ECE M146 Spring 2021 Final
Introduction to Machine Learning June 7, 2021
Instructor: Lara Dolecek Maximum score: 100 points

You have 3 hours to submit your work **directly on Gradescope under the Final submission link**.

Please read and carefully follow all the instructions.

Instructions

- The exam is accessible from 8 am PST on June 7th to 8 am PST on June 8th. Once you open the exam, you will have 3 hours to upload your work (therefore open the exam at least 3 hours before the closing time).
- This exam is open book, open notes. You are allowed to consult your own class notes (homework, discussion, lecture notes, textbook). You are not allowed to consult with each other or solicit external sources for help (e.g., an online forum).
- For each question, start a new sheet of paper. Therefore, the number of pages of your scan should be at least the number of questions. It is ok to write multiple parts of a question on one sheet. Properly erase or cross out any scratch work that is not part of the answer.
- Please submit your exam through the corresponding submission link on Gradescope.
- Make sure to include your **full name** and **UID** in your submitted file.
- Make sure to **show all your work**. Unjustified answers will be at a risk of losing points.
- Calculators are allowed for matrix inversion, entropy calculation and etc.
- **Policy on the Academic Integrity**
  "During this exam, you are **disallowed** to contact with a fellow student or with anyone outside the class who can offer a solution e.g., web forum."
  Please write the following statement on the first page of your answer sheet. You will **lose 10 points** if we can not find this statement.

I $\underline{\quad YourName \quad}$ with UID ____ have read and understood the policy on academic integrity.

1. (30 points) **True or False.**

   Circling the correct answer is worth +3 points, circling the incorrect answer is worth −1 points. Not circling either is worth 0 points.

   (a) The perceptron algorithm does not converge if the training samples are not linearly separable.

   TRUE FALSE

   Solution: True. The perceptron algorithm's update rule is to adjust the parameters if a point is misclassified, so if it is not possible to correctly classify all training data (e.g. linearly separable) then the algorithm will not reach a stable point.

   (b) Logistic Regression is a linear classifier.

   TRUE FALSE

   Solution: True, the decision boundary for a logistic classifier is linear.

   (c) We learned that the soft margin SVM have the primal problem:

$$\begin{aligned} \min_{\xi,w,b} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \\ s.t. \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i, \quad i = 1,\cdots,m \\ & \xi_i \geq 0, \quad i = 1,\cdots,m \end{aligned}$$

   and the dual problem:

$$\begin{aligned} \max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)}y^{(j)}\alpha_i\alpha_j\langle x^{(i)}, x^{(j)}\rangle \\ s.t. \quad & 0 \leq \alpha_i \leq C, i = 1,\cdots,m \\ & \sum_{i=1}^{m}\alpha_i y^{(i)} = 0. \end{aligned}$$

   Suppose we have solved the dual problem and have the optimal α. If we find that $0 < \alpha_i < C$, then $x^{(i)}$ is **inside** the margin, i.e., $y^{(i)}(w^T x^{(i)} + b) < 1$.

   TRUE FALSE

   Solution: False. Based on the KKT condition, those points are on the margin.

   (d) For $x_1, x_2 \in \mathbb{R}$, $K(x_1, x_2) = (1 + x_1 x_2)^2$ is a valid kernel.

   TRUE FALSE

   Solution: True.

$$K(x_1, x_2) = (1 + x_1 x_2)^2 = 1 + 2x_1 x_2 + x_1^2 x_2^2 = \phi(x_1)^T\phi(x_2),$$

where

$$\phi(x) = \begin{bmatrix} 1 \\ \sqrt{2}x \\ x^2 \end{bmatrix}.$$

(e) The Naive Bayes assumption assumes that the features x_i are independent.

TRUE FALSE

Solution: False. They are only conditionally independent given the class label. Conditional independence is not the same as independence.

(f) The decision boundary for the GDA model with equal class priors is linear.

