CS 274A Final

Yushang Lai

TOTAL POINTS

83 / 90

QUESTION 1

1 Written Pledge 0 / 0

✓ - 0 pts Correct

QUESTION 2

Problem 1 30 pts

2.1 Part 1 10 / 10

✓ - 0 pts Correct

2.2 Part 2 9 / 9

✓ - 0 pts Correct

2.3 Part 3 11 / 11

✓ - 0 pts Correct

QUESTION 3

Problem 2 30 pts

3.1 Part 1 10 / 10

✓ - 0 pts Correct

3.2 Part 2 10 / 10

✓ - 0 pts Correct

3.3 Part 3 10 / 10

✓ - 0 pts Correct

QUESTION 4

Problem 3 30 pts

4.1 Part 1 8 / 12

✓ - 2 pts Part (a): Membership weights w_{ik} are not considered to be parameters. Number of parameters is 2Kd + K-1

✓ - 2 pts Part (c): incorrect complexity for E-step. It should be Kd for inverting K diagonal matrices plus NKd for computing the memberships with a diagonal covariance matrix

4.2 Part 2 10 / 10

✓ - 0 pts Correct

4.3 Part 3 5 / 8

✓ - 3 pts Interesting idea, but weights based on subsets of features could potentially lead to non-convergence of EM. A more straightforward way would be to just compute E and M updates based on a subset B of the data samples at each iteration.

QUESTION 5

5 Blank Page 0 / 0

✓ - 0 pts Correct

Take-Home Final Exam, CS 274A, Winter 2020

Time allowed: 10am Wednesday March 18th to noon Thursday March 19th

- Please write your name and sign at the bottom of the page.
- This page must also be accompanied by the following (truthful) pledge written/typed out below and signed by you: "On my honor, I have neither given nor received any unauthorized aid on this examination.
- Write your exam on these pages, not on other pages. If you run out of page space for a problem, you can add pages to your solution and clearly indicate which problem and part at the top (e.g., "Problem 2.2"). But try if you can to fit your solution on these pages.
- Submit your final solution to Gradescope by noon Thursday March 19th. No need to submit any scratch work.
- You can consult any class-related material: lecture notes, homeworks, materials linked to from the class Web page.
- You are not allowed to use any other resources (e.g., other material on the Web, other textbooks, papers, etc).
- You are not allowed to communicate with anyone (other than the instructor) about the exam until noon on Thursday March 19th.
- If you don't understand a question please read it carefully and at least twice. If you believe there is a typo or error in the statement of the question please contact the Professor with a **private message** on Piazza. Please do not ask for hints on how to solve the problem.

WRITTEN PLEDGE: On my honor, I have neither given nor recieved any unauthorized aid on this examination.

SIGNATURE: Yushang Lai

PRINTED NAME: YUSHANG LAI

Problem 1: (30 points)

1. Consider a classification problem with d-dimensional input vector $\underline{x}$ and a class variable y taking K possible values. Given only this information define both (i) a lower bound and (ii) an upper bound on the Bayes error rate P_e^*, for this problem. No need to derive your answer, just state the bounds. The bounds should be as tight as possible given the information.

   lower bound: 0

   upper bound: (K-1)/K

2. For the problem in part 1, lets say we added a new variable to the input space (call it x_{d+1}) and we now consider the Bayes error rate of this new problem, relative to the Bayes error rate of the original problem where we only had d variables. Answer TRUE or FALSE to each of the following:

   (a) The Bayes error rate decreases can

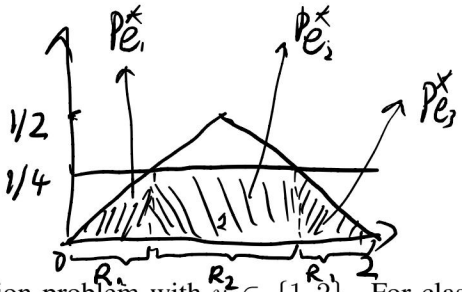
   True

   (b) The Bayes error rate remains the same

   True

   (c) The Bayes error rate increases

   False



3. Consider the following 1-dimensional binary classification problem with $y \in \{1, 2\}$. For class 1, $p(x|y = 1)$ is a uniform density $U(0, 2)$. For class 2, $p(x|y = 2) = x$ for $x \in [0, 1]$, $p(x|y = 2) = 2 - x$ for $x \in [1, 2]$, and $p(x|y = 2) = 0$ otherwise. We also know that $p(y = 1) = p(y = 2)$.

