CS584 – Assignment 2

Generative learning

A20364639
Yung Chi Liu

- Problem statement

In this assignment, the main topic is to implement Generative learning algorithms. It including five algorithms: 1D 2-class Gaussian discriminate analysis (GDA). nD 2-class GDA, nD m-class GDA, Naïve Bayes with Bernoulli features and Naïve Bayes with Binomial features. Select data set on UCI machine learning website. And last, analyses each algorithms' performance by cross validation.

1. 1D 2-class Gaussian discriminant analysis

- Algorithm and Parameters :

1.) Compute μ_j, σj for each class j
2.) Compute membership function $g_j(x)$
3.) Classify $\hat{y} = \text{argMax}_j\ g_j(x)$

Parameters:
Mean: mean of feature x base on class j

$$\mu_j = \frac{1}{m_j}\sum_{i=1}^{m} 1(\ y^{(i)} = j)x^{(i)}$$

Covariance function:

$$\sigma j = \frac{1}{m_j}\sum_{i=1}^{m} 1\left(y^{(i)} = j\right)\left(x^{(i)} - \mu_j\right)^2$$

α_j: Probability of class j in all class

$$\alpha_j = \frac{\text{number of example from class j}}{\text{number of total example}}$$

Member function:

$$g_j(x) = \log\frac{1}{\sqrt{2\pi}} - \log\left(\frac{1}{\sigma j}\right) - \log\left(\frac{-(x-\mu_j)^2}{2\sigma j^2}\right) + \log(\alpha_j)$$

- Data set selection :

I chose 2 data set for this program.

1. mammographic-masses data set

Attribute Information:

1. BI-RADS assessment: 1 to 5 (ordinal)
2. Age: patient's age in years (integer)
3. Shape: mass shape: round=1 oval=2 lobular=3 irregular=4 (nominal)
4. Margin: mass margin: circumscribed=1 microlobulated=2 obscured=3 ill-defined=4 spiculated=5 (nominal)
5. Density: mass density high=1 iso=2 low=3 fat-containing=4 (ordinal)
6. Severity: benign=0 or malignant=1 (binominal)

I select second feature as 1-D feature, because the value is more important than the first one which had bigger range of value, and other features had missing data.

2. Iris Plants Database

Attribute Information:

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm

Class:

Iris Setosa

Iris Versicolour

Iris Virginica

I chose first feature as x and Iris Setosa and Iris Versicolour as 2 classes (Setosa = 0, Versicolour = 1).

Result and Discussion

1.) mammographic-masses data set

10-fold cross validation:

Precision = 0.710372014891

Recall = 0.679793811166

Accuracy = 0.675

F-measure = 3.78833594358

Confusion Matrix:

Testing example = 96

C \ R	P	N
P	34.9	14.6
N	16.6	29.9

As the result, there still had about 30% classify error. One reason is that only use one feature to base on a multiple feature data, the accuracy could not high. Moreover, I used first feature as 1D 2-class the result is worse. Because the first feature one had integers from range 1 to 5.

2.) Iris Plants Database

10-fold cross validation:

Precision = 0.874047619048
Recall = 0.764761904762
Accuracy = 0.84
F-measure = 4.58066001928

Confusion Matrix:
Testing example = 10

C \ R	P	N
P	3.9	0.5
N	1.1	4.5

The result is not best as expected, only got 50% of correct classification. As extract 1 feature (length cm) from 4 features. There is hard to get enough training, and so the classify is not get expect result.

2. nD 2-class Gaussian discriminant analysis

■ Data selection:

Use the same data as 1D 2class GDA. This time use more than one feature as data.

1. mammographic-masses data set

As this data set, I chose 2 feature as classification. I used first 2 features, because the other had missing data. ("?" in the .data file)

2. Iris Plants Database

Use same data as 1D 2feature classification. Now, read all 4 features and do the classification.

■ Algorithm and Parameters

Algorithm:

1.) Select distribution model. (use GDA)
2.) Determine model Parameters.
3.) Compute member function
4.) Classify $\hat{y} = \mathrm{argMax}_j\ g_j(x)$

Parameters:

Mean: mean of feature x base on class j

$$\mu_j = \frac{1}{m_j}\sum_{i=1}^{m} 1(\, y^{(i)} = j)x^{(i)}$$

Covariance function:

$$\sum j = \frac{1}{m_j}\sum_{i=1}^{m} 1\left(y^{(i)} = j\right)\left(x^{(i)} - \mu_j\right)\left(x^{(i)} - \mu_j\right)^T$$

α_j: Probability of class j in all class

$$\alpha_j = \frac{\text{number of example from class j}}{\text{number of total example}}$$

Member function:

$$g_j(x) = -\frac{1}{2}\log(|\sum j|) - \frac{1}{2}\left(x^{(i)} - \mu_j\right)\sum j^{-1}\left(x^{(i)} - \mu_j\right)^T + \log\left(\alpha_j\right)$$

Discriminate Function

$d_j(x) = g_j(0)$- $g_j(1)$ if $d_j(x) > 0$ -> class 0

If $d_j(x) > 0$ -> class 1

- Result and Discussion

1.) mammographic-masses data set

10-fold cross validation:

Confusion Matrix:

Testing example = 96

Precision = 0.702745434712

Recall = 0.90606294992

Accuracy = 0.738541666667

F-measure= 4.38762769869

C \ R	P	N
P	46.6	20.2
N	4.9	24.3

As the result, compare to 1D2Class GDA, the accuracy is better. More than 75% of testing data war classify to the right place. For short, use more feature as data sets could get better classification than just use 1 feature as data sets.

