Homework 5

姓名：毕秋宇　　**学号**：171860624

2019 年 5 月 22 日

第一部分　作业

Problem 1 (Naive Bayes Classifier)

We learned about the naive Bayes classifier using the "property conditional independence hypothesis". Now we have a data set as shown in the following table:

	x_1	x_2	x_3	x_4	y
Instance1	1	1	1	0	1
Instance2	1	1	0	0	0
Instance3	0	0	1	1	0
Instance4	1	0	1	1	1
Instance5	0	0	1	1	1

(1) [10pts] Calculate: $\Pr\{y=1|\mathbf{x}=(1,1,0,1)\}$ and $\Pr\{y=0|\mathbf{x}=(1,1,0,1)\}$.

(2) [10pts] After using Laplacian Correction, recalculate the value in the previous question.

Solution

(1)

$$\Pr\{y=1|\mathbf{x}=(1,1,0,1)\}=\frac{\Pr\{y=1,x=(1,1,0,1)\}}{\Pr\{x=(1,1,0,1)\}}$$

$$=\Pr\{y=1\}\times\Pr\{y=1|x_1=1\}\times\Pr\{y=1|x_2=1\}\times\Pr\{y=1|x_3=0\}\times\Pr\{y=1|x_4=1\}$$

$$=\frac{3}{5}\times\frac{2}{3}\times\frac{1}{2}\times 0\times\frac{2}{3}=0$$

$$\Pr\{y=0|\mathbf{x}=(1,1,0,1)\}=\frac{\Pr\{y=0\}\Pr\{\mathbf{x}=(1,1,0,1)|y=0\}}{\Pr\{\mathbf{x}=(1,1,0,1)\}}$$

$$=\Pr\{y=0\}\times\Pr\{\mathbf{x}_1'=\mathbf{x}_1|y=0\}\times\Pr\{\mathbf{x}_2'=\mathbf{x}_2|y=0\}\times\Pr\{\mathbf{x}_3'=\mathbf{x}_3|y=0\}\times\Pr\{\mathbf{x}_4'=\mathbf{x}_4|y=0\}$$

$$=\frac{2}{5}\times\frac{1}{2}\times\frac{1}{2}\times\frac{1}{2}\times\frac{1}{2}=\frac{1}{40}$$

(2) Using the Laplacian correction, we get that

$$\hat{\Pr}\{y=1|\mathbf{x}=(1,1,0,1)\}=\frac{\hat{\Pr}\{y=1,x=(1,1,0,1)\}}{\hat{\Pr}\{x=(1,1,0,1)\}}$$

$$=\hat{\Pr}\{y=1\}\times\hat{\Pr}\{y=1|x_1=1\}\times\hat{\Pr}\{y=1|x_2=1\}\times\hat{\Pr}\{y=1|x_3=0\}\times\hat{\Pr}\{y=1|x_4=1\}$$

$$=\frac{4}{7}\times\frac{3}{5}\times\frac{2}{5}\times\frac{1}{5}\times\frac{3}{5}=\frac{72}{4375}$$

$$\hat{\Pr}\{y=0|\mathbf{x}=(1,1,0,1)\} = \frac{\hat{\Pr}\{y=0\}\hat{\Pr}\{\mathbf{x}=(1,1,0,1)|y=0\}}{\hat{\Pr}\{\mathbf{x}=(1,1,0,1)\}}$$
$$= \hat{\Pr}\{y=0\} \times \hat{\Pr}\{\mathbf{x}_1'=\mathbf{x}_1|y=0\} \times \hat{\Pr}\{\mathbf{x}_2'=\mathbf{x}_2|y=0\} \times \hat{\Pr}\{\mathbf{x}_3'=\mathbf{x}_3|y=0\} \times \hat{\Pr}\{\mathbf{x}_4'=\mathbf{x}_4|y=0\}$$
$$= \frac{3}{7} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} = \frac{3}{112}$$

☑ Problem 2 (Bayes Optimal Classifier)

For a binary classification task, when data in the two classes satisfies Gauss distribution and have the same variance, please prove that LDA can produce the bayes optimal classifier.

✎ Solution

By the definition of Gauss distribution, we get that $f_c(x) = \frac{e^{-\frac{(x-\mu)^T(x-\mu)}{2\Sigma}}}{\sqrt{(2\pi)^d|\Sigma|}}$
Where Σ is the covariance matrix, μ is the mean vector.
So we can get that the bayes optimal classifier satisfies

$$h(x) = \arg\max_c \Pr\{c|x\} = \arg\max_c \Pr\{c\} f_c(x)$$

The boundary between two classes is $\Pr\{c_1\}f_{c_1}(x) = \Pr\{c_2\}f_{c_2}f_{c_2}(x)$.
Bring the function above and take the logarithm of both sides we get that the boundary is

$$\log(\Pr\{c_1\}) - \frac{(x-\mu_{c_1})^T(x-\mu_{c_1})}{2\Sigma} = \log(\Pr\{c_2\}) - \frac{(x-\mu_{c_2})^T(x-\mu_{c_2})}{2\Sigma}$$