$p(x) = \sum_y p(x, y) = p(x, y=1) + p(x, y=2)$

   (a) Derive from first principles what the decision regions are for this classifier (to minimize 0-1 loss)

   (b) Compute the Bayes error rate for this problem (and show clearly how you obtained your answer).

(a) $1 = p(y=1) + p(y=2) = 2p(y=1) \Rightarrow p(y=1) = p(y=2) = 1/2$

$1 = \int_0^2 p(x|y=1)\,dx = 2 \cdot p(x|y=1) \Rightarrow p(x|y=1) = 1/2$

recall R_k = decision region in input x where $g_k(x) > g_j(x)\ \forall j, j \neq k$ $\quad g_k = p(x|y=k)$

$g_1(x) = p(y=1|x)\,p(x) = p(x|y=1)\,p(y=1) = 1/2 \cdot 1/2 = 1/4$

$g_2(x) = p(y=2|x)\,p(x) = p(x|y=2)\,p(y=2) = \begin{cases} x/2, & x \in (0,1) \\ 1 - x/2, & x \in (1,2) \end{cases}$

on $x \in (0,1)$, let $g_1(x) = g_2(x) \Rightarrow x_1 = 1/2$, or $x_1 \in (1,2)$, let $g_1(x) = g_2(x) \Rightarrow x_2 = 3/2$

thus $R_1 = (0, 1/2) \cup (3/2, 2)$ $\quad R_2 = (1/2, 3/2)$

(b) $p(x) = g_1(x) + g_2(x) = \begin{cases} 1/4 + x/2, & x \in (0,1) \\ 5/4 - x/2, & x \in (1,2) \end{cases}$

$P_e^* = \sum_k \int_{R_k} (1 - p(y=k|x))\,p(x)\,dx = P_{e_1}^* + P_{e_2}^* + P_{e_3}^*$

$= \int_0^{1/2} (1 - p(y=1|x))\,p(x)\,dx + \int_{1/2}^{1} (1 - p(y=2|x))\,p(x)\,dx$

$+ \int_1^{3/2} (1 - p(y=2|x))\,p(x)\,dx + \int_{3/2}^{2} (1 - p(y=1|x))\,p(x)\,dx$

$= \int_0^{1/2} (1/4 + x/2 - 1/4)\,dx + \int_{1/2}^{1} (1/4 + x/2 - x/2)\,dx$

$+ \int_1^{3/2} (5/4 - x/2 - 1 + x/2)\,dx + \int_{3/2}^{2} (5/4 - x/2 - 1/4)\,dx$

$= \frac{x^2}{4}\Big|_0^{1/2} + \frac{x}{4}\Big|_{1/2}^{1} + \frac{x}{4}\Big|_1^{3/2} + x - \frac{x^2}{4}\Big|_{3/2}^{2}$

$= 1/16 + 1/4 - 1/8 + 3/8 - 1/4 + 2 - 1 - 3/2 + 9/16$

$= (1/16 + 1/8) \cdot 2$

$= 3/8$

(Additional space for Problem 1.3, if needed)

Problem 2: (30 points)

1. Say we have a classification problem with K classes and where we have training data $D = \{(\underline{x}_i, y_i)\}, 1 \leq i \leq N$. The $y_i \in \{1, \ldots, K\}$ are class labels with unknown class probabilities $\pi_1, \ldots, \pi_K$, where $\pi_k = p(y = k)$ and $\sum_{k=1}^{K} \pi_k = 1$. Each $\underline{x}_i$ represents a document. The data $\underline{x}_i$ for the ith document, $1 \leq i \leq N$, is an M-dimensional vector of counts, $\underline{x}_i = (r_{i1}, \ldots, r_{iM})$, where $r_{im} \in \{0, 1, 2, \ldots\}$ is the number of times that word m occurs in document i and $m = 1, \ldots, M$. The $\underline{x}_i$ are modeled as multinomials, conditioned on the class label. Each multinomial has its own parameters $\theta_k = \{\theta_{k1}, \ldots, \theta_{kM}\}$, $1 \leq k \leq K$, and $\sum_{m=1}^{M} \theta_{km} = 1$. Let $Dir(\alpha_1, \ldots, \alpha_M)$ be a Dirichlet prior for each parameter vector θ_k. Documents i are assumed to be conditionally independent of each other and words for each document are assumed to be generated by the corresponding class multinomial.

