CS 412 Data Mining (22Sp): Extra Credit Assignment

Xu Ke (UIN: 675776713 NetID: kex5)

April. 30, 2022

1 Problem 1. Bayes Classifier

1.1 For a binary classification problem, prove that Bayes classifier is the optimal classifier compared to all other classifiers.

[Hint: Given a data point (x,y) and an arbitrary classifier f(), conditional on X = x, the prediction made by the classifier f(X = x) is independent to the ground truth y]

To prove that Bayes classifier is the optimal classifier compared to all other classifiers for binary classification problem, we just need to show that Bayes classifier has the smallest classification error rate (also called Bayes error rate) compared to all other classifiers, i.e. $error_{true}(f_{Bayes}(x)) \leq error_{true}(g(x)), \forall g(x)$, shown as follows.

Let's denote domain X, label $Y = \{0,1\}$ with data point (x,y), and define Bayes Classifier f_D over any probability distribution D over $X \times Y$ to be:

$$f_D(x) = \left\{ \begin{array}{ll} 1 & ifP[y=1|x] \geq \frac{1}{2} \\ 0 & otherwise \end{array} \right. \tag{1}$$

Then, we denote $L_D(g) = D\{(x,y)|g(x) \neq y\}$ as the true error of classifier g, which is:

$$L_D(g) = E_{x,y \in D} P[g(x) \neq y] = E_{x,y \in D} \left\{ \begin{array}{ll} P[y \neq 0|x] & ifg(x) = 0 \\ P[y \neq 1|x] & ifg(x) = 1 \end{array} \right. \tag{2}$$

Due to binary classification, it is clear that the optimal classifier should minimize the loss function:

$$\phi(x) = \left\{ \begin{array}{ll} P[y=1|x] & ifg(x) = 0 \\ 1 - P[y=1|x] & ifg(x) = 1 \end{array} \right. \tag{3}$$

To minimize the loss function above, we need have $P[y=1|x] < 1 - P[y=1|x]$ with $g(x) = 0$ and $P[y=1|x] > 1 - P[y=1|x]$ with $g(x) = 1$ pairwisely, ignoring the equal condition without loss of the generality. Thus, we have:

$$g(x) = \left\{ \begin{array}{ll} 0 & ifP[y=1|x] < \frac{1}{2} \\ 1 & ifP[y=1|x] > \frac{1}{2} \end{array} \right. \tag{4}$$

We can easily find that $g(x)$ is identical to the Bayes Classifier $f_D(x)$ shown above, ignoring the equal condition without loss of the generality to be consistent as before. Thus, we can conclude that Bayes Classifier does minimize the error and thus is the optimal classifier, hence proved.

[Acknowledgement:some of my ideas during the proof process are adopted from online resources and book Introduction to Statistical Pattern Recognition, ISBN 0122698517]

Table 1: Training data for Naïve Bayes classifier

ID	Workclass	Education	Salary
1	Private	Bachelors	$\leq 50K$
2	State-Gov	Bachelors	$\leq 50K$
3	Private	Masters	$> 50K$
4	Local-Gov	Bachelors	$> 50K$
5	Private	High-School	$\leq 50K$
6	State-Gov	Masters	$> 50K$
7	Private	High-School	$> 50K$
8	Local-Gov	Masters	$\leq 50K$

1.2 From the given training data in Table 1, we train a Naive Bayes classifier to predict whether an employee's annual salary is more than 50K or not. Each row refers to an employee with 2 categorical features (i.e., Workclass, Education) and one class label (i.e., Salary). Let x be the features of an employee (e.g., x = (Private, Bachelors) for the employee with ID = 1) and y be the salary of the corresponding employee. Please answer the following questions.

1.2.1 How many independent parameters are required for training this Naive Bayes classifier from the given training data? Please list them all.

