

# Homework 4 Part 1

This is an individual assignment.

---

Solve all problems by hand. You may type your answers in markdown cells or [push](#) a PDF file with your handwritten answers.

---

## Problem 1 (10 points)

Consider the k-Nearest Neighbors (kNN) classifier for  $m$  classes.

Define the class prior probability  $P(C_j)$ , data likelihood  $p(\mathbf{x}|C_j)$ , and show that the KNN's posterior probability is

$$P(C_j|\mathbf{x}) = \frac{k_j}{k}$$

where  $k_j$  is the number of neighbors from class  $C_j$ . In addition, explain how you can generate new samples.

## HW 4 part 1.

Q1:

KNN classifier for m classes

(a)

Let  $N$  = total number of points.

$N_k$  = data points in class  $C_k$

$$\sum_k N_k = N$$

Let  $V$  be the volume from which  $k_j$  points ~~are drawn~~ from class  $C_j$

$$\therefore P(x|C_j) = \frac{k_j}{N_k V} \quad \text{and} \quad P(x) = \underbrace{\frac{K}{NV}}_{\text{points in the volume}}$$

$$\therefore P(C_j) = \frac{N_k}{N}$$

$$\therefore P(C_j|x) = \frac{P(x|C_j) P(C_j)}{P(x)} = \frac{\left(\frac{k_j}{N_k V}\right) \frac{N_k}{N}}{\left(\frac{K}{NV}\right)} = \boxed{\frac{\frac{k_j}{V}}{K}}$$

(b) New samples can be generated by considering the prior and likelihood of each class. The class ~~with high prior~~ will ~~be more likely~~ be more likely to generate a new sample in certain region.

## Problem 2 (15 points)

Consider the objective function for the Possibilistic C-Means (PCM)

$$J(\Theta, \mathbf{U}, \mathbf{H}) = \sum_{i=1}^N \sum_{k=1}^K u_{ik}^m d^2(\mathbf{x}_i, \theta_k) + \sum_{k=1}^K \eta_k \sum_{i=1}^N (1 - u_{ik})^m$$

where  $m > 1$  is the *fuzzifier*,  $u_{ik} \in [0, 1]$  is the membership value of sample  $\mathbf{x}_i$  in cluster centroid  $\theta_k$ ,  $d^2(\mathbf{x}_i, \theta_k)$  is the squared distance between sample  $\mathbf{x}_i$  and cluster centroid  $\theta_k$ , and  $\eta_k > 0$  determines the relative significance of the two terms.

1. (5 points) Explain the role of the second term in the objective function for PCM.

The role of second term is to basically ensure that the membership value  $u_{ik}$  is as close to 1 as possible. So the algorithm can mitigate the zero membership assignment problems of FCM.  $\eta_k$  determines the relative degree to which the second term is important compared with the first term.

1. (10 points) Observe that, for each vector  $\mathbf{x}_i$ , the  $u_{ik}$ 's,  $k = 1, \dots, K$ , are independent of each other, we can write  $J(\Theta, \mathbf{U}, \mathbf{H})$  as

$$J(\Theta, \mathbf{U}, \mathbf{H}) = \sum_{k=1}^K J_k$$

where

$$J_k = \sum_{i=1}^N u_{ik}^m d^2(\mathbf{x}_i, \theta_k) + \eta_k \sum_{i=1}^N (1 - u_{ik})^m$$

Each  $J_k$  corresponds to a different cluster and the minimization of  $J(\Theta, \mathbf{U}, \mathbf{H})$  with respect to the  $u_{ik}$ 's can be carried out separately for each  $J_k$ .

Solve for  $J_k$  as a function of  $u_{ik}$ ,  $\eta_k$  and  $m$ . For a fixed  $\eta_k$ , describe the placement of the centroids  $\theta_k$  in the minimization of  $J$ . Discuss the implications of this behavior when the number of selected clusters  $K$  is *larger* than the number  $K_n$  of natural clusters in  $\mathbf{X}$  (i.e.  $K > K_n$ ).

