# CS584 Assignment 2: Report

Cong Liu
Department of Computer Science
Illinois Institute of Technology
March 17, 2016

## 1. Abstract

This is the report of CS584 assignment 2. In this assignment, I implement some generative learning algorithm cases using Gaussian discriminate analysis(GDA) and Naïve Bayes. All datasets are found in UCI website [1]. The program and results analysis are shown in HW2 Q*.html files.

## 2. Problem statement

In this assignment, I implement generative learning by 4 steps.

1) Select distribution model.
2) Determine model parameters (Maximum likelihood)
3) Compute membership $g_i(x)$
4) Classify: prediction = argmax (j) $g_j(x)$

The first two steps are training part. And the 3rd and 4th steps are belonging to testing part.

In the assignment, the first three questions are required to implement Gaussian discriminant analysis (GDA). In GDA, we assume all examples generalized Gaussian distribution.

$$\begin{aligned} y &\sim \text{Bernoulli}(\phi) \\ x|y=0 &\sim \mathcal{N}(\mu_0, \Sigma) \\ x|y=1 &\sim \mathcal{N}(\mu_1, \Sigma) \end{aligned}$$

The possibility of the one data belongs to a class can be estimated using[2]:

$$\begin{aligned} p(y) &= \phi^y(1-\phi)^{1-y} \\ p(x|y=0) &= \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)\right) \\ p(x|y=1) &= \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right) \end{aligned}$$

The parameters are φ, μ0, μ1, ∑, and maximum joint likelihood can be estimated by[3]:

$$\begin{aligned}\ell(\phi, \mu_0, \mu_1, \Sigma) &= \log \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) \\ &= \log \prod_{i=1}^{m} p(x^{(i)}|y^{(i)}; \mu_0, \mu_1, \Sigma) p(y^{(i)}; \phi).\end{aligned}$$

And for testing case, I try to find argmax j of classifier function g(x), j is the classification result of the test case.

$$\begin{aligned}\arg\max_y p(y|x) &= \arg\max_y \frac{p(x|y)p(y)}{p(x)} \\ &= \arg\max_y p(x|y)p(y).\end{aligned}$$

For second part I implement Bernoulli case and Binomial case on Naïve Bayes assumption.

It means that in Naïve Bayes,

$$\begin{aligned}&p(x_1, \ldots, x_{50000}|y) \\ &= p(x_1|y)p(x_2|y, x_1)p(x_3|y, x_1, x_2) \cdots p(x_{50000}|y, x_1, \ldots, x_{49999}) \\ &= p(x_1|y)p(x_2|y)p(x_3|y) \cdots p(x_{50000}|y) \\ &= \prod_{i=1}^{n} p(x_i|y)\end{aligned}$$

For question 4. In Bernoulli case, we assume x are only counted as 0 or 1. The joint distribution of (x, y) are estimate according to

$$\begin{aligned}p(y) &= (\phi_y)^y (1 - \phi_y)^{1-y} \\ p(x|y = 0) &= \prod_{j=1}^{n} p(x_j|y = 0) \\ &= \prod_{j=1}^{n} (\phi_{j|y=0})^{x_j} (1 - \phi_{j|y=0})^{1-x_j} \\ p(x|y = 1) &= \prod_{j=1}^{n} p(x_j|y = 1) \\ &= \prod_{j=1}^{n} (\phi_{j|y=1})^{x_j} (1 - \phi_{j|y=1})^{1-x_j}\end{aligned}$$

Then in test cases, by maximize likelihood function, the test case is classified into different classes.

Queston 5 is very similar with question 4, exept x can choose any numbers, not only 0 and 1.

## 3. Proposed solution

One of the problem I had when doing this assignment is to choose the right data files and format data sets. Especially for question 5, at first, I cannot find an easy and efficient way to compute p, the sum of all features of one example and easy to get the value by x index. Then I try to add p as the last column in the dataset, then the datasets are formatted as {label, features, pi}.

Another problem I met when I choose to compute sigma, the covariance of examples. I try to decrease the time complexity of my implementation, however it didn't work. I spent about 2 hours on this code pieces and finally give it up.

And a general problem is that the data type DataFrame and Series in pandas are very different with other types. During my implementation, lots of bugs are caused by them.

## 4. Implementation details

For the first three questions I use "iris.data" as datasets and format the dataset to fit the requirements. For question 1, I choose the last feature ($4^{th}$ column) and class 0 and 1 ( 1-100 rows) as my dataset. For question 2, I read all data from row 1 to row 100. So the dataset is a 4D 2-classes dataset. For question 3, I read the entire dataset. So it become a 4D 3-classes dataset. In each class, there are 50 examples.

In training and testing part, I randomly split dataset into 5-fold, choose 1 as test data and the remaining as training data.

The second part of this assignment is to implement Naïve Bayes cases. For the $4^{th}$ question I choose "ad.data" downloaded from UCI website. I randomly split dataset into 5-fold, choose 1 as test data and the remaining as training data. The last question I choose "SPECTF.train" as training data and "SPECTF.test" as test data.

## 5. Results and discussion

Results and discuss are shown in the coding attachments.

Demonstrations of correctness are also showed in attachments.

## 6. References

[1]. UCI Machine Learning Repository: http://archive.ics.uci.edu/ml/

[2]. Pattern Recognition and machine learning, Christopher M.Bishop,

[3]. Wikipedia: Naïve Bayes Classifer: https://en.wikipedia.org/wiki/Naive_Bayes_classifier