ECE M146 Homework 6
Introduction to Machine Learning Monday, May 10, 2021
Instructor: Lara Dolecek Due: Monday, May 24, 2021
TA: Zehui (Alex) Chen chen1046@ucla.edu

Please upload your homework to Gradescope by May 24, 4:00 pm.
Please submit a single PDF directly on Gradescope
You may type your homework or scan your handwritten version. Make sure all the work is discernible.

1. Consider the classification problem for two classes, C_0 and C_1. In the generative approach, we model the class-conditional distribution $P(x|C_0)$ and $P(x|C_1)$, as well as the class priors $P(C_0)$ and $P(C_1)$.The posterior probability for class C_0 can be written as

$$P(C_0|x) = \frac{P(x|C_0)P(C_0)}{P(x|C_0)P(C_0) + P(x|C_1)P(C_1)}.$$

   (a) Show that $P(C_0|x) = \sigma(a)$ where $\sigma(a)$ is the *sigmoid* function defined by

$$\sigma(a) = \frac{1}{1+\exp(-a)}.$$

   Find a in terms of $P(x|C_0)$, $P(x|C_1)$, $P(C_0)$ and $P(C_1)$.

   (b) In the LDA model, we have the class conditional distribution as follows

$$P(x|C_0) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)\right),$$

$$P(x|C_1) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right).$$

   Suppose we are able to find the maximum likelihood estimation of $\mu_0, \mu_1, \Sigma, P(C_0)$, and $P(C_1)$. Show that $a = w^Tx + b$ for some w and b. Find w and b in terms of $\mu_0, \mu_1, \Sigma, P(C_0)$, and $P(C_1)$. This shows that the decision boundary is linear.

   (c) In (b), we modeled the class conditional distribution with same covariance matrix Σ. Now let us consider two classes that have difference covariance matrices, as follows

$$P(x|C_0) = \frac{1}{(2\pi)^{n/2}|\Sigma_0|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0)\right),$$

$$P(x|C_1) = \frac{1}{(2\pi)^{n/2}|\Sigma_1|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)\right).$$

   Suppose we are able to find the maximum likelihood estimation of $\mu_0, \mu_1, \Sigma_0, \Sigma_1, P(C_0)$, and $P(C_1)$. Show that $a = x^TAx + w^Tx + b$ for some A, w and b. Find A, w and b in terms of $\mu_0, \mu_1, \Sigma_0, \Sigma_1, P(C_0)$, and $P(C_1)$. This shows that the decision boundary is quadratic.

2. We are given a training set $\{(x^{(i)}, y^{(i)}); i = \{1, \cdots, m\}\}$, where $x^{(i)} \in R^n$ and $y^{(i)} \in \{0, 1\}$. We consider the Gaussian Discriminant Analysis (GDA) model, which models $P(x|y)$ using multivariate Gaussian. Writing out the model, we have:

$$P(y = 1) = \phi = 1 - P(y = 0)$$

$$P(x|y = 0) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_0)^T \Sigma^{-1}(x - \mu_0)\right)$$

$$P(x|y = 1) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_1)^T \Sigma^{-1}(x - \mu_1)\right)$$

The log-likelihood of the data is given by:

$$L(\phi, \mu_0, \mu_1, \Sigma) = \ln P(x^{(1)}, \cdots, x^{(m)}, y^{(1)}, \cdots, y^{(m)}) = \ln \prod_{i=1}^{m} P(x^{(i)}|y^{(i)}) P(y^{(i)}).$$

In this exercise, we want to maximize $L(\phi, \mu_0, \mu_1, \Sigma)$ with respect to ϕ, μ_0. The maximization over Σ is left for discussion.

(a) Write down the explicit expression for $P(x^{(1)}, \cdots, x^{(m)}, y^{(1)}, \cdots, y^{(m)})$ and $L(\phi, \mu_0, \mu_1, \Sigma)$.

(b) Find the maximum likelihood estimate for ϕ. How do you know such ϕ is the "best" but not the "worst"? Hint: Show that the second derivative of $L(\phi, \mu_0, \mu_1, \Sigma)$ with respect to ϕ is negative.

(c) Find the maximum likelihood estimate for μ_0. How do you know such μ_0 is the "best" but not the "worst"? Hint: Show that the Hessian Matrix of $L(\phi, \mu_0, \mu_1, \Sigma)$ with respect to μ_0 is negative definite. You may use the following: if A is positive definite, then A^{-1} is also positive definite. Also B is negative definite if $-B$ is positive definite.

3. In the previous homework, we use the GDA model to classify admitted and rejected UCLA applicants using one feature only. Now we have learned about the GDA model for the vector case. Let's use the vector GDA model to perform classification based on both features GPA and GRE. The dataset that we are going to use is *UCLA_EE_grad_2030.csv*.