2.) Iris Plants Database

10-fold cross validation:

Precision =1.0

Recall = 1.0

Accuracy =1.0

F-measure = 1.0

Confusion Matrix:

Testing example = 10

C \ R	P	N
P	5.0	0
N	0	5.0

As iris data sets use 4 features, there is no error happen. In my opinions, may be first class and second class in this data set NOT had any noise (incorrect data). That's why all the classification gets the right result

3. nD m-class Gaussian discriminant analysis

■ Data selection:

In nD m-class Gaussian discriminant analysis problem, I chose Iris data set. In this case, the data is 4D 3-class. And I modify the label of class by replace Iris-Setosa to class 0, Iris-Versicolour to class 1 and Iris-Virginica to class 2.

■ Algorithm and Parameters

Algorithm:

1.) Select distribution model. (use GDA)

2.) Determine model Parameters.

3.) Compute member function

4.) Classify $\hat{y} = \text{argMax}_j \, g_j(x)$

The parameters use same parameters as in nD 2class GDA.

The only difference in between nD m-class and nD 2-class is Discriminate Function, since we have n-class with n-member ship function. The classifications will chose the LARGEST member ship function as class of each example:

$$d_j(x) = \text{argMax}_j \, g_j(x)$$

- Result and Discussion

1. Iris data:

5-fold cross validation:

Confusion Matrix:

Testing example = 30

C \ R	C0	C1	C2
C0	10	0	0
C1	0	7.4	0
C2	0	0.6	10

Base on Class 0:

Precision =1.0

Recall = 1.0

Accuracy = 0.9799999993

F-measure = 1.0

Base on Class 1:

Precision = 1

Recall = 0.9323308271

Accuracy = 0.98

F-measure = 0.9612612613

Base on Class 2:

Precision =0.7636363636

Recall = 1.0

Accuracy =0.98

F-measure = 0.9646153846

With 3 class as data set, the iris data set had a minute error. One reason could be class 1's feature value is close to class 2's feature value, and class 0's feature value is not close to class 1's feature value. That why when testing as 2-class GDA, there is less error than 3-class GDA.

4. Naïve Bayes with Bernoulli features

- Data selection: (spect heart)

Attribute Information:

Number of Attributes: 23 (22 binary + 1 binary class)

22 feature 2 class data sets

- Algorithm and Parameters

1.) Select distribution model.
2.) Determine model Parameters. α_i ,$\alpha_{j|y=i}$
3.) Compute member function
4.) Classify $\hat{y} = \mathrm{argMax}_j\ g_j(x)$

α_i: Probability of class i in all class

$$\alpha_i = \frac{\sum_{j=1}^{n} 1(y^j = i)}{m}$$

n: number of features in 1 sample.

m: number of total sample.

i: i-th class

j: j-th feature

$\alpha_{j|y=i}$: Probability of j-th feature base on i-th class.

$$\alpha_{j|y=i} = \frac{\sum_{j=1}^{m} 1(y^{(i)} = i)x_j^{(i)}}{\sum_{j=1}^{m} 1(y^{(i)} = i)}$$

Member function:

$$g_i(x) = \prod_{j=1}^{n} \alpha_{j|\,y=i}^{x_j}\left(1 - \alpha_{j|y=i}\right)^{1-x_j}$$

$$= \left[\sum_{j=1}^{m} x_j \log\left(\alpha_{j|\,y=i}\right) + \left(1 - x_j\right)\log\left(1 - \alpha_{j|\,y=i}\right)\right] + \log \alpha_i$$

■ Result and Discussion

1. : spect heart

10-fold cross validation:

Confusion Matrix:

Testing example = 26

Precision = 0.719896802282

Recall = 0.784581646424

Accuracy = 0.726923076923

F-measure = 4.01397241818

C \ R	P	N
P	12.2	4.2
N	2.9	6.7

As the result the testing sample with 10 fold cross validation is expected. Most of the data is classification to the right class.

5. Naïve Bayes Binominal features

- Data selection:

  Monks.data

  Attribute Information:

  1. class: 0, 1
  2. a1: 1, 2, 3
  3. a2: 1, 2, 3
  4. a3: 1, 2
  5. a4: 1, 2, 3
  6. a5: 1, 2, 3, 4
  7. a6: 1, 2

  * IN THIS CASE CLASS DATA PUT AT FIRST VALUE.

