EE447 Lab4 Report

Prof. Luoyi Fu, EE447 Mobile Internet

Spring semester, 2020

Fan Zhou(周凡)
517030910305

1 Requirements

This Lab focuses on learning basic machine learning methods and implementing them on a specific topic, to find advisor- advisee relationships in academic heterogeneous networks. In this Lab, you will learn some machine learning tools and realize your own model based on Python and Sklearn. Moreover, you will use Keras and Tensorflow to build a deep-learning model and compare the performance between deep learning and traditional machine learning methods.

2 3 Used Machine Learning Methods

2.1 SVM

SVM is short version of Support Vector Machine. Usually we use it to perform classification tasks, and the objective is to maximize the classifying margin. Let's denote input as X and the output y can be calculated as:

$$y = \begin{cases} +1 & w^T x + b \geq 0 \\ -1 & w^T x + b < 0 \end{cases}$$

Denote the ground truth label y^*, then we define the margin γ as:

$$\gamma = y^* y = y^*(w^T x + b)$$

For each training data in the training set, we calculate such classifying margin. It is obvious that in such situation, the larger the sum of margins, the better the model will be. Moreover, the problem can be reformed as an optimization problem like below:

$$\min_{w,b} \frac{1}{2}||w||^2$$
$$\text{s.t.} \quad \forall i, -y_i(w^T X_i + b) + 1 \leq 0$$

In practice, we also use different kernel functions to deal with non-linear classification problems.
A very common kernel is RBF kernel:

$$K(X_i, X_j) = \exp\big(-\frac{||X_i - X_j||^2}{2\sigma^2}\big)$$

In this lab, I also use this kernel to perform the AAR problem.

2.2 Decision Tree

A tree-structured model to perform prediction. In my opinion, the decision tree is very similar to human being thinking. If the input's features satisfy some condition, the decision tree will go to one of the branches. Eventually, it comes to a leaf node, *i.e.* the final prediction.

The decision tree is usually thought to have following characteristics:

- highly explainable, can be transferred to series of rules

- can deal with non-linear data; don't need to normalize the input
- can be a sub-optimal solution rather than the optimal solution
- easy to over-fit
- robustness can be a problem, minor change could change the tree structure

2.3 Neural Networks

Neural Networks becomes popular these years. It can be very complicated and hungry for computing resources. Basically, neural networks uses different layers, including linear / non-linear layers. The neural network is a black-box model, and it receives great success nowadays.

3 Experiments: Run the codes

We use different Python packages to perform the prediction task.

Sklearn is a powerful tool packages, which contains many machine learning methods

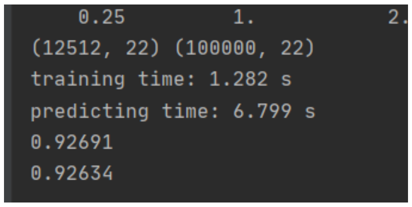
tensorflow, keras: a deep learning framework which supports GPU acceleration.

One more notice is that, when dealing with SVM, remember to normalize data first, *i.e.*

$$x_{processed} = x/\max(x)$$

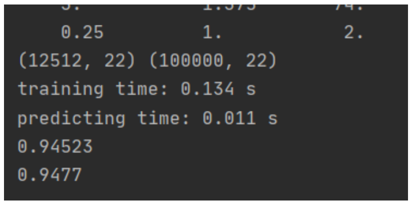
3.1 Results

output of the codes:



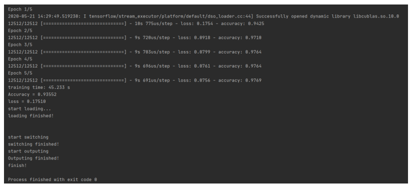
```
     0.25         1.          2.
(12512, 22) (100000, 22)
training time: 1.282 s
predicting time: 6.799 s
0.92691
0.92634
```

图 1: output for SVM



```
    0.25         1.          2.
(12512, 22) (100000, 22)
training time: 0.134 s
predicting time: 0.011 s
0.94523
0.9477
```

图 2: output for Decision Tree



```
Epoch 1/5
2020-05-21 14:29:49.519230: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcublas.so.10.0
12512/12512 [==============================] - 10s 775us/step - loss: 0.1754 - accuracy: 0.9425
Epoch 2/5
12512/12512 [==============================] - 9s 720us/step - loss: 0.0918 - accuracy: 0.9710
Epoch 3/5
12512/12512 [==============================] - 9s 703us/step - loss: 0.0799 - accuracy: 0.9764
Epoch 4/5
12512/12512 [==============================] - 9s 696us/step - loss: 0.0761 - accuracy: 0.9764
Epoch 5/5
12512/12512 [==============================] - 9s 691us/step - loss: 0.0756 - accuracy: 0.9769
training time: 45.233 s
Accuracy = 0.93552
loss = 0.17510
start loading...
loading finished!


start switching
switching finished!
start outputing
Outputing finished!
finish!

Process finished with exit code 0
```

图 3: output for Neural Networks

A comparison table is shown as below

method	training time(s)	training accuracy(%)	testing accuracy(%)	loss
SVM	1.282	92.69	92.63	/
DTree	0.134	94.52	94.77	/
NN	45.233	97.69	93.55	0.17510

4 Some thoughts

Through this lab, we can conclude that this AAR prediction task is relatively simple. The fitting process takes very short time. Instinct tells us that the complicated methods tends to have better accuracy and takes a long time. But I think the result also shows that we need to find suitable methods to specific tasks. For example, Decision tree in this task seems to be a suitable solution.

5 External Links

My github repository for this Lab

- `https://github.com/koalazf99/EE447Lab/tree/master/Lab4`