)$ , a Gaussian distribution over  $x_i \in \mathbb{R}$  having mean  $\mu_k$  and variance  $\sigma_k^2$ .
- How are k-Means and Gaussian Mixture Model related? (There are three conditions)
- Consider the modified Gaussian Mixture Model objective:

$$\min_{\mu} - \sum_{x_i \in \mathcal{D}} \epsilon \log \sum_{k=1}^K \exp(-(x_i - \mu_k)^2 / \epsilon).$$

We denote by  $F_k = (x - \mu_k)^2$ . Show that:

$$\lim_{\epsilon \rightarrow 0} \epsilon \log \sum_{k=1}^K \exp(-F_k / \epsilon) = \sum_{k=1}^K \mathbb{1}_{\{k = \arg\min_k F_k\}} F_k, \quad \epsilon \in \mathbb{R}^+$$

**Hint:** Use l'Hopital rule.

- Conclude that the objective for k-Means is the 0-temperature limit of Gaussian Mixture Model under the conditions you provided in part (e).

Solution:

(a)

$$\sum_{i \in \mathcal{D}} \log \sum_{k=1}^K \pi_k \mathcal{N}(x^{(i)} | \mu_k, \sigma_k)$$

(b)

$$\mu_1 = \frac{1}{N} \sum_{i \in \mathcal{D}} x^{(i)} \text{ where } N \text{ is the size of data set.}$$

$$\sigma_1^2 = \frac{1}{N} \sum_{i \in \mathcal{D}} (x^{(i)} - \mu_1)^2.$$

$$\pi_1 = 1 \text{ as } \sum_k \pi_k = 1.$$

(c)

$$P(\mathbf{z}_i) = \prod_{k=1}^K \pi_k^{z_{ik}}. \text{ Categorical distribution}$$

(d)

$$\frac{\pi_k \mathcal{N}(x^{(i)} | \mu_k, \sigma_k)}{\sum_{k'=1}^K \pi_{k'} \mathcal{N}(x^{(i)} | \mu_{k'}, \sigma_{k'})}$$

(e)

$$\sigma_k^2 \rightarrow 0$$

$$\pi_k = 1/K, \text{ None of } \pi_k \text{ is } 0$$

$$\text{smallest } (x^{(i)} - \mu_k)^2 \text{ is unique for each data point.}$$

(f)

$$\lim_{\epsilon \rightarrow 0} -\epsilon \log \sum_{k=1}^K \exp(-F_k / \epsilon)$$

Assume that  $k = \operatorname{argmin}_k F_k$   
 $= \lim_{\epsilon \rightarrow 0} -\epsilon \log \exp(-F_k/\epsilon) = \lim_{\epsilon \rightarrow 0} -\epsilon(-F_k/\epsilon)$

By l'Hopital Rule,  
 $= \lim_{\epsilon \rightarrow 0} -(-F_k) = F_k = \sum_{k=1}^K \mathbb{1}_{\{k=\operatorname{argmin}_k F_k\}} F_k, \quad \epsilon \in \mathbb{R}^+$  since indicator is true only when index is  $k$ .

(g)

Since  $k = \operatorname{argmin}_k F_k$  is unique for each data, and  $\pi_k$  is constant,

$$\lim_{\epsilon \rightarrow 0} -\sum_{x_i \in \mathcal{D}} \epsilon \log \sum_{k=1}^K \exp(-F_k/\epsilon) = \sum_{x_i \in \mathcal{D}} \sum_{k=1}^K \mathbb{1}_{\{k=\operatorname{argmin}_k F_k\}} F_k = \sum_{x_i \in \mathcal{D}} \sum_{k=1}^K r_{ik} \left\| x^{(i)} - \mu_k \right\|_2^2$$

where  $r_{ik}$  is the indicator for all combinations of each data and cluster.

### 3. Expectation Maximization

In this problem, you will implement an expectation-maximization (EM) algorithm to cluster samples  $\mathcal{D} = \{x^{(i)}\}_{i=1}^n$ , with  $x^{(i)} \in \{0, 1\}^D$  into groups. You will be using a mixture of Bernoullis model to tackle this problem.