- Algorithm and Parameters

  Algorithm:

  1.) Select distribution model (NB binominal)

  2.) Determine model Parameters. α_i ,$\alpha_{j|y=i}$

  3.) Compute member function g_j(x)

  4.) Classify $\hat{y} = \text{argMax}_j\ g_j$(x)

  Parameters:

  α_i: Probability of class i in all class

  $$\alpha_i = \frac{\sum_{j=1}^{n} 1(y^j = i)}{m}$$

  n: number of features in 1 sample.

  m: number of total sample.

  i: i-th class

  j: j-th feature

  $P^{(i)}$ = total number words in document i

  $\alpha_{j|y=i}$: Probability of j-th feature base on i-th class.

  $$\alpha_{j|y=i} = \frac{\sum_{j=1}^{m} 1(y^{(i)} = i) x_j^{(i)}}{\sum_{j=1}^{m} 1(y^{(i)} = i) P^{(i)}}$$

Using Max likelihood derive the parameter estimate equation for NB:

Log likelihood function :

$$\ell(\theta) = \log \prod_{i=1}^{m} P\left(x_j^{(i)} \mid y^{(i)} : \theta\right) P\left(y^{(i)}\right) \quad \text{(Base on IID)}$$

$$= \log \prod_{i=1}^{m} \left[\prod_{j=1}^{n} P\left(x_j^{(i)} \mid y^{(i)} : \theta\right)\right] P\left(y^{(i)}\right) \quad \text{(Base on NB)}$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{n} \log P\left(x_j^{(i)} \mid y^{(i)} : \theta\right) + \sum_{i=1}^{m} P\left(y^{(i)}\right)$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{n} \log \binom{P^{(i)}}{x_j^{(i)}} \alpha_{j\mid y=y^{(i)}}^{x_j^{(i)}} \left(1 - \alpha_{j\mid y=y^{(i)}}^{x_j^{(i)}}\right)^{P^{(i)} - X_j^{(i)}} + \sum_{i=1}^{m} P\left(y^{(i)}\right)$$

$$\theta^* = \underset{\theta}{\text{argMax}}\ \ell(\theta)$$

$$\frac{\partial \ell}{\partial \theta} = 0, \theta = \left[\alpha_{1\mid y=1} \dots \alpha_{n\mid y=k} \dots \alpha_{n\mid y=1} \dots \alpha_{n\mid y=k} \dots \alpha_1 \dots \alpha_k\right]$$

From $\left[\alpha_{1\mid y=1} \dots \alpha_{n\mid y=k} \dots \alpha_{n\mid y=1} \dots \alpha_{n\mid y=k}\right]$

$$\frac{\partial \ell}{\partial \alpha_{i\mid y=j}} = 0,\ \alpha_\ell = \frac{\sum_{j=1}^{n} 1(y^j = \ell)}{m}$$

From $[\alpha_1 \dots \alpha_k]$

$$\frac{\partial \ell}{\partial \alpha_j} = 0, \alpha_{j\mid y=i} = \frac{\sum_{j=1}^{m} 1\left(y^{(i)} = \ell\right) x_j^{(i)} + \epsilon}{\sum_{j=1}^{m} 1\left(y^{(i)} = \ell\right) P^{(i)} + k\epsilon}$$

Compute member ship function:

$$g_i(x) = \sum_{j=1}^{n} \log \binom{P^{(i)}}{x_j^{(i)}} \alpha_{j\mid y=\ell}^{x_j^{(i)}} \left(1 - \alpha_{j\mid y=\ell)}^{x_j^{(i)}}\right)^{P^{(i)} - X_j^{(i)}} + \log(\alpha_\ell)$$

Result and Discussion

1. : monk data

10-fold cross validation:

Confusion Matrix:

Testing example = 55

Precision = 0.543943923723

Recall = 0.543007317858

Accuracy = 0.518181818182

F-measure= 2.81853034612

C \ R	P	N
P	14.4	13.5
N	13.0	14.1

As the result the performance is like 50% chance right.

Conclusion:

In this assignment, the knowledge of Generative learning had been improved by compute the real program base on some real data sets. The First problem is how to find a fit data set for each algorithm. Because not all data set is perfect Gaussian Distributed and there could be some noise in the training data set.

Moreover, I got some problem when compute Naïve Bayes. When compute $\alpha_{j|y=i}$, if all j-th feature happened in i-the class the probability is 0 and that make equation not work. So I add ∈ at Numerator and 2∈ at denominator to solve the problem.

At last, after finish this assignment. It improved my skill in using numpy in Python and understanding of Generative learning.

References

1. data set UCI website. http://archive.ics.uci.e du/ml/

2. Gady Agam. Lecture note. Generative learning