PFI/PFN 2015 Intern coding questions

NOTE: This is unofficial and voluntary translation of the official Japanese question. See here for the Japanese ones: https://research.preferred.jp/2016/07/intern-coding-tasks/ .

Written by Shintaro SHIBA. Please contact me if you find anything.
shiba.shintaro@gmail.com

Use one of the following language.

- C, C++, Ruby, Python, Go, Java, Scala, Lua, Cuda, JavaScript

Evaluation Matrices

The minimum requirement is to complete the program that can solve the problem, regardless of the speed/memory usage of calculation. Additionally, programs are better if they are:

- More computationally efficient.
- Requiring less memory size.
- Based on unique ideas.
- Of good quality (think what "quality" of programs means by yourself).
- Examined about if the program is accurate.

Questions

Download the dataset from https://www.dropbox.com/s/6z6hwcywxiae6p8/data.zip?dl=0 . The format of the data is as follows, where label[i] is integer, data[i, j] is floating point, the separator of the data is space (" ": 0x20), the newline character is CR(0x0A).

```
nm
label[1] data[1, 1] data[1, 2] data[1, 3] ... data[1, m] label[2] data[2, 1]
data[2, 2] data[2, 3] ... data[1, m] ...
label[n] data[n, 1] data[n, 2] data[n, 3] ... data[n, m]
```

- train.txt : data for training. Will be used in question 1, 2, and 3.

- train_nolabel.txt: data for training without labels. Will be used in the question 3.1.
- test.txt : data for test. Will be used in the question 3.1.

We will ask you the question about two-class classification. label[i] is 1 or 2, while 0 means no label.

Q1

Please do not use the libraries except for standard ones (Assess your programming ability).

- 1-1

Implement a function that computes the average for each dimension, subtract it from the data and make the average over each dimension 0. The results will be returned in the same format as the inputs. Here is an example of void SubtractMean(vector<vector<float> >& data);

Input:

```
23
1 1.0 9.0 1.0
2 3.0 1.0 1.0
```

Output:

```
23
1 1.0 4.0 1.0
2 1.0 4.0 1.0
```

- 1-2

Implement a function that computes the standard deviation (square root of the variance) for each dimension of the output data of the Question 1-1, divide the data by it and make the variance over each dimension 1. The results will be returned in the same format as the inputs. Here is an example of void NormalizeVariance(vector<vector<float> >& data);

Input:

```
23
1 1.0 4.0 1.0
2 1.0 4.0 1.0
```

Output:

```
23
1 1.0 1.0 1.0
2 1.0 1.0 1.0
```

- 1-3

Implement a simple perceptron, which executes learning and returns the macro-saveraged accuracy of five-fold cross-validation for the given input dataset. Accuracy should be defined as {number of samples correctly classified} / {the number of the all samples}. Use train.txt for the learning and cross-validation.

```
float CrossValidate(int num_iterations, const vector<vector<float> >& data, const
vector<int>& labels);
```

This is an easy explanation of simple perceptron. Here, x is input vector of m dimension, y is label, which value is +1 or -1 for simplification. w is weight vector of m dimension, we ignore bias, and dot_product(w, x) means dot product function. Simple perceptron computes I(dot_product(w, x)) as prediction, while I(z) = +1 if z >= 0, and 1 otherwise. The learning algorithm is as follows.

```
for num_iterations times do
    for each data x and label y do
        if y * I(dot_product(w, x)) < 0 then
        w←w+yx end if
end for end for
```

Q2

Please use the libraries freely (Assess your problem-solving ability).

- 2-1

Implement a function that applies ZCA whitening for the given input data, which makes variance-covariance matrix identity matrix. The results will be returned in the same format as the inputs. void ZCAWhitening(vector<vector<float> >& data);

http://deeplearning.stanford.edu/wiki/index.php/Implementing_PCA/Whitening

- 2-2

Achieve accuracy over 87% of five-fold cross-validation, using any method for learning algorithm. If you don't have any good idea, one of recommendations are averaged perceptron or SVM + RBF kernel.

Q3 (optional)

Achieve accuracy over 95% of five-fold cross-validation, using any method for learning algorithm. You can use train_nolabel.txt. In this file, `label[]` is 0. Classify the dataset from text.txt using trained model, and output the result file, which includes of i-th result of classification (1 or 2) for i-th column.