   Derive the maximum a posteriori estimate for the θ_{km} parameters. Show all steps clearly in your derivation.

$$P(\underline{x}_i \mid \theta_{k_i}) = \prod_{m=1}^{M} \theta_{k_i}^{r_{im}}, \quad P(y_i = k_i \mid \pi) = \pi_{k_i}, \quad P(\theta_{k_i} \mid \alpha) = \prod_{m=1}^{M} \theta_{k_i}^{\alpha_m - 1}$$

$$P(\theta \mid D, \pi, \alpha) \propto P(D \mid \pi, \theta, \alpha)\, P(\theta \mid \pi, \alpha) \propto \prod_{i=1}^{N} P(\underline{x}_i \mid \theta_{k_i})\, P(y_i = k_i \mid \pi)\, P(\theta_{k_i} \mid \alpha)$$

$$= \prod_{i=1}^{N} \prod_{m=1}^{M} \theta_{k_i m}^{r_{im} + \alpha_m - 1} \cdot \pi_{k_i}$$

$$\ell(\theta) = \log\left(P(\theta \mid D, \pi, \alpha)\right) = \sum_{i=1}^{N} \log \prod_{m=1}^{M} \theta_{k_i m}^{r_{im} + \alpha_m - 1} \cdot \pi_{k_i} \qquad \text{Note } \sum_{m=1}^{M} \theta_{km} = 1$$

$$= \sum_{i=1}^{N} \sum_{m=1}^{M} \log \theta_{k_i m}^{r_{im} + \alpha_m - 1} \cdot \pi_{k_i}$$

$$= \sum_{i=1}^{N} \sum_{m=1}^{M} (r_{im} + \alpha_m - 1) \log \theta_{k_i m} + \log \pi_{k_i}$$

add

$$= \left(\sum_{i=1}^{N} \sum_{m=1}^{M} (r_{im} + \alpha_m - 1) \log \theta_{k_i m} + \log \pi_{k_i} \right) - \lambda \left(\sum_{m=1}^{M} \theta_{km} - 1 \right)$$

$$\frac{\partial \ell(\theta)}{\partial \theta_{k_i m}} = \sum_{i \in I_k} (r_{im} + \alpha_m - 1) \frac{1}{\theta_{k_i m}} - \lambda \overset{\text{want}}{=\!=} 0$$

let I_k be the subset of $\{1, \ldots, N\}$ s.t. $k_i = k$ for $\forall\, i \in I_k$

thus we have $\sum_{i \in I_k} (r_{im} + \alpha_m - 1) \cdot \frac{1}{\theta_{km}} = \lambda$

$$\theta_{km} = \frac{\sum_{i \in I_k} (r_{im} + \alpha_m - 1)}{\lambda}$$

(Additional space for Problem 2.1, if needed)

since $\sum_{m=1}^{M} \theta_{km} = 1 \Rightarrow \sum_{m=1}^{M} \sum_{i \in I_k} (r_{im} + \alpha_m - 1) = \lambda$

thus MAP $\hat{\theta}_{km} = \frac{\sum_{i \in I_k} (r_{im} + \alpha_m - 1)}{\sum_{m=1}^{M} \sum_{i \in I_k} (r_{im} + \alpha_m - 1)}$

where I_k is a subset of $\{1, 2, \cdots, N\}$ s.t. $k_i = k$ for $\forall i \in I_k$

2. Consider the problem in 2.1, where instead of learning the parameters we assume they are known, i.e., we know θ_k and π_k for each class. Consider a new document $\underline{x}$ with counts $(r_1, \ldots, r_M)$. Define an optimal discriminant function $g_k(\underline{x})$ to classify $\underline{x}$ (assuming 0-1 loss), $k = 1, \ldots, K$. Clearly define the discriminant function as a function of the θ_k's, the π_k's, and $\underline{x}$. Can the optimal discriminant function be defined as a linear discriminant? (answer YES or NO). If YES, show how it can be written in this form.

