2020

CS 331 – Introduction to Artificial Intelligence

ASSIGNMENT 03 – PART 02

PREPARED BY: DYASS KHALID / MARHABA TALAL

Honor Code

You are to do this assignment by yourself. All of the submitted code should be your creation and a result of your own thought process. No cooperation is allowed under any circumstances. Any help outside of the course staff is prohibited. You are also given the responsibility of reporting any cooperation incidences to the course staff. All code will be checked for similarities, and cases will be promptly forwarded to relevant authorities for action.

Important Instructions

- You are not allowed to use any pre-existing machine learning libraries. You need to implement the naïve Bayes algorithm yourself.
- This assignment needs to be done in Python.
- You are required to use Numpy where necessary.
- You have to take command line arguments:

```
python classifier.py Spect_train Spect_test
```

Make sure you follow this format.

Naïve Bayes Classifier

The goal of this assignment is to implement Naïve Bayes Classifier. You will use the Bernoulli naïve Bayes model for the classification task. Bernoulli model requires that all attributes value is binary as a result the dataset of SPECT, provided to you, contains only binary values.

Explanation of dataset:

Each patient is classified into two categories: Normal and Abnormal, depending on the number of medical tests he/she passes. The database contains 267 patients' data, every person underwent 22 medical tests and each test was either pass or fail. As a result, for each patient 22 binary values were extracted.

You have been provided with two files, **Spect_train** and **Spect_test**. Spect_train has a total 80 data points and Spect_test has 187 data points.

You will use Spect_train patient data to train your naïve Bayes classifier and Spect_test to test it.

A single patient in the dataset is described as a single line of the file. So, each line has 23 values, the first value of each line describes whether the person was described as normal (value of 1) or abnormal (value of 0). All other 22 values define which test number the patient failed and which he/she passed.

For example:

The first line of SPECT_train is 1,0,0,0,1,0,0,0,1,1,0,0,0,1,1,0,0,0,0,0,0,0,0.

It can be interpreted as:

Patient	Test1	Test2	Test3	Test4	Test5	…	Test20	Test21	Test22
Normal (1)	Fail (0)	Fail (0)	Fail (0)	Pass (1)	Fail (0)	…	Fail (0)	Fail (0)	Fail (0)

If the above data point started from zero: 0,0,0,0,1,0,0,0,1,1,0,0,0,1,1,0,0,0,0,0,0,0,0

Patient	Test1	Test2	Test3	Test4	Test5	…	Test20	Test21	Test22
Abnormal (0)	Fail (0)	Fail (0)	Fail (0)	Pass (1)	Fail (0)	…	Fail (0)	Fail (0)	Fail (0)

Tasks

You have to implement the Bernoulli naïve Bayes classifier for the above set such that given 22 medical test reports of a person, your classifier predicts whether the person is normal or abnormal. You will test your classifier using **Spect_test** file.

You also have to write a short report on the process, and you will be marked on the quantity of the content, so write anything you feel deserves credit. (Should not be more than a page). Take both **Spect_test** and **Spect_train** files as input from command line as mentioned in the instructions. Both your training and testing code will be in the same file (classifier.py), you can divide it in classes or functions as appropriate.

Sample Output:

You should correctly show your output on the screen.

##########

Starting to Train on 80 data points . . .

Training Complete

Testing on 187 data point . . .

Total Accuracy: = 90%

##########

Marking Criteria:

Implementation	45%
Accuracy	30%
Report	20%
Output	05%

- Failing to follow the instructions will lead to loss of marks.

Files to Submit:

- Classifier.py

- Report.txt

Zip all the files in a folder named: YourRollNumber_Assignment3_Part02.zip. All files will go through Moss and no plagiarism act will be tolerated.

Some Notes below to help you Get Started:

Bayes Classifier

- Bayes rule: $p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$
- Estimate class-conditional distribution $p(\mathbf{x}|C_k)$ and class prior probability $p(C_k)$
- Classification rule: assign $\mathbf{x}$ label C_i such that $i = \arg\max_i p(\mathbf{x}|C_i)p(C_i)$

In our case →

C1 = Normal
C2 = Abnormal

Class Prior, $p(C_k)$

C1 → class 1
C2 → class 0

output 0 or 1

- For $K = 2$ (binary classification problem), $t \in \{C_1, C_2\}$ is a binary variable following Bernoulli distribution
  - ML estimate for $p(C_1)$ is N_1/N and that for $p(C_2)$ is N_2/N
- For $K > 2$ (K-class problem), $t \in \{C_1, C_2, \dots, C_K\}$ is a categorical variable following Categorical distribution
  - ML estimate for $p(C_k)$ is N_k/N

Naïve Bayes classifier

- Assume each variable is independent of the others given the class label
  - $p(\mathbf{x}|C_k) = \prod_{i=1}^{D} p(x_i|C_k)$
- Classification rule: assign label C_i to $\mathbf{x}$ such that $i = \arg\max_{i} p(C_i) \prod_{j=1}^{D} p(x_j|CI\ )$

Bernoulli Naïve Bayes Classifier (BNBC) (1)

- $\mathbf{x}$ is a vector of binary (0,1) variables following Bernoulli distribution
- $p(\mathbf{x}|C_k) \sim \prod_{i=1}^{D} Bern(x_i|\mu_{ik}) = \prod_{i=1}^{D} \mu_{ik}^{x_i}(1-\mu_{ik})^{(1-x_i)}$
- ML estimate: $\mu_{ik} = \frac{n_{ik}}{N_k}$
  - n_{ik}: count of $x_i = 1$ in C_k, N_k: no. of examples in C_k

Good luck!