$\boxed{14}$ independent parameters are required for training the Naive Bayes classifier from the given training data. Let's denote salary > 50 as 1 and ≤ 50 as 0 for simplicity, then we will have:

$$\boxed{\left\{\begin{array}{c}
P(x_1 = Private|y = 1) \\
P(x_1 = Private|y = 0) \\
P(x_1 = State - Gov|y = 1) \\
P(x_1 = State - Gov|y = 0) \\
P(x_1 = Local - Gov|y = 1) \\
P(x_1 = Local - Gov|y = 0) \\
\\
P(x_2 = Bachelors|y = 1) \\
P(x_2 = Bachelors|y = 0) \\
P(x_2 = Masters|y = 1) \\
P(x_2 = Masters|y = 0) \\
P(x_2 = High - School|y = 1) \\
P(x_2 = High - School|y = 0) \\
\\
P(y = 0) \\
P(y = 1)
\end{array}\right.} \tag{5}$$

1.2.2 Please estimate the values of these parameters based on the observations in Table 1.

Let's denote salary > 50 as 1 and ≤ 50 as 0 again to be consistent with (a), then we will have:

$$\boxed{\left\{\begin{array}{c}
P(x_1 = Private|y = 1) = \frac{2}{4} = 0.5 \\
P(x_1 = Private|y = 0) = \frac{2}{4} = 0.5 \\
P(x_1 = State - Gov|y = 1) = \frac{1}{4} = 0.25 \\
P(x_1 = State - Gov|y = 0) = \frac{1}{4} = 0.25 \\
P(x_1 = Local - Gov|y = 1) = \frac{1}{4} = 0.25 \\
P(x_1 = Local - Gov|y = 0) = \frac{1}{4} = 0.25 \\
\\
P(x_2 = Bachelors|y = 1) = \frac{1}{4} = 0.25 \\
P(x_2 = Bachelors|y = 0) = \frac{2}{4} = 0.5 \\
P(x_2 = Masters|y = 1) = \frac{2}{4} = 0.5 \\
P(x_2 = Masters|y = 0) = \frac{1}{4} = 0.25 \\
P(x_2 = High - School|y = 1) = \frac{1}{4} = 0.25 \\
P(x_2 = High - School|y = 0) = \frac{1}{4} = 0.25 \\
\\
P(y = 0) = \frac{4}{8} = 0.5 \\
P(y = 1) = \frac{4}{8} = 0.5
\end{array}\right.} \tag{6}$$

1.2.3 Given a new data point with features x = (State-Gov, Bachelors), would the Naive Bayes classifier predict the label as $y \leq 50K$ or $y \geq 50K$ for this data point? Why?

$$P(y = 1|x = (State - Gov, Bachelors)) \propto P(State - Gov|y = 1) \times P(Bachelors|y = 1) \times P(y = 1) = 0.03125 \tag{7}$$

$$P(y=0|x=(State-Gov,Bachelors)) \propto P(State-Gov|y=0)\times P(Bachelors|y=0)\times P(y=0) = 0.0625 \quad (8)$$

Since P(y=0—x=(State-Gov,Bachelors)) should be either 0 or 1, i.e. $P(y=0|x=(State-Gov,Bachelors)) + P(y=1|x=(State-Gov,Bachelors)) = 1$, thus after normalization, we have:

$$\boxed{\begin{cases} P(y=1|x=(State-Gov,Bachelors)) = \frac{1}{3} \\ P(y=0|x=(State-Gov,Bachelors)) = \frac{2}{3} \end{cases}} \quad (9)$$

Thus, it will be labeled as 0, which means $\boxed{y \leq 50K}$.

2 Problem 4. Gaussian Mixture Model

Given a 1-dimensional data set: {-67,-48,6,8,14,16,23,24}, consider using a Gaussian Mixture Model with 2 components (k = 2) to fit your data.

2.1 How many independent parameters are there in this GMM? Please justify your answer.

$\boxed{6}$ independent parameters are there in this GMM, namely w_1, w_2 for priors, μ_1, μ_2 for means , and σ_1, σ_2 for standard deviations. Since the general formula is $p(x|\theta) = \sum_{k=1}^{2} w_k P(x|\mu_k, \sigma_k)$ and in this given case we are given k=2 for 2 components.

2.2 What will your parameters be after 1 iteration of E-M (Expectation Maximization) algorithm? Show your major calculations in both the E-step and the M-step. Only giving out the final results will NOT grant you any score. Feel free to initialize your parameters any way you prefer. Please show the initialized parameters you choose and the intermediate steps for obtaining the full points.

