Assignment 3: Classification

For due date please see course management system
75 marks total, graded part = 45 marks

This assignment is to be done individually.

Important Note: The university policy on academic dishonesty (cheating) will be taken very seriously in this course. You may not provide or use any solution, in whole or in part, to or by another student.

You are encouraged to discuss the concepts involved in the questions with other students. If you are in doubt as to what constitutes acceptable discussion, please ask! Further, please take advantage of office hours offered by the instructor and the TA if you are having difficulties with this assignment.

DO NOT:

- Give/receive code or proofs to/from other students
- Use Google to find solutions for assignment

DO:

- Meet with other students to discuss assignment (it is best not to take any notes during such meetings, and to re-work assignment on your own)
- Use online resources (e.g. Wikipedia) to understand the concepts needed to solve the assignment

---

Question 1 (10 marks)

Question 4.9 in PRML.

Start by writing down the log-likelihood. You then need to make use of a constraint on the π_k, using a Lagrange multiplier.

Question 2 (10 marks)

Logistic regression uses the cross-entropy error

$$E(\boldsymbol{w}) = -\sum_{n=1}^{N} (t_n \cdot ln(\sigma(\boldsymbol{w} \bullet \boldsymbol{x}_n)) + (1 - t_n) \cdot ln(1 - \sigma(\boldsymbol{w} \bullet \boldsymbol{x}_n)))$$

where

$$\sigma(y) = \frac{1}{1 + exp(-y)}$$

Show that

$$\nabla E(\boldsymbol{w}) = \sum_{n=1}^{N} (\sigma(\boldsymbol{w} \bullet \boldsymbol{x}_n) - t_n)\boldsymbol{x}_n$$

Hint: Use the fact that

$$\frac{d\sigma}{dy} = \sigma(1 - \sigma).$$

Question 3 (10 marks)

Construct by hand a neural network that computes the XOR function of two inputs. Make sure to specify what sort of units you are using, and which activation functions. You may like using the UBC Aispace neural network tool to assist in developing the neural network, and to help visualize it.

Use the representation with 1 = true, 0 = false, as shown in Table 1 below.

Table 1: The X-OR function. x_1 is the value of the first input node, x_2 is the value of the second input node, and t is the target value for the output node.

x1	x2	t
1	1	0
0	1	1
1	0	1
0	0	0

Question 4 (40 marks)

In this question you will compare 2 methods for optimization for logistic regression.

1. Download the assignment 3 code and data from the website. Run the script `logistic_regression.m` in the lr directory.

   This code performs gradient descent to find $\boldsymbol{w}$ which maximizes the likelihood (more precisely, minimizes negative log-likelihood).

   Include the final output of Figures 2 and 3 (plot of separator path in slope-intercept space; plot of neg. log likelihood over iterations) in your report.

Why are these plots oscillating? **Briefly explain why in your report.**

How might you fix this? **Fix this, and include new plots in your report and an explanation of your fix.**

2. Modify this code to use iterative reweighted least squares (IRLS, Eqn. 4.99). The built-in MATLAB function `diag` is useful for Eqn. 4.98.

   Note that this only takes about 3 lines of code to implement. If you're doing more work, stop, read the textbook, or ask me or the TA for help.

   Include new plots of Figures 2 and 3 using IRLS in your report.

   Yes, it is that fast.

Question 5 (5 marks)

In the next assignment 4 you will use support vector machines for SPAM email[1] detection. The goal is to get you started early on becoming acquainted with the data and the support software, so that you can proceed quickly to experimenting on the next assignment. Hopefully this will also prepare you for the more theoretical discussion of SVMs in class. The data are in the tarball on the website, in the spam directory.

The directories easy_ham, and spam (which are .tar.gzipped) contain email messages. These have been parsed into feature vectors for you, and stored in the .mat files. Each .mat file contains a single variable V of word counts. V is $n_{messages}$-by-d, where d is the dictionary size. For interest, the dictionary is also provided.

Download libsvm, which has a MATLAB interface on it, from: `http://www.csie.ntu.edu.tw/~cjlin/libsvm/#matlab`

Download the first link, "A simple MATLAB interface." Instructions for installation are available on the website and included in the readme. If you have problems, please ask in class or e-mail the TA. **I advise installing this early.**

I also recommend looking at the authors' user guide, "A Practical Guide to Support Vector Classification", which I have posted on the course site for your convenience. This briefly reviews some of the key points and discusses questions like how to do cross validation with SVMs.

Use a Gaussian RBF kernel with the default parameters. Do the following:

1. Use cross-validation on the easy_ham and spam as your training data to estimate the accuracy of SVM classification with these choices. Note that libsvm comes with various utility files that you may find useful. Specifically, produce the following two plots.

   (a) Include a plot that shows the training error. By this I mean the resubstitution error, which results by applying the trained classifier to the original training sets (easy_ham and spam).

[1] The data come from the SpamAssasin public mail corpus `http://spamassassin.apache.org/publiccorpus/`.

(b) Include a plot that shows the result of applying cross-validation to the training data to estimate how well the classifier generalizes.

2. Run the classifier on the unlabeled data in `test_data.mat`. This file contains a matrix V of word counts, which is 2796-by-1373. Produce an output vector v that is 2796-by-1, with the target value of -1 for spam messages, and 1 for ham (non-spam) messages.

   Save the vector v and your login to identify you in a file spamtest.mat:

```
v = ...
name = 'your login';
save('spamtest.mat','v','name');
```

Submitting Your Assignment

You should create a report with the answers to questions and figures described above in PDF format. Make sure it is clear what is shown in each figure. **DO NOT INCLUDE SOURCE CODE.**

Include the spamtest.mat file and your PDF report in an archive, and submit it.

Submit your assignment using the online assignment submission server at: `https://courses.cs.sfu.ca`.