(a) **Mixture of Bernoullis.**

- i. Assume each variable  $x_d$  is drawn from a Bernoulli( $q_d$ ) distribution,  $P(x_d = 1) = q_d$ . Let  $q = [q_1, \dots, q_d] \in [0, 1]^D$  be the resulting vector of Bernoulli parameters. Write an expression for  $P(x|q)$  as a function of  $q_d$  and  $x_d$ .
- ii. Now suppose we have a mixture of  $K$  Bernoulli distributions: each vector  $x^{(i)}$  is drawn from some vector of Bernoulli random variables with parameters  $p^{(k)} = [p_1^{(k)}, \dots, p_d^{(k)}]$ , that we call Bernoulli( $p^{(k)}$ ). Let  $\{p^{(1)}, \dots, p^{(K)}\} = p$ . Assume a distribution  $\pi$  over the selection of which set of Bernoulli parameters  $p^{(k)}$  is chosen. Write an expression for  $P(x^{(i)}|p, \pi)$ , as a function of  $\pi_k$  and  $P(x|p^{(k)})$ . Here  $\pi_k$  denotes the probability associated with the  $k^{\text{th}}$  Bernoulli component.
- iii. Using the above, write an expression for the log-likelihood of the data  $\mathcal{D}$ ,  $\log P(\mathcal{D}|\pi, p)$ .

(b) **Expectation step.**

- i. Let  $z^{(i)} \in \{0, 1\}^K$  be an indicator vector, such that  $z_k^{(i)} = 1$  if  $x^{(i)}$  was drawn from a Bernoulli( $p^{(k)}$ ), and 0 otherwise. Let  $Z = \{z^{(i)}\}_{i=1}^n$ . Write down the expression of  $P(z^{(i)}|\pi)$  as a function of  $\pi_k$  and  $z_k^{(i)}$ .
- ii. Write down the expression of  $P(x^{(i)}|z^{(i)}, p, \pi)$  as a function of  $P(x^{(i)}|p^{(k)})$  and  $z_k^{(i)}$ .
- iii. Using the two quantities above, derive the likelihood of the data and latent variables  $P(Z, \mathcal{D}|\pi, p)$ .
- iv. Let  $\eta(z_k^{(i)}) = \mathbb{E}[z_k^{(i)}|x^{(i)}, \pi, p]$ . Show that

$$\eta(z_k^{(i)}) = \frac{\pi_k \prod_{d=1}^D (p_d^{(k)})^{x_d^{(i)}} (1 - p_d^{(k)})^{1-x_d^{(i)}}}{\sum_j \pi_j \prod_{d=1}^D (p_d^{(j)})^{x_d^{(i)}} (1 - p_d^{(j)})^{1-x_d^{(i)}}}$$

- v. Let  $\tilde{p}$ ,  $\tilde{\pi}$  be the new parameters that we would like to maximize.  $p$ ,  $\pi$  are from the previous iteration. Use this to derive the following final expression for the E-step in the EM algorithm:

$$\mathbb{E}[\log P(Z, \mathcal{D}|\tilde{p}, \tilde{\pi})|\mathcal{D}, p, \pi] = \sum_{i=1}^n \sum_{k=1}^K \eta(z_k^{(i)}) \left[ \log \tilde{\pi}_k + \sum_{d=1}^D (x_d^{(i)} \log \tilde{p}_d^{(k)} + (1 - x_d^{(i)}) \log(1 - \tilde{p}_d^{(k)})) \right]$$

(c) **Maximization step.** In the following, we will find  $\tilde{p}$  and  $\tilde{\pi}$  that maximize the above expression.

- i. Show that  $\tilde{p}$  that maximizes the E-step is:

$$\tilde{p}^{(k)} = \frac{\sum_{i=1}^n \eta(z_k^{(i)}) x^{(i)}}{N_k},$$

where  $N_k = \sum_{i=1}^n \eta(z_k^{(i)})$ .

- ii. Prove that the value of  $\tilde{\pi}$  that maximizes the E-step is:

$$\tilde{\pi}_k = \frac{N_k}{\sum_{k'} N_{k'}}.$$

Solution:

(a) i.

$$P(x | q) = \prod_{d=1}^D q_d^{x_d} (1 - q_d)^{1-x_d}$$

(a) ii.

$$P(x^{(i)} | p, \pi) = \sum_{k=1}^K \pi_k P(x^{(i)} | p^{(k)})$$

(a) iii.

$$\sum_{i \in D} \log \sum_{k=1}^K \pi_k P(x^{(i)} | p^{(k)})$$

(b) i.

$$P(z^{(i)} | \pi) = \prod_{k=1}^K \pi_k^{z_k^{(i)}}$$

(b) ii.

$$\prod_{k=1}^K P(x^{(i)} | p^{(k)})^{z_k^{(i)}}$$

(b) iii.

