ELG 5255 Applied Machine Learning Fall 2021

Assignment 2 (Bayesian Decision)

Submission

You must submit your assignment on-line with Bright Space. This is the only method by which we accept assignment submissions. We do not accept assignments sent via email, and we are not able to enter a mark if the assignment is not submitted on Bright Space! The deadline date is firm since you cannot submit a assignment passed the deadline. It is student's responsibility to ensure that the assignment has been submitted properly. A mark of 0 will be assigned to any missing assignment.

Goal

This assignment aims to understand Naïve Bayes Classification and implement Risk-based Bayesian Decision Theory Classifier.

Dataset

During this assignment, the well-known Iris flower dataset is used.

Iris dataset contains 150 samples, 4 features (i.e., sepal length, sepal width, petal length, petal width), and 3 classes (i.e., Iris-Setosa, Iris-Versicolour, Iris-Virginica).

The dataset is already available in sklearn lib; therefore the following code can help you load the dataset

```
from sklearn import datasets
from sklearn.model_selection import train_test_split
iris = datasets.load_iris()
X, y = iris.data, iris.target
trX, teX, trY, teY = train_test_split(X, y, random_state=0)
```

Part 1

Given the training data in Table 1 (from Iris dataset), predict the class of of following examples in Table 2 using Naïve Bayes Classification (assume each feature has normal distribution). (Please note that you must answer the following questions and show your calculation process and the formulas you use. You won't receive marks if you only provide answers.)

Table 1: Training Data

sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	Label
6.5	3.0	5.8	2.2	2
4.9	2.4	3.3	1.0	1
5.7	3.0	4.2	1.2	1
7.7	2.6	6.9	2.3	2
5.4	3.0	4.5	1.5	1
5.1	3.4	1.5	0.2	0
5.0	3.4	1.5	0.2	0
5.6	2.5	3.9	1.1	1
4.8	3.0	1.4	0.1	0
5.0	3.3	1.4	0.2	0

Table 2: Testing Data

sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	Label
5.7	2.8	4.5	1.3	
5.4	3.9	1.3	0.4	

Q1: Calculate mean value of each feature in Table 1 (**5 marks**)

Q2: Calculate variance of each feature in Table 2 (**5 marks**)

Q3: Calculate the posterior probabilities for each sample for each class in Table 2 and predict classes (**10 marks**)

Part 2

Please note the difference among Iris dataset (with 4 features and 3 classes), 2D Iris dataset (with 2 features and 3 classes) which is obtained by dropping the last 2 features (i.e., petal length and petal width) and keep Sepal length and Sepal width.

Please submit your code and report (including screen shot of code and the relevant figures).

1. Load Iris Dataset

   Please set random_state ad 0

   ```
   from sklearn import datasets
   from sklearn.model_selection import train_test_split
   iris = datasets.load_iris()
   X, y = iris.data, iris.target
   trX, teX, trY, teY = train_test_split(X, y, random_state=0)
   ```

2. Drop the petal length and petal width features to form a 2D Iris dataset

3. Apply Naïve Bayes Classifier to get training and testing accuracy (**10 marks**)
4. Tune hyperparameters of Naive Bayes Classifier (i.e., var_smoothing). Try var_smoothing as 1e-9, 1e-8, 1e-7. Plot accuracy vs var_smoothing curve for training and testing set. (**10 marks**)
5. Develop Risk-based Bayesian Decision Theory Classifier (RBDTC)

   Please refer the lecture slides `ELG5255EG00_Lecture3_Fall21` for detailed explanation

   - The classifier must inherit BaseEstimator and ClassifierMinxin in sklearn. Please check the link for detailed information about developing
   - The classifier should check inputs and model status as other well defined models in sklearn
   - Implement `__init__` function in proper way. `__init__` should be able to take risk matrix and any kind of base estimator as inputs (**5 marks**)
   - Implement `fit` function in proper way. `fit` should be able to take training dataset (X and y) as input. In this function the base estimator should be trained while original base estimator should not be changed (**5 marks**)
   - Implement `predict_proba` function. `predict_proba` should only take testing X as input. The output of `predict_proba` should be a matrix $\mathbb{R}^{N*M}$, where N is the number of testing samples and M is the number of classes. The element of each element in the matrix $R(\alpha_j|x_i) = \sum_{k=1}^{M} \lambda_{jk} P(C_k|x_i)$, where x_i is the i^{th} sample in testing set, C_k is the k^{th} class, λ_{jk} is the j^{th} row and k^{th} element in risk matrix. The element of output matrix does not have to be scaled to 0~1. Use no more than one loop (**5 marks**)
   - Implement `predict`function in proper way. `predict` should be able to output text format y (**5 marks**)
6. Apply Risk-based Bayesian Decision Theory Classifier which takes Naïve Bayes Classifier as base estimator and uses Table 3 as risk matrix (**10 marks**)

	0	**1**	**2**
0	-10	-5	-5
1	-5	-10	-5
2	-5	-5	-100

7. Plot decision boundary and calculate precision, recall and accuracy for training and testing set (**10 marks**)
8. Compare and analysis the performance between NB and RBDTC regarding to their decision boundary, precision, recall, and accuracy(**10 marks**)
9. Provide a conclusion section on your report. Include overview of what you have done and learnt during the assignment. Aim no less than one third of a page and no more than half page. (**10 Marks**)