For this E-M algorithm, I initialize my parameters as:

$$\boxed{\begin{cases} w_1^1 = 0.3 \\ w_2^1 = 0.7 \\ \mu_1^1 = -30 \\ \mu_2^1 = 20 \\ \sigma_1^1 = 5 \\ \sigma_2^1 = 2 \end{cases}} \quad (10)$$

2.2.1 E-Step:

The general formula is $w_{ij} = \frac{w_j^t P(x_i|\mu_j^t, \sigma_j^t)}{\sum_k w_k^t P(x_i|\mu_k^t, \sigma_k^t)}$, thus we can get the following, respectively:

$$w_{11}^2 = \frac{w_1^1 P(x_1|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_1|\mu_1^1, \sigma_1^t) + w_2^1 P(x_1|\mu_2^1, \sigma_2^1)} = \boxed{1.0} \quad (11)$$

$$w_{21}^2 = \frac{w_1^1 P(x_2|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_2|\mu_1^1, \sigma_1^t) + w_2^1 P(x_2|\mu_2^1, \sigma_2^1)} = \boxed{1.0} \quad (12)$$

$$w_{31}^2 = \frac{w_1^1 P(x_3|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_3|\mu_1^1, \sigma_1^t) + w_2^1 P(x_3|\mu_2^1, \sigma_2^1)} = \boxed{0.0398} \quad (13)$$

$$w_{41}^2 = \frac{w_1^1 P(x_4|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_4|\mu_1^1, \sigma_1^t) + w_2^1 P(x_4|\mu_2^1, \sigma_2^1)} = \boxed{3.2282 \times 10^{-6}} \quad (14)$$

$$w_{51}^2 = \frac{w_1^1 P(x_5|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_5|\mu_1^1, \sigma_1^t) + w_2^1 P(x_5|\mu_2^1, \sigma_2^1)} = \boxed{2.3579 \times 10^{-16}} \quad (15)$$

$$w_{61}^2 = \frac{w_1^1 P(x_6|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_6|\mu_1^1, \sigma_1^t) + w_2^1 P(x_6|\mu_2^1, \sigma_2^1)} = \boxed{5.2885 \times 10^{-19}} \tag{16}$$

$$w_{71}^2 = \frac{w_1^1 P(x_7|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_7|\mu_1^1, \sigma_1^t) + w_2^1 P(x_7|\mu_2^1, \sigma_2^1)} = \boxed{2.1086 \times 10^{-25}} \tag{17}$$

$$w_{81}^2 = \frac{w_1^1 P(x_8|\mu_1^1, \sigma_1^1)}{w_1^1 P(x_8|\mu_1^1, \sigma_1^t) + w_2^1 P(x_8|\mu_2^1, \sigma_2^1)} = \boxed{5.9514 \times 10^{-26}} \tag{18}$$

$$w_{12}^2 = \frac{w_2^1 P(x_1|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_1|\mu_1^1, \sigma_1^t) + w_2^1 P(x_1|\mu_2^1, \sigma_2^1)} = \boxed{0.0} \tag{19}$$

$$w_{22}^2 = \frac{w_2^1 P(x_2|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_2|\mu_1^1, \sigma_1^t) + w_2^1 P(x_2|\mu_2^1, \sigma_2^1)} = \boxed{3.6136 \times 10^{-248}} \tag{20}$$

$$w_{32}^2 = \frac{w_2^1 P(x_3|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_3|\mu_1^1, \sigma_1^t) + w_2^1 P(x_3|\mu_2^1, \sigma_2^1)} = \boxed{0.9602} \tag{21}$$

$$w_{42}^2 = \frac{w_2^1 P(x_4|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_4|\mu_1^1, \sigma_1^t) + w_2^1 P(x_4|\mu_2^1, \sigma_2^1)} = \boxed{0.9999967718192685} \tag{22}$$

$$w_{52}^2 = \frac{w_2^1 P(x_5|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_5|\mu_1^1, \sigma_1^t) + w_2^1 P(x_5|\mu_2^1, \sigma_2^1)} = \boxed{0.9999999999999997} \tag{23}$$