$$\begin{aligned} P(Z, \mathcal{D} | \pi, p) &= \prod_{i=1}^N P(x^{(i)}, z^{(i)} | \pi, p) = \prod_{i=1}^N P(x^{(i)} | z^{(i)}, \pi, p) P(z^{(i)} | \pi) \\ &= \prod_{i=1}^N \left[ \prod_{k=1}^K \left[ P(x^{(i)} | p^{(k)}) \right]^{z_k^{(i)}} \right] \left[ \prod_{k=1}^K \pi_k^{z_k^{(i)}} \right] \end{aligned}$$

(b) iv.

$$\mathbb{E}[z_k^{(i)} | x^{(i)}, \pi, p] = P[z_k^{(i)} = 1 | x^{(i)}, \pi, p]$$

$$\begin{aligned} &= \frac{P(x^{(i)} | z_k^{(i)}=1, \pi, p) P(z_k^{(i)}=1 | \pi, p)}{\sum_j P(x^{(i)} | z_j^{(i)}=1, \pi, p) P(z_j^{(i)}=1 | \pi, p)} \\ &= \frac{\pi_k \prod_{d=1}^D (p_d^{(k)})^{x_d^{(i)}} (1-p_d^{(k)})^{1-x_d^{(i)}}}{\sum_j \pi_j \prod_{d=1}^D (p_d^{(j)})^{x_d^{(i)}} (1-p_d^{(j)})^{1-x_d^{(i)}}} \end{aligned}$$

(b) v.

$$\begin{aligned} \log P(Z, \mathcal{D} | \tilde{\pi}, \tilde{p}) &= \sum_{i=1}^N \left[ \sum_{k=1}^K z_k^{(i)} \log \left[ P(x^{(i)} | \tilde{p}^{(k)}) \right] \right] + \left[ \sum_{k=1}^K z_k^{(i)} \log \tilde{\pi}_k \right] \\ &= \sum_{i=1}^N \sum_{k=1}^K z_k^{(i)} \left[ \log [P(x^{(i)} | \tilde{p}^{(k)})] + \log \tilde{\pi}_k \right] \\ &= \sum_{i=1}^N \sum_{k=1}^K z_k^{(i)} \left[ \log \tilde{\pi}_k + \log \prod_{d=1}^D (\tilde{p}_d^{(k)})^{x_d^{(i)}} (1-\tilde{p}_d^{(k)})^{1-x_d^{(i)}} \right] \\ &= \sum_{i=1}^N \sum_{k=1}^K z_k^{(i)} \left[ \log \tilde{\pi}_k + \sum_{d=1}^D \left[ x_d^{(i)} \log \tilde{p}_d^{(k)} + (1-x_d^{(i)}) \log (1-\tilde{p}_d^{(k)}) \right] \right] \end{aligned}$$

Now take expected value and use  $\eta(z_k^{(i)}) = \mathbb{E}[z_k^{(i)} | x^{(i)}, \pi, p]$  from iv. Then we get:

$$\sum_{i=1}^N \sum_{k=1}^K \eta(z_k^{(i)}) \left[ \log \tilde{\pi}_k + \sum_{d=1}^D (x_d^{(i)} \log \tilde{p}_d^{(k)} + (1-x_d^{(i)}) \log (1-\tilde{p}_d^{(k)})) \right]$$

(c) i.

$$\frac{\partial}{\partial p_d^{(k)}} \mathbb{E} \left[ \log P(Z, \mathcal{D} | \tilde{\pi}, \tilde{p}) \right] = \sum_{i=1}^N \eta(z_k^{(i)}) \left[ \frac{x_d^{(i)}}{\tilde{p}_d^{(k)}} - \frac{1-x_d^{(i)}}{1-\tilde{p}_d^{(k)}} \right]. \text{ Set this to 0.}$$

Then

$$\tilde{p}_d^{(k)} = \frac{\sum_{i=1}^N \eta(z_k^{(i)}) x_d^{(i)}}{N_k}. \text{ This implies that } \tilde{p}^{(k)} = \frac{\sum_{i=1}^N \eta(z_k^{(i)}) x^{(i)}}{N_k}$$

(c) ii.