TRUE FALSE

Solution: False. The decision boundary for the GDA model with equal class conditional covariance matrix is linear.

(g) For a given fixed set of data points and a fixed k, k-means always converges to the same clustering of the data.

TRUE FALSE

Solution: False, the initialization have impacts on the final results.

(h) In PCA, we want to **minimize** the variance of the projected data when we project the data onto a lower dimension.

TRUE FALSE

Solution: False.

(i) To do bagging, we sample the original dataset without replacement and train weak classifiers using the sampled dataset. This procedure can be done in parallel.

TRUE FALSE

Solution: False. Sample with replacement.

(j) In Adaboost, the weight α for each weak classifier is calculated based on the ratio of misclassified examples to the total number of examples.

TRUE FALSE

Solution: False. The error is weighted.

2. (15 pts) ***k*-Means Clustering**

The algorithm for k-means clustering are as follows: choose initial 'representatives' $z_1, ..., z_k$ for the k groups and repeat:

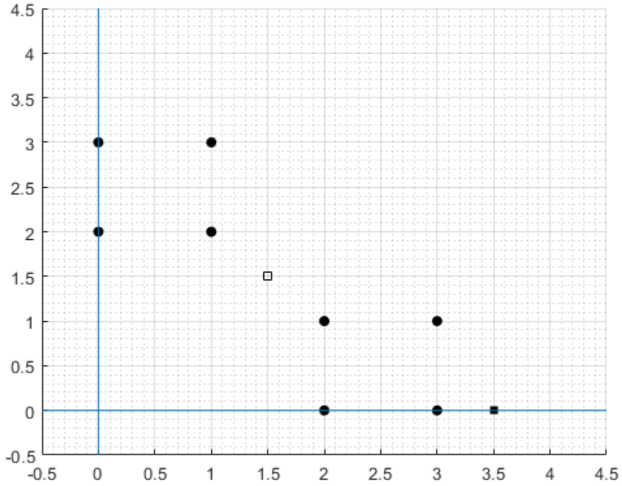
(a) assign each vector x_i to the nearest group representative z_j.

(b) set the representative z_j to the mean of the vectors assigned to it.

In this problem, you will perform k-means algorithm with $k = 2$ in the following data set: $x_1 = (0, 3)$, $x_2 = (1, 3)$, $x_3 = (0, 2)$, $x_4 = (1, 2)$, $x_5 = (2, 1)$, $x_6 = (3, 1)$, $x_7 = (2, 0)$, $x_8 = (3, 0)$, and initial 'representatives': $z_1 = (1.5, 1.5)$, $z_2 = (3.5, 0)$.

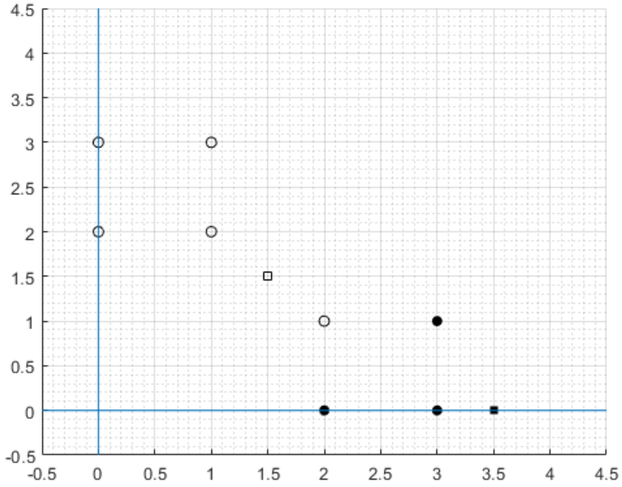
(a) In the following figure, plot the data x_i with filled circle '●', one initial 'representatives' with empty square '□', and the other representatives with filled squares '■'.