   (a) In the GDA model, we assume the class label follows a Bernoulli distribution and we model the class conditional distributions as multivariate Gaussians with the same covariance matrix (Σ) and different means (μ_0 and μ_1). Find the maximum likelihood estimate of the parameters $P(y = 0)$ (parameter for the Bernoulli distribution), μ_0, μ_1 and Σ given this data set.

   (b) Using your ML estimate of model parameters, find the decision boundary parameterized by $w^T x + b = 0$. Report w, b and plot the decision boundary on the plot in (a).

   (c) Visualize your results by plotting the contour of the two distributions $P(x, y = 0)$ and $P(x, y = 1)$. For consistency, set 'LevelList' ('level' for python) to logspace(-2,-0.6,7). Does your decision boundary pass through the points where the two distributions have equal probabilities ?

4. Suppose we have a data set $\{x_1, \cdots, x_N\}$ and out goal is to partition the data set in to K clusters with μ_k representing the center of the k-th cluster. Recall that in K-means clustering we are attempting to minimize an objective function defined as follows:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|x_n - \mu_k\|_2^2,$$

where $r_{nk} \in \{0, 1\}$ and $r_{nk} = 1$ only if x_n is assigned to cluster k.

(a) What is the minimum value of the objective function when $K = n$ (the number of clusters equals to the number of samples)?

(b) Adding a regularization term, the objective function now becomes:

$$J = \sum_{k=1}^{K} \left[\lambda \|\mu_k\|_2^2 + \sum_{n=1}^{N} r_{nk} \|x_n - \mu_k\|_2^2 \right].$$

Consider the optimization of μ_k with all r_{nk} known. Find the optimal μ_k for

$$\text{argmin}_{\mu_k} \lambda \|\mu_k\|_2^2 + \sum_{n=1}^{N} r_{nk} \|x_n - \mu_k\|_2^2.$$

Discuss your answer. How would the regularization affect each step of the K-means clustering algorithm?

5. We have unlabeled data $x_n \in \mathbf{R}^M, n = 1, \cdots, N$. Suppose we want to use L_1 distance instead of L_2 distance to cluster the data into K clusters. The objective function we are minimizing becomes:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|x_n - \mu_k\|_1,$$

where $\|z\|_1 = \sum_{i=1}^{M} |z_i|$ for $z \in \mathbf{R}^M$. The parameters $r_{nk} \in \{0, 1\}$ and $r_{nk} = 1$ only if x_n is assigned to cluster k.
In the maximization step, with r_{nk} fixed, define $C_k = \{n | r_{nk} = 1\}$ for the k-th cluster. Then we need to find μ_k that minimizes the following function:

$$f(\mu_k) = \sum_{n \in C_k} \|x_n - \mu_k\|_1. \tag{1}$$

Define x_{ni} to be the i-th element in x_n and μ_{ki} to be the i-th element in μ_k. We can expand (1) as following:

$$f(\mu_k) = \sum_{n \in C_k} \|x_n - \mu_k\|_1 = \sum_{n \in C_k} \sum_{i=1}^{M} |x_{ni} - \mu_{ki}| = \sum_{i=1}^{M} \sum_{n \in C_k} |x_{ni} - \mu_{ki}|.$$

The above expansion shows that we can optimize for each element in μ_k separately.

(a) Consider first the problem of finding $\bar{y}^*$ that minimizes $f(\bar{y}) = \sum_{j=1}^{N_k} |y_j - \bar{y}|$ for $y_j \in \mathbf{R}$. Because $f(\bar{y})$ is not differentiable everywhere, we need the notion of *subgradient*. We say $g \in \mathbf{R}$ is a subgradient of f at $x \in \mathbf{dom} f$ of for all $z \in \mathbf{dom} f$:

$$f(z) \geq f(x) + g(z - x).$$

The subgradient of f at point x where f is differentiable equals to the derivative of f at x. A function f is called subdifferentiable at x if there exists at least one subgradient at x. The set of subgradients of f at point x is called *subdifferential* of f at x and is denoted as $\partial f(x)$. Show that the subdifferential $\partial f(x)$ of $f(x) = |x|$ is:

$$\partial f(x) = \begin{cases} -1, x < 0 \\ [-1, 1], x = 0 \\ 1, x > 0 \end{cases}$$

Hint: Use the definition of subgradient to find the subdifferential for the case $x = 0$.

(b) Suppose we have $y_1 = 1, y_2 = 2, y_3 = 3, y_4 = 4$ and $y_5 = 5$ and a function $f(\bar{y}) = \sum_{j=1}^{5} |y_j - \bar{y}|$. Evaluate $f(\bar{y})$ at $\bar{y} = 2, 3$ and 3.5. Using the results in (a), find the subdifferential $\partial f(\bar{y})$ for the $\bar{y} = 2, 3$ and 3.5.

(c) Suppose we have $y_1 = 1, y_2 = 2, y_3 = 3, y_4 = 4, y_5 = 5$ and $y_6 = 6$ and a function $f(\bar{y}) = \sum_{j=1}^{6} |y_j - \bar{y}|$. Evaluate $f(\bar{y})$ at $\bar{y} = 3, 3.5, 4$ and 5. Using the results in (a), find the subdifferential $\partial f(\bar{y})$ for the $\bar{y} = 3, 3.5, 4$ and 5.

