Midterm Test

Name:

Student Number:

Closed book. 1-page, single-sided cheat sheet allowed (8.5x11in, no more than 6000 characters, 12pt font or larger if typed). No other aids allowed.

50 minutes. 8 pages. 4 questions. 76 points. Write all answers on the test booklet, using the backs of pages if necessary. The last page is blank. Clear, concise answers will receive more marks than long, rambling ones. Unless specified otherwise, all answers should be justified. Good luck!

I don’t know policy: If you do not know the answer to a question (or part), and you write “I don’t know”, you will receive 20% of the marks of that question (or part). If you just leave a question blank with no such statement, you will get 0 marks for that question.
1. (20 points total) True or False.

For each of the following statements, state whether it is true or false without giving an explanation (2 points each):

(a) If a function overfits data, then the training error is too small and the test error is too large.

(b) In regularized linear regression, the regularization term increases the size of the weights, $w_1, \ldots, w_m$.

(c) Cross validation produces a better estimate of validation error than simple validation.

(d) In regularized linear regression, the optimal solution minimizes the validation error.

(e) Logistic regression is better than nearest neighbours at fitting multi-modal data.

(f) Logistic regression is a good choice when the data are linearly separable.

(g) In multi-class logistic regression, we learn a linear function for each class.

(h) Learning to recognize hand-written digits is an example of binary classification.

(i) Linear least-squares regression tries to find values of $w$ and $w_0$ that minimize the value of $\sum_n [y^{(n)} - (w^T x^{(n)} + w_0)^2]$ where the sum is over the training data.

(j) In nearest neighbours, learning from training data is slow, but classification of test data is fast.
2. (18 points total) Numpy Programming

Suppose A is a 100 × 1000 matrix, x is a 100-dimensional (rank 2) column vector, and y is a 1000-dimensional (rank 1) vector. Solve each problem below using a single line of Python code and no loops (3 points each). You may assume that your program begins with the following line:

```python
import numpy as np
```

(a) Reshape A into a 10 × 100 × 100 array. Call the result B. A itself does not change.

(b) Add x to each column of A. Call the result C. A itself does not change.

(c) Add y to row 3 of A and assign the result to row 37 of A. (Only row 37 changes.)

(d) Compute the sum of each row of A.

(e) Print rows 12, 22, 32 and 42 of matrix A.

(f) Compute the logarithm of each element of $A^T \ast A \ast y$, where $\ast$ denotes matrix multiplication, and $^T$ denotes matrix transpose.
3. (19 points total) *Linear and Logistic Regression.*

This question includes two figures, each displaying a set of 1-dimensional training data. Each point in a figure represents a training pair \((x, t)\), where \(x\) is the input value and \(t\) is the target value. \(x\) is a real number, and \(t\) is either 0 or 1.

(a) (6 points) Figure 1 shows a training set. Draw a plausible function that would be fitted to the data by linear least-squares regression. Also, draw a plausible function that would be fitted to the data by logistic regression. Clearly identify each function.

(b) (2 points) If we use these functions for binary classification, indicate the resulting decision boundaries on the figure. Clearly identify each boundary.
(c) (6 points) Figure 2 shows a training set. Draw a plausible function that would be fitted to the data by linear least-squares regression. Also, draw a plausible function that would be fitted to the data by logistic regression. Clearly identify each function.

(d) (2 points) If we use these functions for binary classification, indicate the resulting decision boundaries on the figure. Clearly identify each boundary.

(e) (3 points) In 20 words or less, what are the implications of parts (a) to (d) when using linear regression and logistic regression for classification? (Only the first 20 words of your answer will be graded.)
4. (19 points total) We are given training data \((x^{(1)}, t^{(1)}),..., (x^{(N)}, t^{(N)})\) where each input, \(x^{(n)}\), is a vector, and each target value, \(t^{(n)}\), is a real number. In addition, our learning problem has the following loss function:

\[
l(w) = \frac{1}{2} \sum_n [t^{(n)} - w^T x^{(n)}]^2
\]

where the sum is over the training data, and \(w\) is a weight vector we are trying to learn. To minimize the loss, it is convenient to let \(X\) be a matrix whose \(n\)th row is \(x^{(n)}\), and let \(T\) be a column vector whose \(n\)th element is \(t^{(n)}\). Prove each of the following equations:

(a) (3 points) \(w^T x^{(n)} = [Xw]_n\)

(b) (4 points) \(\partial l(w)/\partial w_m = - \sum_n [t^{(n)} - w^T x^{(n)}] x^{(n)}_m\)
(c) (12 points) \[ \frac{\partial l(w)}{\partial w} = X^T(Xw - T) \]
This page is for answers and rough work