Solution:



(b) In the following figure, assign each data point x_i to the nearest group representative z_j. Plot one group of data points with empty circle '○' and its 'representatives' with empty box '□', and the other group of data points with filled circle '●' and its 'representatives' with filled box '■'. (In the figure, your 'representatives' should be the same as what you have in part (a)).

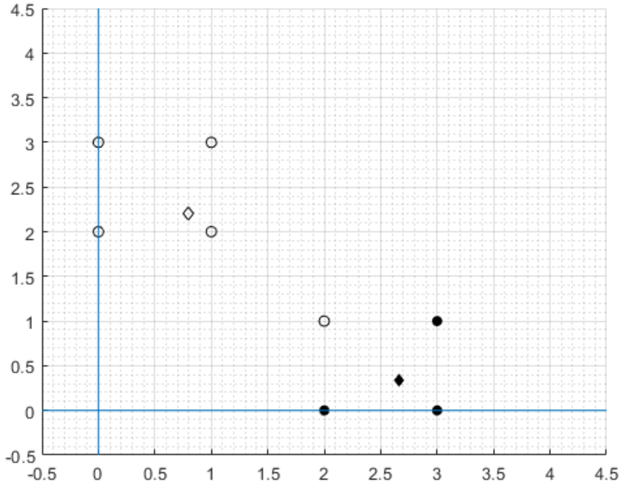
Solution:



(c) Update the representative z_j to the mean of the vectors assigned to it. What are the values of updated ‘representative’ z_1 and z_2? Copy you data points from part b, and plot one of the updated ‘representative’ with empty diamond ‘◇’ and the other with filled diamond ‘♦’.

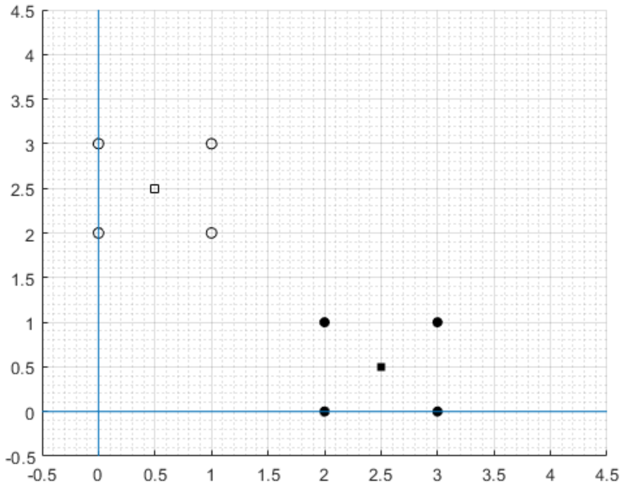
Solution:

$z_1 = (4/5, 11/5)$ and $z_2 = (8/3, 1/3)$



(d) Perform the algorithm untill it converges, i.e., the ‘representative’ doesn’t change anymore. Plot your final result in the following figure. Plot one group of data points with empty circle ‘○’ and its ‘representatives’ with empty box ‘□’, and the other group of data points with filled circle ‘●’ and its ‘representatives’ with filled box ‘■’.

Solution:


4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
-0.5
-0.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5

3. (15 pts) **Optimization**
Solve the following optimization problem. Justify your answer.

$$\begin{array}{ll} \min\limits_{x} & x^T A x \\ \text{subject to} & x^T x = 1, \end{array}$$

where $A = \begin{bmatrix} 5 & 3 \\ 3 & 5 \end{bmatrix}$ and x is a vector in $\mathbb{R}^2$. In you solution, you should justify the following:

(a) Is the problem a convex probelm, why?

(b) What is the Lagrangian?

(c) What is the optimal value of $x^T A x$?

(d) What is the optimal x?

