Machine Learning HW2 Report

b03901057

In this homework, I implemented two algorithms for spam classification. The first algorithm is logistic regression as required by this homework. The second approach is the artificial neural network (ANN). The remainder of the this report is structured as follows: In the section 1, the code for the logistic regression function is pasted and explained. In the section 2, the ANN approach is explained. Then in the section 3, some problems relating to this homework are discussed.

1 Logistic Regression

My code for logistic regression is shown below:

```
def train_linear(train_fea, train_tar, eta, lamb, maxIter,...) :
    dataNum = train_fea.shape[0]
    dimNum = train_fea.shape[1]
    beta = np.zeros([dimNum+1, 1]) # coefficients

    # Pad leading 1s to the beginging of training data
    train_fea = padOne(train_fea, 1)
    test_fea = padOne(test_fea, 1)

    converged = False

    while not converged :
        # f(x) = 1/(1+e^(-z)), z = wx+b
        tmp = f(train_fea, beta) - train_tar
        # Update beta by gradient descent
        beta = beta - 2*eta*np.dot(np.transpose(train_fea), tmp)\
                         + lamb * beta

def padOne(X):
    dataNum = X.shape[0]
    leadOnes = np.ones((dataNum, 1))
    X = np.concatenate((leadOnes, X), axis=dim)

def f(X, beta) :
    return sigmoid(np.dot(X, beta))

def sigmoid(X):
    return 1 / (1 + np.exp(-1*X))
```

2 Artificial Neural Network

I decided to use ANN as the second approach for its ability to fit many nonlinear functions pretty well. The details of the neural network is discussed in the following paragraph.

Strucutre

I used a 3 layer neural network (1 hidden layer), with hidden layer consists of 128 nodes to train the classifier.

Algorithm

- **Activation function** : Sigmoid. I have tried other activation functions, like tanh or ReLU. According to my experiments, sigmoid is still the best one.

- **Backpropagation** : I implemented the standard backpropagation algorithm. The code for backpropagation is omitted here to save space.

- **Gradient Descent** : After obtaining the gradients of all the weights in the backpropagation step, I have tried different ways to update the weights.
  My final method is stochastic gradient descent + mini batch technique + adam. The batch size can be chosen arbitrarily, however if the batch size is set to the whole training data, the update of the weights will become very slow. This is because the weights are only updated after seeing all the training data. On the contrary, if the batch size is too small, the optimization may converge very slowly since the weights are updated whenever the program sees a new training data. My final choice of the batch size is 50, which is decided by try and error. In addition, the adam technique is used here to adaptively decide the learning rate throughout the training process. There will be more discussions on this the section 3.

Comparison

Comparing the results of logistic regression and ANN, it can be clearly seen that ANN improves the performance, though not by a large margin. Experimental results are listed below:

Table 1: My caption

	Training Error	Validation Error	Kaggle Score	Training Iterations
Logistic	0.0635	0.0775	0.93	1000000
ANN	0.0519	0.0551	0.94	700

The training set consists of the first 90% of the data, while the rest are reserved as the validation set. Note that, ANN needs much fewer iterations than logistic regression, but ANN actually takes more time. The reason is that in ANN, each iteration consists of several times of update of weights, since the mini batch technique is used, and each update of weights are more computationally expensive than logistic regression.

3 Dicsussions

Some previous work

The spam classification problem has been studied for a long time. In general, a solution to this problem consists of two steps: feature extraction and machine learning. Though in this homework, the feature extraction work is done, it should be noticed that practically many other features are used, and proved useful. For example, the sender's address and url are useful information. Nonetheless, this homework focuses on the learning part.

There are many learning algorithm proposed, among which, the naive bayes method is the most well-known, and probably the most widely used for its simplicity and good performance. R. Kishore Kumar et al. reviewed various machine learning algorithms for spam classification, and compared their performance. The naive bayes method is treated as the baseline method, and it even beats logistic regression. Neural network, or multilayer perceptron, has good performance but still not the best one. The best algorithm is based on random tree. However, it should be noticed that even the baseline method has acceptable performance, so this is why naive bayes still enjoys high popularity.

ANN parameter update methods

Since the optimization step of ANN uses gradient descent, it will encounter the same problem as we do in gradient descent. In many situations, the naive parameter update method does not work in ANN since as the objective function becomes more complex, it might have more saddle points or flat regions. Therefore, a constant learning may not work in ANN. Some common parameter update strategy includes **momentum update**, **Nesterov Mementum**, **L-BFGS**, **Adagrad**, and **Adam**. In my homework, I tried Adam as it is perceived as the default algorithm to use, and also Nesterov Mementum, which has stronger theoretical convergence guarantees than the momentum update. The performance of the ANN adopting the two update schemes are shown in the table below:

Table 2: Adam v.s. Nesterov

Method	learning rate	Train Error	Validation Error	Kaggle Score	Iterations
Adam	1e-5	0.0519	0.0551	0.9467	700
Nesterov	5e-3	0.0499	0.0496	0.94	700

It can be seen that, Nesterov can achieve lower error rate in the same number of iterations compared to Adam, but Adam has higher Kaggle score. Therefore, these two methods are approximately the same in terms of their performance.

4 Reference

1. Kumar, R. Kishore, G. Poonkuzhali, and P. Sudhakar. ”Comparative study on email spam classifier using data mining techniques.” Proceedings of the International MultiConference of Engineers and Computer Scientists.

2. Guzella, Thiago S., and Walmir M. Caminhas. ”A review of machine learning approaches to spam filtering.” Expert Systems with Applications