since θ_k, π_k are known

define $g_k(\underline{x}) = \log\left[p(\underline{x} \mid y=k)\, p(y=k)\right]$

$$= \log\left(C \prod_{m=1}^{M} \theta_{km}^{r_m} \cdot (\pi_k)\right)$$

$$= \sum_{m=1}^{M} r_m \log \theta_{km} + \log \pi_k$$

$$= (\log \underline{\theta}_k)^t \underline{x} + \log \pi_k$$

Yes, it can be define as linear as

$$g_k(\underline{x}) = \alpha_k \underline{x} + \beta_k \qquad \text{where} \quad \alpha_k = (\log \underline{\theta}_k)^t$$

$$\beta_k = \log \pi_k$$

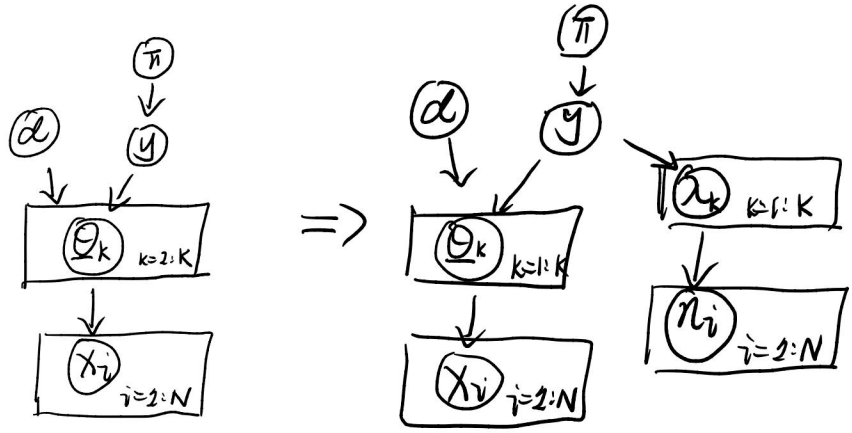
3. Say we know that the length of a document, the total number of words, $n_i = \sum_{m=1}^{M} r_{im}$, might also be related to the class label for a document and we want to extend the multinomial model for documents (from problem 2.1, 2.2) to a model where document length can be included when predicting the class label for a new document x.

   (a) Describe one approach (multiple approaches are possible) that shows how the model from parts 2.1/2.2 could be extended to define a generative model that includes document lengths n_i.

   (b) Show clearly how maximum likelihood could be used to learn any parameters needed for modeling document length in your extended model (and clearly state any assumptions in your derivation). (Ignore the Dirichlet priors in this part of the problem).

   (c) Write the optimal discriminant for your extended model in an additive form.

(a)



where n_i is a passion distribution with λ_k

$p(\lambda) = \text{Normal}(a y_i, b)$

$a = 20,\ b = 10$

(b)

$$p(\theta, \lambda \mid D) \propto p(D \mid \theta, \lambda)\, p(\theta, \lambda)$$

$$= \prod_{i=1}^{N} p(x_i \mid \theta_{k_i})\, p(y_i = k_i \mid \pi)\, p(n_i \mid \lambda_{k_i}) \cdot p(\theta)\, p(\lambda)$$

$$= \prod_{i=1}^{N} \prod_{m=1}^{M} \theta_{k_i m}^{r_{im}} \cdot \pi_{k_i} \frac{\lambda_{k_i}^{n_i} e^{-\lambda_{k_i}}}{n_i!} \cdot \text{parts not depend on } \theta_{km}, \lambda_k$$

$$\ell(\theta, \lambda) = \log(p(\theta, \lambda \mid D))$$

$$= \left[\sum_{i=1}^{N} \sum_{m=1}^{M} \left(r_{im} \log \theta_{k_i m} + \log \pi_{k_i} + n_i \log \lambda_{k_i} - \lambda_{k_i} - \log(n_i!) \right) \right] + \mu \left(\sum_{k=1}^{m} \theta_{km} - 1 \right)$$