Solution:
The Eigenvalue Decomposition of A is $\begin{bmatrix} 5 & 3 \\ 3 & 5 \end{bmatrix} = \begin{bmatrix} -\frac{2}{\sqrt{2}} & \frac{2}{\sqrt{2}} \\ \frac{2}{\sqrt{2}} & \frac{2}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix} \begin{bmatrix} -\frac{2}{\sqrt{2}} & \frac{2}{\sqrt{2}} \\ \frac{2}{\sqrt{2}} & \frac{2}{\sqrt{2}} \end{bmatrix}$ The problem is convex problem since A is p.s.d (all eigenvalues are greater than zero) and the constraint is a convex set.

We have the Lagrangian as:

$$L = x^T A x + \lambda_1 (x^T x - 1).$$

Setting the gradient respect to x to be 0, we get:

$$Ax = \lambda_1 x.$$

Therefore, the optimal x should be an eigenvector of A. Plugging this back to the objective function, we wish to minimize λ_1. Note that this part is different from PCA. The optimal value is therefore 2, and the optimal x is $[-\frac{2}{\sqrt{2}}, \frac{2}{\sqrt{2}}]^T$

4. (15 pts) **Weighted Least Square**

We have a set of data $x_n \in \mathbb{R}^M, y_n \in \mathbb{R}, n = 1, \cdots, N$. Suppose we want to find $w \in \mathbb{R}^M$ that minimize the following objective weighted least square loss function:

$$J(w) = \sum_{n=1}^{N} \alpha_n (w^T x_n - y_n)^2,$$

where $\alpha_n > 0$ are the weights for each data point.

(a) Write the loss function in matrix-vector form, i.e., as a squared 2-norm of some vector. Hint: you may use the following matrix: $A^{\frac{1}{2}}$ with $A^{\frac{1}{2}}_{nn} = \sqrt{\alpha_n}, n = 1, \cdots, N$ and all other elements being 0.

Solution: Define $X = \begin{bmatrix} x_1^T \\ \vdots \\ x_N^T \end{bmatrix}$,$y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$. Using the matrix $A^{\frac{1}{2}}$ in the hint, we have:

$$J(w) = \|A^{\frac{1}{2}}(Xw - y)\|^2.$$

(b) The normal equation for the w that minimizes $J(w)$ is of the form $Bw = c$. Find B and c in terms of the data and the weights. Hint: you may use the following matrix: A with $A_{nn} = \alpha_n, n = 1, \cdots, N$ and all other elements being 0.

Solution: Setting the gradient of $J(w)$ with respect to w equals zeros, we have:

$$X^T A X w = X^T A y.$$

We therefore find $B = X^T A X$ and $c = X^T A y$.

(c) Show that $J(w)$ has a global minimum. Hint: show that the Hessian matrix is positive-definite.

Solution:

The Hessian matrix is:

$$\nabla_w^2 J(w) = X^T A X.$$

We find that it is positive definite by showing:

$$u^T X^T A X u = v^T X v = \sum_n \alpha_n v_n^2 \geq 0, \forall u \in \mathbb{R}^M,$$

where $v = Xu$.

5. (15 pts) **Maximum Likelihood Estimation**

Let $x_1, x_2, \cdots, x_n$ be independent samples from the following distribution:

$$P(x|\theta) = \theta x^{-\theta-1} \quad \text{where} \quad \theta > 1, x \geq 1.$$

(a) Find the maximum likelihood estimate of θ that maximizes $P(x_1, x_2, \cdots, x_n|\theta)$.

Solution:

$$P(x_1, x_2, \cdots, x_n|\theta) = \prod_{i=1}^{n} \theta x_i^{-\theta-1} = \theta^n \prod_{i=1}^{n} x_i^{-\theta-1}.$$

$$\ln P(x_1, x_2, \cdots, x_n|\theta) = n \ln \theta - (\theta + 1) \sum_{i=1}^{n} \ln x_i.$$

$$\frac{\partial \ln P(x_1, x_2, \cdots, x_n|\theta)}{\partial \theta} = \frac{n}{\theta} - \sum_{i=1}^{n} \ln x_i = 0.$$