That is to say,

$$\log(\frac{\Pr\{c_1\}}{\Pr\{c_2\}}) = \frac{(x-\mu_{c_1})^T(x-\mu_{c_1}) - (x-\mu_{c_2})^T(x-\mu_{c_2})}{2\Sigma}$$

So we get that

$$\log(\frac{\Pr\{c_1\}}{\Pr\{c_2\}}) = \frac{x^Tx - 2x^T\mu_{c_1} + \mu_{c_1}^T\mu_{c_1} - x^Tx + 2x^T\mu_{c_2} - \mu_{c_2}^T\mu_{c_2}}{2\Sigma} = \frac{2x^T(\mu_{c_2}-\mu_{c_1}) + \mu_{c_1}^T\mu_{c_1} - \mu_{c_2}^T\mu_{c_2}}{2\Sigma}$$

That is

$$2\log(\frac{\Pr\{c_1\}}{\Pr\{c_2\}}) - 2x^T\Sigma^{-1}(\mu_{c_2}-\mu_{c_1}) - \mu_{c_1}^T\Sigma^{-1}\mu_{c_1} + \mu_{c_2}^T\Sigma^{-1}\mu_{c_2}$$

From LDA method, we know $w = \Sigma^{-1}(\mu_{c_1} - \mu_{c_2})$ in the target function $w^Tx + b$, it is same as we get from bayes optimal classifier.
Finally, we proved that LDA can produce the bayes optimal classifier.

☑ Problem 3 (Ensemble Methods in Practice)

Due to their outstanding performance and robustness, ensemble methods are very popular in machine community. In this experiment we will practice ensemble learning methods based on two classic ideas: Boosting and Bagging.

In this experiment, we use an UCI dataset Adult. You can refer to the link[1] to see the data description and download the dataset.

Adult is an class imbalanced dataset, so we select AUC as the performance measure. You can adopt sklearn to calculate AUC.

(1) [10pts] You need finish the code in Python, and only have two files: AdaBoost.py, RandomForest-Main.py. (The training and testing process are implemented in one file for each algorithm.)

(2) [40pts] The is experiment requires to finish the following methods:

1. Implement AdaBoost algorithm according to the Fig(8.3), and adopt decision tree as the base learner (For the base learner, you can import sklearn.)
2. Implement Random Forest algorithm. Please give a pseudo-code in the experiment report.
3. According to the AdaBoost and random forest, analysis the effect of the number of base learners on the performance. Specifically, given the number of base learners, use 5-fold cross validation to obtain the AUC. The range of the number of base learners is decided by yourself.
4. Select the best number of base classifiers for AdaBoost and random forests, and obtain the AUC in the test set.

(3) [10pts] In the experimental report, you need to present the detail experimental process. The experimental report needs to be hierarchical and organized, so that the reader can understand the purpose, process and result of the experiment.

Solution

(1) Two files are in the same folders.

(2) The pseudo-code of Random Forest algorithm are as follows:

Algorithm 1 RANDOM FOREST

Input: X_{train} , y_{train}

Output: $Model$

function FIT(X_{train}, y_{train})
    for $tree$ in $forest$
    $idx = randsamp(X_{train}.features)$
    $x_{train} = X_{train}[idx]$
    $tree.fit(x_{train})$
    return $model$

(3) Firstly, I download the "adult.data" and "adult.test" files and then use read csv function to read the data. However, the data exists deficiency and string value, so I use fillna and getdummies to convert it to the workable data. When preprocessing the string values, I find that the types of test file are different from the train file, so I concat those two DataFrame using getdummies function and then split it to the original size. And then, I create AdaBoost and RandomForest algorithm using the DecisionTreeRegresser from sklearn. I choose DecisionTreeRegresser other than DecisionTreeClassigier because I need to calculate the AUC value. In the predict part, if regression value is bigger than 0.5 we decide it as positive, we decide it as negative otherwise.

Last but not least, I need to decide the hype-parameters in AdaBoost and RandomForest algorithm. I use gridsearchCV function to select the best values.

The final AUC are as follows:

```
AUC =  0.9103534113218168
python RandomForestMain.py  84.41s user 49.87s system 95% cpu 2:20.01 total
```

Figure 1: AUC of RandomForest

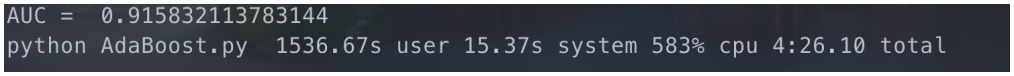

Figure 2: AUC of AdaBoost

And the number of weak classifiers is 2000, the number of estimators is 2000, min samples split is 2, min impurity decrease is 0 and splitter is random.

第二部分　订正

第三部分　反馈