Q2

$$J_K = \sum_{i=1}^N u_{ik}^m d^2(x_i, \theta_k) + \eta_k \sum_{i=1}^N (1 - u_{ik})^m$$

$$\frac{\partial J_K}{\partial u_{ik}} = 0 \Leftrightarrow m u_{ik}^{m-1} (x_i - \theta_k)^2 + (-1) \eta_k m (1 - u_{ik})^{m-1}$$

$$\therefore m u_{ik}^{m-1} d^2(x_i, \theta_k) = \eta_k m (1 - u_{ik})^{m-1}$$

$$\therefore d^2(x_i, \theta_k) = \eta_k \left( \frac{1 - u_{ik}}{u_{ik}} \right)^{m-1}$$

$$\therefore J_K = \sum_{i=1}^N u_{ik}^m \eta_k \left( \frac{1 - u_{ik}}{u_{ik}} \right)^{m-1} + \eta_k \sum_{i=1}^N (1 - u_{ik})^m$$

$$\overline{u_{ik}} = \frac{1}{1 + \left( \frac{d^2(x_i, \theta_k)}{\eta_k} \right)^{m-1}} = \frac{1}{1 + \overline{\eta_k}}$$

If the number of selected cluster  $K$  are larger than the natural cluster  $k_n$ , then each data point will be considered as a separate cluster resulting in minimum distance with itself. Since the centroid of ~~the partition~~ a cluster will coincide with the ~~centroids~~

~~data point itself, the value of membership will be very close to 1 and the  $J_K$  will be at minimum.~~

### Problem 3 (10 points)

Suppose that you are interested in designing a clustering method that will cluster a dataset into spherical clusters with radius  $r_j$  centered at centroid  $\theta_j$ ,  $j = 1, \dots, K$ .

Conditions:

- The solution must find  $r_j, j = 1, \dots, K$  so that all  $K$  clusters have **minimum volume**.
- Allow a **minimal number of points** to lie outside the volume of its respective sphere/cluster.
- Introduce a penalty  $\xi_i > 0, i = 1, \dots, N$  for points lying outside sphere.

Design an objective function for this clustering algorithm satisfying the conditions above. Define all hyperparameters (if any), parameters, and solve for parameter solutions. Explain your reasoning.

$$Q3: J = \underbrace{\sum_{j=1}^K \frac{4}{3} \pi (\theta_j - r_j)^3}_{(1)} + \underbrace{\sum_{i=1}^N \xi_i \left[ (x_i - \theta_j)^2 - (r_j)^2 \right]}_{(2)}$$

where  $\theta_j$  = cluster centroid  
 $r_j$  = spherical cluster radius  
 $\xi_i$  = penalty for each data point that lies outside the sphere.  
 $x_i$  = data point "i"

The first term will ~~not~~ minimize the radius of each cluster and the second term will become positive as the distance of  $x_i$  from centroid  $\theta_j$  is greater than the radius  $r_j$ .

$$\therefore \frac{\partial J}{\partial r_j} = \frac{4}{3} \pi 3(\theta_j - r_j)^2 (-1) - \sum_{i=1}^N \xi_i 2(\theta_j - r_j) (-1)$$

assuming  
 $\xi_i$  constant for all data points

$$= -4\pi(\theta_j - r_j)^2 + 2(\theta_j - r_j)\xi_i = 0$$

$$\therefore 4\pi(\theta_j - r_j)^2 = 2(\theta_j - r_j)\xi_i$$

$$\therefore \boxed{r_j = \theta_j - \frac{1}{2}\xi_i}$$

$$\begin{aligned} \therefore \frac{\partial J}{\partial \theta_j} &= \frac{4}{\delta} \pi \sum_{i=1}^N (\theta_j - r_j)^2 + \sum_{i=1}^N \varepsilon_i \left[ (2)(x_i - \theta_j)(-1) - (-2)(1) \right] \\ &\text{assuming } \varepsilon_i \text{ constant over each data point} \\ &= 4\pi(\theta_j - r_j)^2 + \varepsilon_i [-2(x_i - \theta_j) - 2] = 0 \\ \therefore 4\pi(\theta_j - r_j)^2 &= \varepsilon_i [-(x_i - \theta_j) - 1] \\ \therefore 2(\theta_j^2 - 2\theta_j r_j + r_j^2) &= -\varepsilon_i x_i + \varepsilon_i \theta_j - \varepsilon_i \\ \therefore 2\theta_j^2 - 4\theta_j r_j - \varepsilon_i \theta_j &= -\varepsilon_i x_i - \varepsilon_i - 2r_j \\ \text{solving this will lead to } \theta_j \end{aligned}$$