$$\hat{\theta}_{MLE} = \frac{n}{\sum_{i=1}^{n} \ln x_i}.$$

Since $\theta > 1$, any $\hat{\theta}_{MLE} \leq 1$ has a zero probability of generating any data, so our best estimate of θ when $\hat{\theta}_{MLE} \leq 1$ is $\hat{\theta}_{MLE} = 1$. Therefore, the final answer is $\max(1, \frac{n}{\sum_{i=1}^{n} \ln x_i})$. However, we will still accept $\hat{\theta}_{MLE} = \frac{n}{\sum_{i=1}^{n} \ln x_i}$ as the correct answer.

(b) Show that the estimator you get in (a) indeed maximizes $P(x_1, x_2, \cdots, x_n|\theta)$ instead of minimizing it. I.e, show that the second derivative of $\ln P(x_1, x_2, \cdots, x_n|\theta)$ with respect to θ is non-positive.

Solution:

$$\frac{\partial^2 \ln P(x_1, x_2, \cdots, x_n|\theta)}{\partial \theta^2} = -\frac{n}{\theta^2} \quad \text{which is non-positive.}$$

(c) Estimate θ if you have data $x_1 = 3, x_2 = 5, x_3 = 2$, and $x_4 = 10$. You may leave you answer with natural log(s).

Solution:

We get the estimate by directly plugging the data in the estimator we get in (a).

$$\hat{\theta}_{MLE} = \frac{4}{\ln 3 + \ln 5 + \ln 2 + \ln 10}.$$

6. (10 pts) **Expectation maximization**

You learned that the log likelihood function for the Gaussian mixture model is of this form:

$$J = \ln P(X;\pi,\mu,\Sigma) = \sum_{n=1}^{N} \ln\left\{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n;\mu_k,\Sigma_k)\right\}.$$

Here, π_k is the prior probability of the latent variable; μ_k and Σ_k are the mean and covariance matrix for the k-th Gaussian component. Suppose the data points x_n are D-dimensional.

Suppose we want to maximize J with respect to μ_l. Show that the μ_l that maximize J is of the form:

$$\hat{\mu}_l = \frac{\sum_{n=1}^{N} \gamma_{nl} x_n}{\sum_{n=1}^{N} \gamma_{nl}},$$

where

$$\gamma_{nk} = \frac{\pi_k \mathcal{N}(x_n;\mu_k,\Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n;\mu_j,\Sigma_j)}.$$

You may assume that all γ_{nk} are known for this step. You **must show all necessary steps (e.g., application of chain rules) to get full credit** for this question. Matrix calculus results can be used without proof.

Solution:

$$\frac{\partial J}{\partial \mu_l} = \sum_{n=1}^{N} \frac{\pi_l}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n;\mu_k,\Sigma_k)} \frac{\partial}{\partial \mu_l}\left[\frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma_l|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x_n-\mu_l)^T\Sigma_l^{-1}(x_n-\mu_l)\right)\right] \quad (1)$$

$$= \sum_{n=1}^{N} \frac{\frac{\pi_l}{(2\pi)^{\frac{D}{2}}|\Sigma_l|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x_n-\mu_l)^T\Sigma_l^{-1}(x_n-\mu_l)\right)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n;\mu_k,\Sigma_k)} \frac{\partial}{\partial \mu_l}\left(-\frac{1}{2}(x_n-\mu_l)^T\Sigma_l^{-1}(x_n-\mu_l)\right) \quad (2)$$

$$= \sum_{n=1}^{N} \gamma_{nl} \times \left(-\frac{1}{2}\right) \times (-2) \times \Sigma^{-1}(x_n-\mu_l). \quad (3)$$

Setting the gradient equals 0, we get:

$$\mu_l \sum_{n=1}^{N} \gamma_{nl} = \sum_{n=1}^{N} \gamma_{nl} x_n,$$

and hence the result.