$$w_{62}^2 = \frac{w_2^1 P(x_6|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_6|\mu_1^1, \sigma_1^t) + w_2^1 P(x_6|\mu_2^1, \sigma_2^1)} = \boxed{1.0} \tag{24}$$

$$w_{72}^2 = \frac{w_2^1 P(x_7|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_7|\mu_1^1, \sigma_1^t) + w_2^1 P(x_7|\mu_2^1, \sigma_2^1)} = \boxed{1.0} \tag{25}$$

$$w_{82}^2 = \frac{w_2^1 P(x_8|\mu_2^1, \sigma_2^1)}{w_2^1 P(x_8|\mu_1^1, \sigma_1^t) + w_2^1 P(x_8|\mu_2^1, \sigma_2^1)} = \boxed{1.0} \tag{26}$$

2.2.2 M-Step:

The general formula is $\mu_j = \frac{\sum_i w_{ij}^{t+1} x_i}{\sum_i w_{ij}^{t+1}}$, thus we can get the following, respectively:

$$\mu_1^2 = \frac{\sum_{i=1}^8 w_{i1}^2 x_i}{\sum_{i=1}^8 w_{i1}^2} = \boxed{-56.2613} \tag{27}$$

$$\mu_2^2 = \frac{\sum_{i=1}^8 w_{i2}^2 x_i}{\sum_{i=1}^8 w_{i2}^2} = \boxed{15.2279} \tag{28}$$

$$w_1^2 = \frac{\sum_{i=1}^8 w_{i1}^2 x_i}{8} = \boxed{0.2550} \tag{29}$$

$$w_2^2 = \frac{\sum_{i=1}^8 w_{i2}^2 x_i}{8} = \boxed{0.7450} \tag{30}$$

$$\sigma_1^2 = \sqrt{\frac{\sum_{i=1}^8 w_{i1}^2 (x_i - \mu_1)^2}{\sum_{i=1}^8 w_{i1}^2}} = \boxed{12.8691} \tag{31}$$

$$\sigma_2^2 = \sqrt{\frac{\sum_{i=1}^8 w_{i2}^2 (x_i - \mu_2)^2}{\sum_{i=1}^8 w_{i2}^2}} = \boxed{6.7736} \tag{32}$$

3 Problem 5. Convolutional Neural Networks

Suppose we are given a set of gray-scale images $D = I_{i}{}_{i=1}^{n}$, where $I_i \in R^{5\times5}$, n is the number of images. We want to build up a two-layer convolutional neural networks (CNN) to classify these images. Here, we only use one kernel in each CNN layer. Let $K_1 \in R^{2\times2}$ be the kernel of the first CNN layer and $K_2 \in R^{2\times2}$ be the kernel of the second CNN layer. (You may assume that the stride length of kernel is 1.) For simplicity, we use squared Frobenius norm of the output of the second CNN layer as our loss function.

Given an image $I_1 = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}$, the kernel K_1 and K_2 are initialized as $K_1 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$ and $K_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$.

Here is the procedure of the designed CNN model. We first input I_1 into the first layer of CNN, i.e., $S_1 = I_1 \bigodot K_1$, where $\bigodot$ denotes CNN operation. Then, we input S_1 into the second layer of CNN, i.e., $S_2 = S1 \bigodot K_2$, where $\bigodot$ denotes CNN operation. Finally, we use Frobenius norm of the output of the second CNN layer as our loss function, i.e., $L = ||S_2||_F^2$, where $||A||_F = \sqrt{\sum_i \sum_j (A[i][j])^2}$.