(d) Assume we have a dataset contains $y_i, i = 1, \cdots, N$. All y_i(s) are distinct and the data is ordered, i.e., $y_1 < y_2 < \cdots < y_{N-1} < y_N$. Based on your results in (b) and (c), can you make an educated guess on what is the optimal $\bar{y}$ that minimizes $f(\bar{y}) = \sum_{j=1}^{N} |y_j - \bar{y}|$?
Use the following theorem:
A point x^* is a minimizer of a convex function f if and only if f is subdifferentiable at x^* and $0 \in \partial f(x^*)$, i.e., $g = 0$ is a subgradient of f at x^*.
Show that the median of $\{y_1, \cdots, y_N\}$ minimizes $f(\bar{y}) = \sum_{j=1}^{N} |y_j - \bar{y}|$.

(e) Write a two-step algorithm similar to the K-means algorithm that minimizes J. Comment on the advantage of this algorithm compared to the K-means algorithm.

6. Answer the following questions regarding positive (semi-)definite matrix. A symmetric real matrix M is said to be positive definite if the scalar $z^T M z$ is positive for every non-zero column vector z.

   (a) Consider the matrix

   $$A = \begin{bmatrix} 9 & 6 \\ 6 & a \end{bmatrix}.$$

   What should a satisfy so that the matrix A is positive definite?

   (b) Suppose we know matrix B is positive definite. Show that B^{-1} is also positive definite. Hint: use the definition and the fact that every positive definite matrix is non-singular (invertible).

   (c) Show that the data covariance matrix S in PCA is positive semi-definite.

7. One application of the K-means algorithm is image segmentation and image compression. The goal of image segmentation is to partition an image into regions that have relatively similar visual appearance. Each pixel in an image can be viewed as a point in a 3-dimensional space which contains the intensity of the 3 color red, green and blue. K-means algorithm can be used to cluster the points in the 3-dimensional space in to K clusters therefore achieve segmentation. After segmentation, compression is achieved by replacing each pixel with the {R,G,B} triplet given by μ_k, the center the cluster to which it is assigned.
In this exercise, you will implement the K-means algorithm to segment and compress the image *UCLA_Bruin.jpg*. Note: for submission, you may turn in the required images in black and white.

   (a) **Visualization.** The picture of the famous Bruin bear contains 300×400 pixels. Read the image into MATLAB (or python) and show the image.

   (b) **K-means Algorithm with $K = 4$.** Implement the K-means algorithm using all of the following specifications:

   - Partition the pixels into $K = 4$ clusters.
   - To allow for a deterministic result, initialize the cluster centers using the *furthest-first* heuristic on page 180 of *A Course in Machine Learning*. The heuristic is sketched below:
     - Pick the first pixel (pixel on the upper left corner) of the image, whose {R,G,B} values are $[147, 200, 250]$, as the center for the first cluster, i.e., $\mu_1 = [147, 200, 250]$.
     - For $k = 2, \cdots, K$: find the example n^* that is as far as possible from all previously selected means. Namely, $n^* = \underset{n}{\operatorname{argmax}} \min_{k'<k} \|x_n - \mu_{k'}\|^2$. Set $\mu_k = x_{n^*}$.

     Report your initial centers.
   - Run the K-means algorithm for 10 iterations. An iteration consists the following two steps:
     - Step 1, assign each example to the cluster whose center is the closest.
     - Step 2, re-estimate the center of each cluster.

   Calculate the K-means objective function:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|x_n - \mu_k\|_2^2,$$

   at the end of each iteration. The parameters $r_{nk} \in \{0, 1\}$ and $r_{nk} = 1$ only if x_n is assigned to cluster k. For this image, we have $x_n \in \mathbf{R}^3, i = 1, \cdots, N$, $N = 120000$. Generate a plot showing Js v.s. iterations. Comment on the convergence of the K-means algorithm.

   (c) **Compression with $K = 4, 8$ and 16.** For $K = 4, 8$ and 16, compress the *UCLA_Bruin* image using your K-means algorithm. For compression, replace the

{R,G,B} values of each pixel with the center of the cluster to which it belongs. Use the same specifications in (b) and report the value of the objective function after your last iteration. Show your compressed image using *imshow*. Comment on how K affects the quality of the compressed image.

(d) **Compression Ratio.** In the original image, each of the 300×400 pixels comprises {R,G,B} values each of which is stored with 8 bits of precision, i.e., $0 - 255$.How many bits do you need to store the original image?
Now you have compressed your image using the K-means algorithm. For each pixel, you store only the index of cluster to which it is assigned. You also need to store the value of the K centers with 8 bits of precision per color. How many bits do you need to store the compressed image with $K = 4, 8$ and 16? What are the compression ratios?