$$\frac{\partial \ell(\theta, \lambda)}{\partial \theta_{km}} = \left(\sum_{i \in I_k} r_{im} \frac{1}{\theta_{km}} \right) - \mu \overset{\text{want}}{=\!=} 0$$

$$\theta_{km} = \frac{\sum_{i \in I_k} r_{im}}{\mu}$$

since $1 = \sum_{m=1}^{M} \theta_{km} = \frac{\sum_{m=1}^{M} \sum_{i \in I_k} r_{im}}{\mu}$ thus $\mu = \sum_{m=1}^{M} \sum_{i \in I_k} r_{im}$

where I_k be the subset of $\{1 \dots N\}$ s.t. $k_i = k$ for $\forall i \in I_k$

(Additional space for Problem 2.3, if needed)

MAP thus $\hat{\theta}_{km} = \frac{\sum_{i \in I_k} r_{im}}{\sum_{m=1}^{M} \sum_{i \in I_k} r_{im}}$

$$\frac{\partial \ell(\theta, \lambda)}{\partial \lambda_k} = \sum_{i \in I_k} \sum_{m=1}^{M} \left(\frac{n_i}{\lambda_k} - 1 \right) \overset{\text{want}}{=} 0$$

thus $\frac{\sum_{i \in I_k} n_i}{\lambda_k} - |I_k| = 0$

MAP $\hat{\lambda}_k = \frac{\sum_{i \in I_k} n_i}{|I_k|}$ where I_k is the subset of $\{1 \dots N\}$ s.t $k_i = k$ for $\forall i \in I_k$

$|I_k|$ is the number of elems in I_k

(c) $g_k(x, n) = \log p(x, n | y = k) p(y = k)$

$= \log p(x | y = k) \; p(n | y = k) p(y = k)$

$= (\log \underline{\theta}_k)^t \cdot \underline{x} + (\log \lambda_k)^t \cdot n - \log(n!) + \log \pi_k$

$\hat{\underline{\theta}}_k = (\hat{\theta}_{k1}, \hat{\theta}_{k2}, \dots, \hat{\theta}_{km})$

$\hat{\lambda}_k = \frac{\sum_{i \in I_k} n_i}{|I_k|}$

where λ_k is a vector of λ_i when $k_i = k$

optimal discriminant function

$$\hat{g}_{x,n} = (\log \hat{\underline{\theta}}_k)^t \cdot \underline{x} + (\log \hat{\lambda}_k) \cdot n - \log(n!) + \log \pi_k$$

Problem 3: (30 points)

Consider a d-dimensional variable $\underline{x}$ where we have conditionally independent observations $D = \{\underline{x}_1, \ldots, \underline{x}_N\}$. We wish to use a mixture model and EM to cluster the data into K Gaussian components, each with mean μ_k and covariance matrix Σ_k, $1 \leq k \leq K$. Let w_{ik} be the membership weights as discussed in lectures, $1 \leq i \leq N, 1 \leq k \leq K$.

1. Say we are told that each Gaussian component should be modeled using a naive Bayes model.
   (a) How many total parameters are in this mixture model?
   (b) Define the M-step for updating the covariance matrix terms for this specific mixture model.
   (c) What is the time complexity of a single EM iteration for this model?

(a) w_{ik} N by K, $\underline{\mu}_k$ d by 1, Σ_k d by 1 since NB thus diagnal d_k K by 1

total parameters, $Nk + kd + kd + k = Nk + 2kd + k$

(b) for $i = 1:K$

① $N_k = \sum_{i=1}^{N} w_{ik}$ effective data pts assigned to component

② $\hat{\alpha}_k = N_k / N$

③ $\hat{\mu}_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik}$

④ $\hat{\Sigma}_k = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} \begin{pmatrix} (x_{i2} - \mu_{k1})^2 & & 0 \\ & \ddots & \\ 0 & & (x_{id} - \mu_{kd})^2 \end{pmatrix}$

where $\underline{x}_i = \begin{pmatrix} x_{i1} \\ \vdots \\ x_{id} \end{pmatrix}$ $\underline{\mu}_k = \begin{pmatrix} \mu_{k1} \\ \vdots \\ \mu_{kd} \end{pmatrix}$