Using Lagrange, since we need  $\sum_k \tilde{\pi}_k = 1$ ,

$$L(\tilde{\pi}, \alpha) = - \sum_{i=1}^N \sum_{k=1}^K \eta(z_k^{(i)}) \log \tilde{\pi}_k + \alpha \left( \sum_{k=1}^K \tilde{\pi}_k - 1 \right) \text{ where } \alpha \text{ is dual variable.}$$

$$\frac{\partial}{\partial \tilde{\pi}_k} L = - \sum_{i=1}^N \frac{\eta(z_k^{(i)})}{\tilde{\pi}_k} + \alpha. \text{ Set this to 0. Then } \tilde{\pi}_k = \frac{N_k}{\alpha}.$$

$$\text{So } L(\alpha) = - \sum_{i=1}^N \sum_{k=1}^K \eta(z_k^{(i)}) (\log N_k - \log \alpha) + \left( \sum_{k=1}^K N_k - \alpha \right)$$

$$\frac{\partial}{\partial \alpha} L = \frac{1}{\alpha} \sum_{i=1}^N \sum_{k=1}^K \eta(z_k^{(i)}) - 1. \text{ Set this to 0. Then } \alpha = \sum_{i=1}^N \sum_{k=1}^K \eta(z_k^{(i)}) = \sum_{k=1}^K N_k. \text{ Then } \tilde{\pi}_k = \frac{N_k}{\sum_{k'} N_{k'}}$$

## 4. K-Means 1


- (a) Mention if K-Means is a supervised or an un-supervised method and state the reason.
- (b) Assume that you are trying to cluster data points  $x_i$  for  $i \in \{1, 2, \dots, D\}$  into  $K$  clusters each with center  $\mu_k$  where  $k \in \{1, 2, \dots, K\}$ . The objective function for doing this clustering involves minimize the euclidean distance between the points and the cluster centers. It is given by

$$\min_{\mu} \min_r \sum_{i \in D} \sum_{k=1}^K \frac{1}{2} r_{ik} \|x_i - \mu_k\|_2^2.$$

How do you ensure hard assignment of one data point to one and only one cluster at a given time?

**Hint:** By hard assignment we mean that you are 100 % sure that a point either belongs or doesn't belong to a cluster.

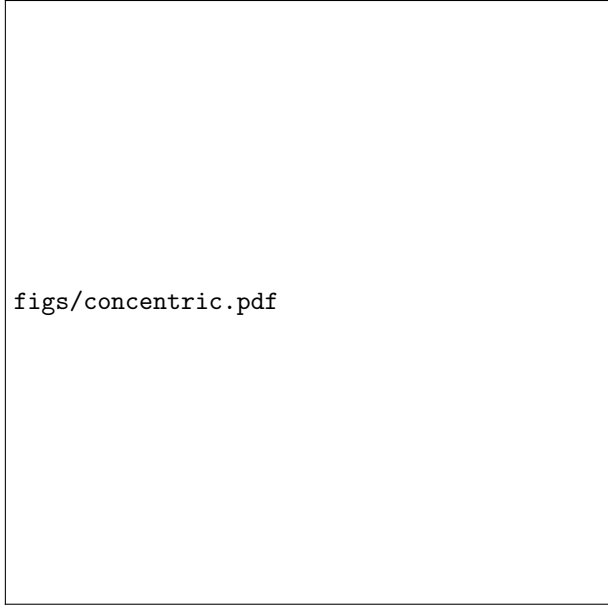
- (c) How does your answer to part b change if we want to obtain a soft assignment instead?
- Hint:** By soft assignment we mean that a point belongs to a cluster with some probability.
- (d) Looking at the following plot, what is the best choice for the number of clusters?



figs/cluster.pdf

- (e) Would K-Means be an efficient algorithm to cluster the following data? Explain your answer in a couple of lines.





figs/concentric.pdf

Solution:

(a)

un-supervised because data don't have labels.

(b)

$$r_{ik} \in \{0, 1\} \quad \forall i, k$$

$$\sum_{k=1}^K r_{ik} = 1 \quad \forall i$$

(c)

$$r_{ik} \in \{0, 1\} \rightarrow r_{ik} \in [0, 1] \quad \forall i, k$$

(d)

5.

(e)

No. K-mean always tries to construct a spherical shape around the centroid. So k-mean do poor job on complicated geometric shapes.

## 5. K-Means 2

We are given a dataset  $\mathcal{D} = \{(x)\}$  of 2d points  $x \in \mathbb{R}^2$  which we are interested in partitioning into  $K$  clusters, each having a cluster center  $\mu_k$  ( $k \in \{1, \dots, K\}$ ) via the  $k$ -Means algorithm. This algorithm optimizes the following cost function:

$$\min_{\mu_k, r} \sum_{x \in \mathcal{D}, k \in \{1, \dots, K\}} \frac{1}{2} r_{x,k} \|x - \mu_k\|_2^2 \quad \text{s.t.} \quad \begin{cases} r_{x,k} \in \{0, 1\} & \forall x \in \mathcal{D}, k \in \{1, \dots, K\} \\ \sum_{k \in \{1, \dots, K\}} r_{x,k} = 1 & \forall x \in \mathcal{D} \end{cases} \quad (2)$$

- What is the domain for  $\mu_k$ ?
- Given fixed cluster centers  $\mu_k \forall k \in \{1, \dots, K\}$ , what is the optimal  $r_{x,k}$  for the program in Eq. 2? Provide a reason?
- Given fixed  $r_{x,k} \forall x \in \mathcal{D}, k \in \{1, \dots, K\}$ , what are the optimal cluster centers  $\mu_k \forall k \in \{1, \dots, K\}$  for the program in Eq. 2?

