ECE368: Probabilistic Reasoning

Lab 1: Classification with Multinomial and Gaussian Models

Name: Mingyu Zheng **Student Number:** 1003797661

You should hand in: 1) A scanned .pdf version of this sheet with your answers (file size should be under 2 MB); 2) one figure for Question **1**.2.(c) and two figures for Question **2**.1.(c) in the .pdf format; and 3) two Python files classifier.py and ldaqda.py that contain your code. All these files should be uploaded to Quercus.

1 Naïve Bayes Classifier for Spam Filtering

1. (a) Write down the estimators for p_d and q_d as functions of the training data $\{\mathbf{x}_n, y_n\}, n = 1, 2, \ldots, N$ using the technique of "Laplace smoothing". (1 **pt**)

spam: $p_d = \dfrac{x_{nd}+1}{x_{n1}+\cdots+x_{nN}+N}$ $\{x_n, 1\}$

ham: $q_d = \dfrac{x_{nd}+1}{x_{n1}+\cdots+x_{nN}+N}$ $\{x_n, 0\}$

N: distinct words in spam & ham.

(b) Complete function learn_distributions in python file classifier.py based on the expressions. (1 **pt**)

2. (a) Write down the MAP rule to decide whether $y = 1$ or $y = 0$ based on its feature vector $\mathbf{x}$ for a new email $\{\mathbf{x}, y\}$. The d-th entry of $\mathbf{x}$ is denoted by x_d. Please incorporate p_d and q_d in your expression. Please assume that $\pi = 0.5$. (1 **pt**)

$y = \arg\max_y P(y|x) = \arg\max_y \dfrac{P(x|y)P(y)}{P(x)}$

$= \arg\max_y P(x|y)$

$= \arg\max_y \dfrac{(x_1+x_2+\cdots+x_D)!}{(x_1)!(x_2)!\cdots(x_D)!} \prod_{d=1}^{D} P(w_d|y)^{x_d} = \arg\max_y \prod_{d=1}^{D} P(w_d|y)^{x_d}$

$y=1$ $P(w_d|y) = p_d$

$y=0$ $P(w_d|y) = q_d$

$\prod_{d=1}^{D}(p_d)^{x_d} \underset{ham}{\overset{spam}{\gtrless}} \prod_{d=1}^{D}(q_d)^{x_d}$

(b) Complete function classify_new_email in classifier.py, and test the classifier on the testing set. The number of Type 1 errors is $\boxed{2}$, and the number of Type 2 errors is $\boxed{4}$. (1 **pt**)

(c) Write down the modified decision rule in the classifier such that these two types of error can be traded off. Please introduce a new parameter to achieve such a trade-off. (0.5 **pt**)

$\dfrac{\prod_{d=1}^{D}(p_d)^{x_d}}{\prod_{d=1}^{D}(q_d)^{x_d}} \underset{ham}{\overset{spam}{\gtrless}} P_{trade\text{-}off}$

new parameter.

Write your code in file classifier.py to implement your modified decision rule. Test it on the testing set and plot a figure to show the trade-off between Type 1 error and Type 2 error. In the figure, the x-axis should be the number of Type 1 errors and the y-axis should be the number of Type 2 errors. Plot at least 10 points corresponding to different pairs of these two types of error in your figure. The two end points of the plot should be: 1) the point with zero Type 1 error; and 2) the point with zero Type 2 error. Please save the figure with name **nbc.pdf**. (1 **pt**)

(d) If we do not use Laplace smoothing and simply use maximum likelihood estimation in the training phase, what will go wrong? What kind of emails such a classifier would fail to classify? (0.5 **pt**)

For the test files, if a word only shows in one type of email (spam/ham) then without Laplace Smoothing, we treat the probability of that word show in the other type as 0, which is not rigorous enough.

And $P_{wd} = \frac{n}{n} = 1$ Not depends on n(sample size)

2 Linear/Quadratic Discriminant Analysis for Height/Weight Data

1. (a) Write down the maximum likelihood estimates of the parameters $\boldsymbol{\mu}_m$, $\boldsymbol{\mu}_f$, $\boldsymbol{\Sigma}$, $\boldsymbol{\Sigma}_m$, and $\boldsymbol{\Sigma}_f$ as functions of the training data $\{\mathbf{x}_n, y_n\}, n = 1, 2, \ldots, N$. (1 **pt**)

$$\underline{\mu_m} = \frac{\sum_{i=1}^{n} I\{y_n=1\} \underline{x_i}}{\#\text{ of male}} \qquad \underline{\Sigma_m} = \frac{1}{\#\text{ of male}} \sum_{i=1}^{n} (\underline{x_i} - \underline{\mu_m})(\underline{x_i} - \underline{\mu_m})^T I\{y_n=1\}$$

$$\underline{\mu_f} = \frac{\sum_{i=1}^{n} I\{y_n=2\} \underline{x_i}}{\#\text{ of female}} \qquad \underline{\Sigma_f} = \frac{1}{\#\text{ of female}} \sum_{i=1}^{n} (\underline{x_i} - \underline{\mu_f})(\underline{x_i} - \underline{\mu_f})^T I\{y_n=2\}$$

$$\underline{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (\underline{x_i} - \underline{\mu})(\underline{x_f} - \underline{\mu})^T$$

where $\underline{\mu} = \frac{1}{n}(\underline{\mu_m} * \#\text{ of male} + \underline{\mu_f} * \#\text{ of female})$

(b) In the case of LDA, write down the decision boundary as a linear equation of $\mathbf{x}$ with parameters $\boldsymbol{\mu}_m$, $\boldsymbol{\mu}_f$, and $\boldsymbol{\Sigma}$. Note that we assume $\pi = 0.5$. (0.5 **pt**)

$$\underline{\mu_m}^T \Sigma^{-1} \underline{x} - \frac{1}{2} \underline{\mu_m}^T \Sigma^{-1} \underline{\mu_m}$$
$$\|$$
$$\underline{\mu_f}^T \Sigma^{-1} \underline{x} - \frac{1}{2} \underline{\mu_f}^T \Sigma^{-1} \underline{\mu_f}$$

In the case of QDA, write down the decision boundary as a quadratic equation of $\mathbf{x}$ with parameters $\boldsymbol{\mu}_m$, $\boldsymbol{\mu}_f$, $\boldsymbol{\Sigma}_m$, and $\boldsymbol{\Sigma}_f$. Note that we assume $\pi = 0.5$. (0.5 **pt**)

$$-\frac{1}{2}\log|\Sigma_m| - \frac{1}{2}(\underline{x} - \underline{\mu_m})^T \Sigma_m^{-1} (\underline{x} - \underline{\mu_m})$$
$$\|$$
$$-\frac{1}{2}\log|\Sigma_f| - \frac{1}{2}(\underline{x} - \underline{\mu_f})^T \Sigma_f^{-1} (\underline{x} - \underline{\mu_f})$$

(c) Complete function discrimAnalysis in ldaqda.py to visualize LDA and QDA models and the corresponding decision boundaries. Please name the figures as lda.pdf, and qda.pdf. (1 **pt**)

2. The misclassification rates are $\boxed{0.1182}$ for LDA, and $\boxed{0.1091}$ for QDA. (1 **pt**)