(c) E step

for $i = 1:N$; for $k = 1:K$ $w_{ik} = E[z_{ik} | \underline{x}_i, \theta] = P(z_{ik} = 1 | \underline{x}_i, \theta)$

$= \frac{P(\underline{x}_i | z_{ik} = 1, \theta_k) \, P(z_{ik} | \theta) \quad ②}{\sum_{k=1}^{K} (\text{above part})}$ ← $P(x|\theta)$ $K \cdot d$

$(x-\mu)^t \Sigma_k^{-1} (x-\mu)$ ↑

E ① $N \times K \times d \times d$ ② $K \times d$ inverse matrix

M ① kN ② K ③ k ④ kNd

total for one iteration $O(NKd^2)$

In parts 3.2 and 3.3 of this problem below assume that we are now using unconstrained covariance matrices Σ_k (i.e., no naive Bayes assumption).

2. Say that m of the N datapoints (where $m < N$) are labeled, i.e., for these m datapoints (only) the data consists of pairs $(\underline{x}_i, y_i)$, with $y_i \in \{1, \ldots, K\}$ indicating the true label. Describe below how you would do maximum likelihood estimation of the μ_k's and Σ_k's for such a model. A high-level sketch of the key idea for your proposed algorithm is fine.

I will use EM Algorithm (Gaussian Mixture) to do maximum likelihood estimation of μ_k's and Σ_k's.

let $\theta_k = \{\mu_k, \Sigma_k\}$ $\quad \theta = \{\theta_1, \cdots, \theta_K\}$

let $D_x = (\underline{x}_i)$ $\quad D_z = \{\underline{z}_i\}$ where $\underline{z}_i = (z_{i1}, \cdots, z_{ik})$ $\quad z_{ik}=1$ if k true, $z_{ik}=0$ otherwise

since we have m data with labels, we fix those lines w_{ik} in there true columns and not update its membership weights in each iteration's E step.

then based on Jensen's Inequality

we maximize the likelihood of μ_k's and Σ_k's by ascent of the lower bound of $\ell(\theta) = \sum_{i=1}^{N} \log p(x_i; \theta)$

Since time complexity linear depend on N, hard to speed up by variable N.

3. Consider now a standard Gaussian mixture problem with N unlabeled data points. In class we discussed stochastic gradient for speeding up gradient-based training. Clearly describe some method (e.g, step-by-step, pseudocode, etc) that could use an approach similar to stochastic gradient to speed up the EM algorithm for Gaussian mixtures. Your method need not have a guarantee that the likelihood is non-decreasing at each iteration. If the mini-batch size is B, how much speed-up per iteration would your proposed method gain?

one iteration

inspect that time complexity highly depends on the dimension of X which is d

E step

random order feature d, split into subset of size B, with the last one may less than B

WLOG say G subsets, $\{N_1, \cdots, N_G\}$

$W_{ik} = 1$ for $\forall i \in N,\ k \in K$

for $i = 1:N$.
  for $g = 1:G$
    for $k = 1, \cdots, K$

X_{ik}^g mean that we only use the features in subset $N_g \subset \{1, \cdots, d\}$

$$W_{ik}^* = \frac{P(X_{ik}^g \mid Z_{ik}=1, \theta_k)\, P(Z_{ik}=1 \mid \theta_k)}{\sum_k \text{terms on top}}$$

the M step

for $g = 1:G$
  for $i = 1:K$

$$N_{gk} = \sum_{i=1}^{N_g} W_{ik}$$

$$\hat{\alpha}_k = N_{gk} / N_g$$

$$\hat{\mu}_k = \frac{1}{N_{gk}} \sum_{i=1}^{N_g} W_{ik}$$

$$\Sigma_k = \frac{1}{N_{gk}} \sum_{i=1}^{N} W_{ik} (X_i^g - \hat{\mu}_k)(X_i^g - \hat{\mu}_k)^t$$

speed up per iteration: previous E step $O(KNd^3 + NKd^2)$ M step $O(NKd^2)$

new E step $O(KB^3 \cdot \frac{d}{B} + NKB^2 \cdot \frac{d}{B}) = O(KB^2 d + NKBd)$ M step $O(NKB^2 \frac{d}{B}) = O(NKBd)$

based on E step performance may speed up $(\frac{d}{B})$ times