A11385W1

# DEGREE OF MASTER OF SCIENCE

**Computer Science**

---

# MACHINE LEARNING

---

**HILARY TERM 2017**

**Friday 13th January, 9:30 am – 12:30 pm**

*Candidates should answer all questions.*
*Please start the answer to each question on a new page.*

Do **not** turn over until told that you may do so.

## Question 1

Consider the following data:

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 1 \\ 3 & 8 & 2 \\ 1 & 4 & 2 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 800 \\ 125 \\ -450 \end{bmatrix}, \quad \mathbf{\Gamma} = \begin{bmatrix} 1 & 20 & 0 \\ 1 & 10 & 1 \\ 0 & 10 & 1 \end{bmatrix}.$$

Here, each row of $\mathbf{X}$ corresponds to a data point, represented by three features, the vector $\mathbf{y}$ represents the three output values and $\mathbf{\Gamma}$ is a regularisation operator. You want to carry out regression on the given data, i.e. compute a column vector parameter $\mathbf{w} \in \mathbb{R}^3$. The goal is to minimise the function $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \|\mathbf{\Gamma}\mathbf{w}\|^2$ .

(a) You begin by solving the optimisation problem.

(i) Write the optimisation objective function in terms of addition, subtraction, multiplication and the transpose of matrices and vectors. (2 marks)

(ii) Derive a closed-form formula for $\mathbf{w}$ in terms of $\mathbf{X}$, $\mathbf{y}$ and $\mathbf{\Gamma}$. You do not have to perform the actual computation to get numerical values for the entries of $\mathbf{w}$. (6 marks)

(b) Give the formula for predicting a new output $y_{(\text{new})} \in \mathbb{R}$, given $\mathbf{w}$ and a new input data point represented by a column vector $\mathbf{x}_{(\text{new})} \in \mathbb{R}^3$. (2 marks)

(c) For the problem above, the optimal $\mathbf{w}$ is given as follows.

$$\mathbf{w} \approx \begin{bmatrix} 169.82 \\ -8.53 \\ -89.82 \end{bmatrix}$$

(i) What can you say about the relative importance of features 2 and 3 for the prediction? Justify your answer. (5 marks)

(ii) How would you modify the algorithm computing $\mathbf{w}$ in order to be able to say more? (5 marks)

(d) Assume that the matrix $\mathbf{X}$ is very big (i.e. of size $10000000 \times 1000$), the vector $\mathbf{y}$ is correspondingly long, and $\mathbf{\Gamma}$ is of size $10 \times 1000$. You seek to get an answer quickly. State the drawback of simply using an implementation of the formula from point (a). Provide a short description of a different algorithm that avoids these drawbacks. (10 marks)

## Question 2

You are designing an image classification system based on neural networks. The task of the system is to read a $64 \times 64$ pixel greyscale image and return a single probability that a person is somewhere in the image. The network is trained in a supervised fashion. The pixel values are standardised to lie in the range $[0, 1]$. The outputs in the training set are zeros and ones.

(a) First, you design the network output.

(i) Describe the output layer of the network and specify the non-linearity it uses. (2 marks)

(ii) State the cross entropy loss function in terms of one particular output from the final layer $o$ and the training label $y \in \{0, 1\}$. (3 marks)

We will now consider possible architectures for the network.

(b) Assume that the network consists of an input layer, 4 fully-connected hidden layers with as many neurons each as the input layer; this is followed by the fully-connected output layer. All layers except the output use a rectified linear unit nonlinearity.

(i) How many distinct parameters are there in the network (one parameter is one number)? (2 marks)

(ii) Assume we change the training data by flipping all images upside down, then train. Would the network still work? Would the performance of the network be similar to the network trained with original data? (3 marks)

(c) Assume that the network consists of an input layer, 4 convolutional hidden layers with a $5 \times 5$ kernel, a stride of 1 and zero-padding of 4; this is followed by the fully-connected output layer. The biases are shared in the hidden units within one layer. All hidden layers use a rectified linear unit nonlinearity.

(i) How many distinct parameters are there in the network (one parameter is one number)? (3 marks)

(ii) Assume we change the training data by flipping all images upside down, then train. Would the network still work? Would the performance of the network be similar to the network trained with original data? (3 marks)

(iii) Assume we train the network using the original images, but then test it using images flipped upside down. Would you expect the network to still work? (3 marks)

(d) Compare the two networks proposed above and answer the following questions.

(i) Which one will require more examples to train? (3 marks)

(ii) For which one will back-propagation be more expensive, other things being equal? By what factor? (5 marks)

(iii) Which network makes more prior assumptions about the data? Describe the meaning of these assumptions. (5 marks)

(e) Typically, when a computer system allocates new memory to a program, it is initialised to be all zeros. When initialising weights in the two networks described above, would leaving them at zero be a good choice? If not, what would you change? Justify your answer. (8 marks)

## Question 3

You are building an algorithm to provide a map for regions in the brain. You can assume that those regions in the brain have complicated shapes. Your input consists of a set of voxels (points in 3D space). The representation of each voxel consists of a vector of 100 real numbers, which correspond to measurements performed in this voxel at successive times. No other data is available. You can assume that a brain map is just a clustering of voxels.

(a) First, consider the approach where you treat each voxel as a vector in $\mathbb{R}^{100}$ and perform k-means clustering. How well do you think this approach would work? (5 marks)

(b) Assume an expert has provided you with a function $d$ which tells you the difference between two vectors corresponding to the voxels. You are only allowed to use this function and must not interact with the voxels in any other way. You decide to perform spectral clustering on the data. Would you expect it to work better or worse than the algorithm from part (a)? How does the fact that you have no direct access to the voxels, but only to evaluations of the function $d$, reflect on the quality of the clustering? (8 marks)

(c) Consider an undirected weighted graph where the weight of edge $(i, j)$ is given by $w_{ij} = w_{ji} \geq 0$. A weight of zero means there is no edge. Consider the symmetric weight matrix $\mathbf{W}$ and the diagonal degree matrix $\mathbf{D}$ defined by

$$\{\mathbf{W}\}_{ij} = w_{ij} = w_{ji}, \quad \{\mathbf{D}\}_{ii} = \sum_j w_{ij}.$$

Consider the Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Consider a real function $f$ of the vertices of the graph, represented as a vector $\mathbf{f}$ containing the evaluation of the function at each vertex. Assume that the graph consists of exactly two connected components $\mathcal{G}_1$ and $\mathcal{G}_2$. You can assume that the graph has more than three vertices.

(i) Consider the functional $g(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}\mathbf{f}$. Assume that $\mathbf{f}_1$ is constant on all vertices and $\mathbf{f}_2$ has a value of 1 over $\mathcal{G}_1$ and 10 over over $\mathcal{G}_2$. Consider $g(\mathbf{f}_1)$ and $g(\mathbf{f}_2)$. Can you say that one is larger than the other? Which one? Justify your answer. (5 marks)

(ii) Give a closed-form expression for $\min_{\mathbf{f}\neq\mathbf{0}} \frac{\mathbf{f}^\top \mathbf{L}\mathbf{f}}{\|\mathbf{f}\|^2}$. Justify your answer. (5 marks)

(iii) Consider the following variant of the optimisation above:

$$\mathbf{v}_1 = \arg\min_{\mathbf{f}\neq\mathbf{0}} \frac{\mathbf{f}^\top \mathbf{L}\mathbf{f}}{\|\mathbf{f}\|^2}.$$

What is the connection between $\mathbf{v}_1$ and the spectral clustering algorithm? (4 marks)

(iv) What constraint would you add to the optimisation above to get another feature? (3 marks)