**Hint:** Reason by first computing the derivative w.r.t  $\mu_k$ .

- Using Pseudo-code, sketch the algorithm which alternates the aforementioned two steps. Is this algorithm guaranteed to converge and why? Is this algorithm guaranteed to find the global optimum? What is the reason?

**Hint:** you can provide a counter-example to invalidate a statement.

- Please implement the aforementioned two steps. For the given dataset, after how many updates does the algorithm converge, what cost function value does it converge to and what are the obtained cluster centers? Visualize clusters at each step and attach the plots here. Please at least report numbers with one decimal point.

**Remark:** how we count updates: when computing a set of new centroids from initialization, we call this one update.

**Hint:** You may find `hw4_utils.vis_cluster` useful.

Solution:

- $\mathbb{R}^2$ , where data lie in.

- For each  $x^{(i)}$ , I look all  $\mu_k$  and I choose one  $k$  for which distance of  $x^{(i)}$  and  $\mu_k$  is the smallest. Then we can get the smallest cost. So:  
 $r_{x,k} = 1$  if  $k = \operatorname{argmin}_k \|x - \mu_k\|_2^2$ , otherwise  $r_{x,k} = 0$

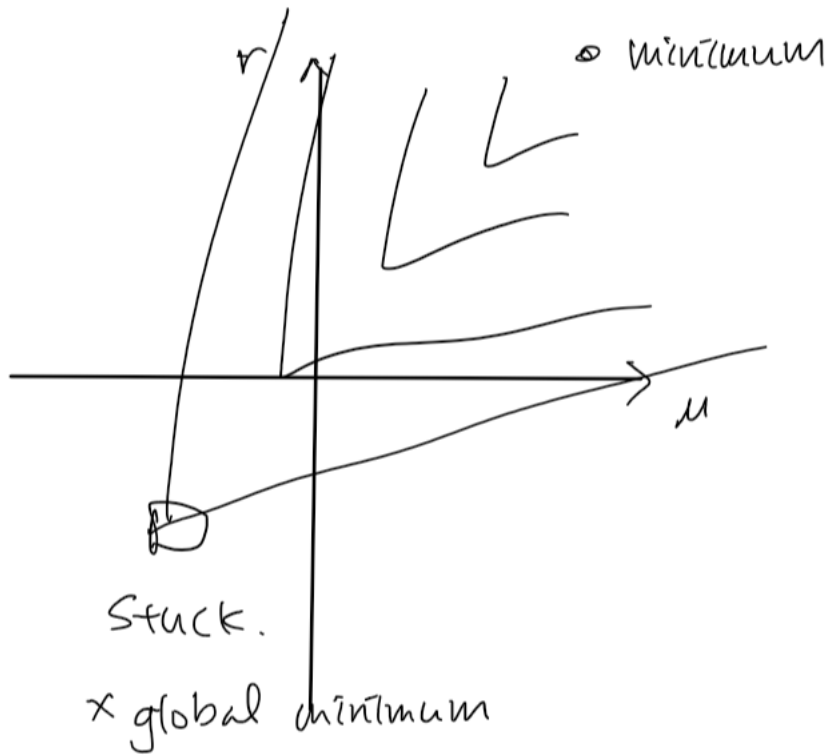
- $\nabla_{u_k} \text{cost function} = \sum_{x \in \mathcal{D}} r_{x,k} (x - u_k)$ . Set this to 0. Then  $u_k = \frac{\sum_{x \in \mathcal{D}} r_{x,k} \cdot x}{\sum_{x \in \mathcal{D}} r_{x,k}}$

- pseudo-code:  
 for each data point x:  
   find nearest centroid  
   assign a data point to that cluster

for each cluster:  
   new centroid = mean of all points assigned to the cluster

It is guaranteed to converge since cost function is lower bounded and in every step of optimization, cost is decreasing. However it's not guaranteed to find the global optimum. The following picture is a counter

example.



(e)

After two updates, the algorithm converges and the cost function value that it converges is 4.56.

The obtained cluster center is  $\begin{bmatrix} 1.9163 & -2.0952 \\ -1.9143 & 2.0540 \end{bmatrix}$