3.1 Given the input image I_1, please compute the value of matrix S_1, S_2 and L in the first iteration.

Since we know 2D convolution (Lecture 10 Page 23), in this give case, (with stride length 1) we will have:

$$S_1[i][j] = \sum_{q=-\frac{Q-1}{2}}^{\frac{Q-1}{2}} \sum_{p=-\frac{P-1}{2}}^{\frac{P-1}{2}} I_1[i+p][j+q]K_1[p][q] = \boxed{\begin{bmatrix} 2 & 0 & 2 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 1 \end{bmatrix}} \tag{33}$$

$$S_2[i][j] = \sum_{q=-\frac{Q-1}{2}}^{\frac{Q-1}{2}} \sum_{p=-\frac{P-1}{2}}^{\frac{P-1}{2}} S_1[i+p][j+q]K_2[p][q] = \boxed{\begin{bmatrix} 4 & 2 & 4 \\ 3 & 3 & 3 \\ 4 & 3 & 3 \end{bmatrix}} \tag{34}$$

$$L = ||S_2||_F^2 = \sum_i \sum_j (S_2[i][j])^2 = 4^2 + 2^2 + 4^2 + 3^2 + 3^2 + 3^2 + 4^2 + 3^2 + 3^2 = \boxed{97} \tag{35}$$

3.2 Please calculate the partial derivative of loss function L with respect to the kernel K_1 and K_2 in the first iteration.

[Hint: You may use the chain rule to solve this question, e.g., $\frac{\partial L}{\partial K_2[1][1]} = \sum_i \sum_j \frac{L}{\partial S_2[i][j]} \times \frac{\partial S_2[i][j]}{\partial K_2[1][1]}$

Let's use chain rule to solve this question. The general formula for $\frac{\partial L}{\partial K_2[p][q]}$ will be:

$$\frac{\partial L}{\partial K_2[p][q]} = \sum_i \sum_j \frac{L}{\partial S_2[i][j]} \times \frac{\partial S_2[i][j]}{\partial K_2[p][q]} = \sum_i \sum_j 2S_2[i][j] \times S_1[i+p][j+q] \tag{36}$$

Thus, plug in numerical data, we will have following result:

$$\begin{cases} \frac{\partial L}{\partial K_2[0][0]} = 2 \times (4 \times 2 + 2 \times 0 + 4 \times 2 + 3 \times 1 + 3 \times 1 + 3 \times 1 + 4 \times 1 + 3 \times 1 + 3 \times 1) = 70 \\ \frac{\partial L}{\partial K_2[0][1]} = 2 \times (4 \times 0 + 2 \times 2 + 4 \times 1 + 3 \times 1 + 3 \times 1 + 3 \times 1 + 4 \times 1 + 3 \times 1 + 3 \times 1) = 54 \\ \frac{\partial L}{\partial K_2[1][0]} = 2 \times (4 \times 1 + 2 \times 1 + 4 \times 1 + 3 \times 1 + 3 \times 1 + 3 \times 1 + 4 \times 2 + 3 \times 1 + 3 \times 1) = 66 \\ \frac{\partial L}{\partial K_2[1][1]} = 2 \times (4 \times 1 + 2 \times 1 + 4 \times 1 + 3 \times 1 + 3 \times 1 + 3 \times 1 + 4 \times 1 + 3 \times 1 + 3 \times 1) = 58 \end{cases} \tag{37}$$

which means,

$$\frac{\partial L}{\partial K_2} = \boxed{\begin{bmatrix} 70 & 54 \\ 66 & 58 \end{bmatrix}} \tag{38}$$

Similarly, the general formula for $\frac{\partial L}{\partial K_1[p][q]}$ will be:

$$\frac{\partial L}{\partial K_1[p][q]} = \sum_i \sum_j \frac{L}{\partial S_2[i][j]} \times \sum_n \sum_m \frac{\partial S_2[i][j]}{\partial S_1[n][m]} \frac{\partial S_1[n][m]}{\partial K_1[p][q]} = \sum_i \sum_j 2S_2[i][j] \times \sum_n \sum_m K_2[n][m] \times I[i+n+p][j+m+q] \tag{39}$$

Thus, plug in numerical data, we will have following result:

$$\begin{cases} \frac{\partial L}{\partial K_1[0][0]} = 92 \\ \frac{\partial L}{\partial K_1[0][1]} = 90 \\ \frac{\partial L}{\partial K_1[1][0]} = 102 \\ \frac{\partial L}{\partial K_1[1][1]} = 80 \end{cases} \tag{40}$$

which means,

$$\frac{\partial L}{\partial K_1} = \boxed{\begin{bmatrix} 92 & 90 \\ 102 & 80 \end{bmatrix}} \tag{41}$$