Assignment 2: Classification

Due October 28 at 11:59pm
90 marks total

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

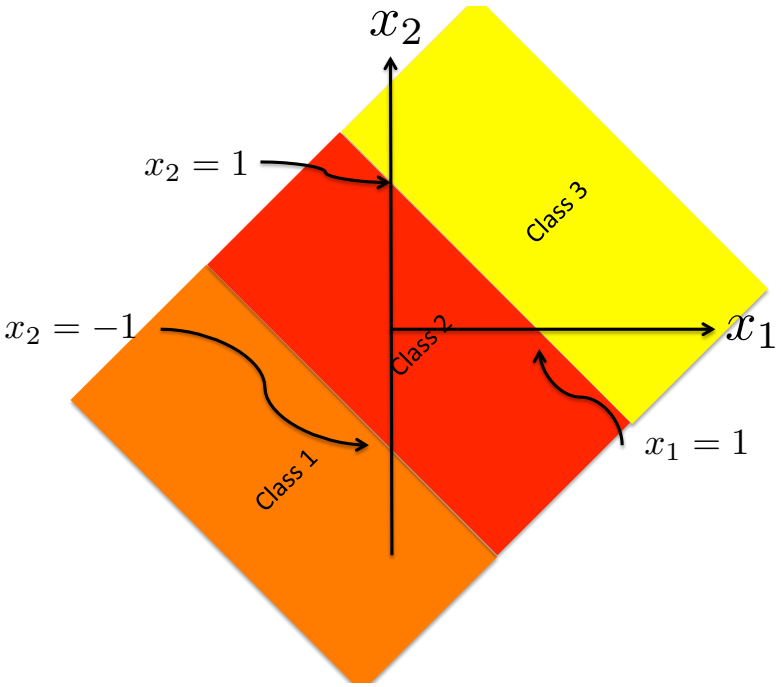
DO:

- 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

---

1 Linear Models for Classification (10 marks)

Provide a set of 3 linear functions $y_1(\boldsymbol{x}), y_2(\boldsymbol{x}), y_3(\boldsymbol{x})$ that would produce the decision regions shown in the figure below.



I.e. the decision region for class 3 contains all points with $x_1 + x_2 > 1$.

2 Kernels (20 marks)

1. Polynomial kernels (10 marks).

   In lecture we looked at $k(\boldsymbol{x}, \boldsymbol{z}) = (1 + \boldsymbol{x}^T \boldsymbol{z})^d$. This kernel function contains polynomial terms up to degree d. However, the coefficients in front of the terms vary. E.g. for $d = 2$ and a two-dimensional input, this kernel function is equivalent to the mapping $\boldsymbol{x} \mapsto (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)$.

   Consider using this kernel for regression. Do these coefficients (i.e. $\sqrt{2}$) matter? Would the resulting regression model be the same using this kernel in kernelized regression versus using a direct polynomial mapping? What if the regression uses a regularizer on the weights? Explain.

2. Combining kernels (10 marks).

   Suppose $k_a(\boldsymbol{x}, \boldsymbol{z})$ and $k_b(\boldsymbol{x}, \boldsymbol{z})$ are valid kernels corresponding to dot products in spaces given by $\boldsymbol{x} \mapsto (\phi_1^a(\boldsymbol{x}), \phi_2^a(\boldsymbol{x}), \ldots, \phi_N^a(\boldsymbol{x}))$ and $\boldsymbol{x} \mapsto (\phi_1^b(\boldsymbol{x}), \phi_2^b(\boldsymbol{x}), \ldots, \phi_M^b(\boldsymbol{x}))$ respectively.

   Show that $k_c(\boldsymbol{x}, \boldsymbol{z}) = \alpha k_a(\boldsymbol{x}, \boldsymbol{z}) + \beta k_b(\boldsymbol{x}, \boldsymbol{z})$, where $\alpha, \beta > 0$, is also a valid kernel. Do this by explicitly constructing the space in which $k_c(\boldsymbol{x}, \boldsymbol{z})$ corresponds to a dot product.

3 Logistic Regression (40 marks)

In this question you will examine optimization for logistic regression.

1. Download the assignment 2 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 minimizes negative log-likelihood (i.e. maximizes 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.

2. Create a MATLAB script `logistic_regression_mod.m` for the following.

   Modify `logistic_regression.m` to run gradient descent with the learning rates $\eta = 0.005, 0.003, 0.001, 0.0005, 0.0001$.

   Include in your report a single plot comparing negative log-likelihood versus iteration for these different learning rates.[1]

   Compare these results. What are the relative advantages of the different rates?

3. Create a MATLAB script `logistic_regression_sg.m` for the following.

   Modify this code to do stochastic gradient descent. Use the parameters $\eta = 0.1, 0.05, 0.03, 0.02, 0.01, 0.001, 0.0001$.

   Include in your report a new plot comparing negative log-likelihood versus iteration using stochastic gradient descent.

   Is stochastic gradient descent faster than gradient descent? Explain using your plots.

4. Create a MATLAB script `logistic_regression_irls.m` for the following.

   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 TAs for help.

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

   Yes, it is that fast.

Still need to insert plots

[1] `legend(cellstr(num2str(etas')));` produces a nice legend if `etas` is a vector of learning rates.

4 Kernelized Perception (20 marks)

In this question you will implement kernelized perceptron and use it for SPAM email[2] detection. 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 `email.mat`. This .mat file contains `Ftrain`, word counts for each email. `Ftrain` is $n_{messages}$-by-d, where d is the dictionary size. For interest, the dictionary is also provided. `Ltrain` contains labels (0-1 encoding).

Complete the implementation of kernelized perceptron learning in `do_kernel_perceptron.m`. Fill in the TO DO blanks.

Experiment with different kernels using cross-validation on `email.mat` as your training data.

Choose what you think is the best classifier, then run it on the unlabeled data in `test.mat`. This file contains a matrix `Ftest` of word counts, which is 2796-by-1373. Produce an output vector `Fn` that is 2796-by-1, with the target value of -1 for spam messages, and 1 for ham (non-spam) messages.

Save the vector `Fn` and your SFU email to identify you in a file spamtest.mat:

```
Fn = ...
email = 'sfuid@sfu.ca';
save('spamtest.mat','Fn','email');
```

Describe the kernels with which you experiemented in your report, and give their cross-validation errors. State which kernel/parameter values you used for producing the `Fn` you submitted.

Bonus marks and a prize will be given to the student(s) with the best classification performance!

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

Submitting Your Assignment

The assignment must be submitted online at `https://courses.cs.sfu.ca`. You must submit three files:

1. An assignment report in **PDF format**, called `report.pdf`. This report must contain the solutions to questions 1-2 as well as the figures / explanations requested for 3-4.
2. A .zip file of all your code, called `code.zip`. This must contain the directories `lr` and `spam` (no leading path names), in which all of your files must appear[3].
3. spamtest.mat for question 4.

[3]This includes the data files and others which are provided as part of the assignment.