## Problem 4 (10 points)

Let  $\mathbf{e}_i, i = 1, \dots, D$ , be any orthonormal basis in the  $D$ -dimensional space. Consider a  $D$ -dimensional random vector  $\mathbf{x}$ , which is approximated by

$$\hat{\mathbf{x}} = \sum_{i=1}^m y_i \mathbf{e}_i + \sum_{i=m}^D c_i \mathbf{e}_i$$

where  $c_i$  are non-random constants. Show that the minimum mean square error  $\mathbb{E} [\|\mathbf{x} - \hat{\mathbf{x}}\|^2]$  is achieved if

1.  $c_i = \mathbb{E}[y_i], i = m, \dots, D$ ,
2. the orthonormal basis consists of the eigenvectors of  $\Sigma_x$  (covariance of  $\mathbf{X}$ ); and
3.  $\mathbf{e}_i, i = m, \dots, D$ , correspond to the eigenvectors associated with the  $D - m$  smallest eigenvalues.

$$\underline{84.} \quad \hat{x} = \sum_{i=1}^M y_i e_i + \sum_{i=m}^D c_i e_i$$

$$E[\|x - \hat{x}\|^2] = E\left[\|x - \sum_{i=1}^m y_i e_i - \sum_{i=m}^D c_i e_i\|^2\right]$$

$e_i$  are orthogonal

$$\therefore \langle e_i, e_j \rangle = \delta_{ij} \leftarrow \text{Kronecker delta} = \begin{cases} 1, & i=j \\ 0, & \text{otherwise} \end{cases}$$

$$\therefore E[\|x - \hat{x}\|^2] = E[\langle x - \hat{x}, x - \hat{x} \rangle] \\ = E[\langle x, x \rangle] - 2 E[\langle x, \hat{x} \rangle] + E[\langle \hat{x}, \hat{x} \rangle]$$

$$\bullet E[\langle x, x \rangle] = \cancel{\text{det}} \text{ Trace}(\Sigma_x) \leftarrow \substack{\text{covariance} \\ \text{matrix}}.$$

$$\bullet E[\langle \mathbf{x}, \hat{\mathbf{x}} \rangle] = \sum_{i=1}^m E[\langle \mathbf{x}, y_i e_i \rangle] + \sum_{i=m}^D E[\langle \mathbf{x}, c_i e_i \rangle]$$

$$= \sum_{i=1}^m E[y_i] + \sum_{i=m}^D E[c_i]$$

$$\bullet E[\langle \hat{\mathbf{x}}, \hat{\mathbf{x}} \rangle] = E\left[\left\langle \sum_{i=1}^m y_i e_i + \sum_{i=m}^D c_i e_i, \sum_{j=1}^m y_j e_j + \sum_{j=m}^D c_j e_j \right\rangle\right]$$

$$= \sum_{i=1}^m E[y_i^2] + 2 \sum_{i=1}^m \sum_{j=m}^D E[y_i c_j] + \sum_{i=m}^D E[c_i^2]$$

$\therefore$  minimize MSE w.r.t.  $c_i$

$$\hookrightarrow \frac{\partial E[\|\mathbf{x} - \hat{\mathbf{x}}\|^2]}{\partial c_i} = 0$$

$$\therefore \frac{\partial E[\|\mathbf{x} - \hat{\mathbf{x}}\|^2]}{\partial c_i} = -2E[\langle \mathbf{x}, e_i \rangle] + 2c_i = 0$$

$$\therefore c_i = E[\langle \mathbf{x}, e_i \rangle] = E[y_i]$$

Optimal basis vectors = eigen vectors of covariance matrix of  $\Sigma_{\mathbf{x}}$

$\therefore d_1, d_2, \dots, d_D$  be the eigen vectors stored in decreasing order to minimize the sum of  $D-m$  smallest eigenvalues.

## On-Time (5 points)

Submit your assignment before the deadline.

# Submit Your Solution

Confirm that you've successfully completed the assignment.

Along with the Notebook, include a PDF of the notebook with your solutions.

`add` and `commit` the final version of your work, and `push` your code to your GitHub repository.

Submit the URL of your GitHub Repository as your assignment submission on Canvas.

---