Version MJFG/3

EGT3
ENGINEERING TRIPOS PART IIB

Monday 23 April 2018 9.30 to 11.10

Module 4F10

DEEP LEARNING AND STRUCTURED DATA

*Answer not more than **three** questions.*

All questions carry the same number of marks.

*The **approximate** percentage of marks allocated to each part of a question is indicated in the right margin.*

*Write your candidate number **not** your name on the cover sheet.*

STATIONERY REQUIREMENTS
Single-sided script paper

SPECIAL REQUIREMENTS TO BE SUPPLIED FOR THIS EXAM
CUED approved calculator allowed
Supplementary page: two extra copies of Fig. 1 (Questions 3 (c)(i) and 3 (c)(ii))
Engineering Data Book

10 minutes reading time is allowed for this paper at the start of the exam.

You may not start to read the questions printed on the subsequent pages of this question paper until instructed to do so.

1 A deep neural network (multi-layer perceptron) is to be trained for a K-class problem, with a d-dimensional observation vector for each of the training samples. The activation function for the output layer is a *softmax* activation function. Sigmoid activation functions are used for all L hidden layers. The number of nodes for layer l is $N^{(l)}$. Thus for hidden layer l of the network

$$\mathbf{z}^{(l)} = \mathbf{W}^{(l)}\mathbf{x}^{(l)}; \quad \mathbf{x}^{(l+1)} = \phi\left(\mathbf{z}^{(l)}\right); \quad \phi(\mathbf{z}^{(l)}) = \begin{bmatrix} 1/(1+\exp(-z_1^{(l)})) \\ \vdots \\ 1/(1+\exp(-z_{N^{(l)}}^{(l)})) \end{bmatrix}$$

(a) How many parameters does the network have? How would this influence your selection of L and $N^{(l)}$? [15%]

(b) The network is to be trained using gradient-descent with supervised training data $\mathscr{D} = \{\{\mathbf{x}_1, y_1\}, \ldots, \{\mathbf{x}_n, y_n\}\}$ where $\mathbf{x}_i$ is the observation vector for the i-th sample and $y_i \in \{\omega_1, \ldots, \omega_K\}$. The model is to be trained using a cross-entropy cost function.

(i) Give an expression for the cross-entropy training criterion that can be used in this case. You should clearly define all the terms in the expression. [15%]

(ii) Why is a softmax activation function a sensible choice for this task and training criterion? Include the form of the activation function in your answer. [10%]

(iii) For a single layer l of the network derive the form of the derivative matrix $\frac{\partial \mathbf{x}^{(l+1)}}{\partial \mathbf{x}^{(l)}}$. The matrix should be ordered so that element (i,j) is $\partial x_j^{(l+1)}/\partial x_i^{(l)}$ [25%]

(c) It is decided to make the network very deep, but restrict all the hidden layers to have the same number of nodes, the same weights, $\mathbf{W}^{(l)} = \mathbf{W}$, and have linear activation functions, $\phi(\mathbf{z}) = \mathbf{z}$. The derivative for the error function, $\mathscr{E}$, with respect to the input to layer l can be expressed as

$$\frac{\partial \mathscr{E}}{\partial \mathbf{x}^{(l)}} = \left(\prod_{j=l}^{L} \frac{\partial \mathbf{x}^{(j+1)}}{\partial \mathbf{x}^{(j)}}\right) \frac{\partial \mathscr{E}}{\partial \mathbf{x}^{(L+1)}}$$

(i) Show that the following form of approximation can be derived when $L >> l$

$$\frac{\partial \mathscr{E}}{\partial \mathbf{x}^{(l)}} \approx \eta^{L-l+1}\mathbf{A}\frac{\partial \mathscr{E}}{\partial \mathbf{x}^{(L+1)}}$$

What are the scalar η and the matrix $\mathbf{A}$? Note $\mathbf{A}$ is not a function of l. You may assume that $\mathbf{W}$ is of full rank. [25%]

(ii) Discuss the implications of Part (c)(i) for training deep neural networks. [10%]

2 A set of linear classifiers, where each classifier has the form

$$y(x) = ax$$

is to be trained for a two-class problem. The d-dimensional features for the two classes, ω_1 and ω_2, are known to be Gaussian distributed and the dimensions uncorrelated. Initially, a dimension is selected to train one classifier. For this dimension class ω_1 has a mean of 0 and variance of 2, and class ω_2 a mean of 2 and variance of 4. The classes are known to have equal priors. There are N training examples equally split between the two classes.

(a) What is the general form of Bayes' decision rule for a two class problem? [10%]

(b) The linear classifier is to be trained using least squares estimation with target values of 0 for class ω_1 and 1 for class ω_2. For N samples this criterion has the form

$$E(a) = \frac{1}{N}\sum_{i=1}^{N}\left(y_i(ax_i)^2 + (1-y_i)(1-ax_i)^2\right)$$

where y_i is 1 if the observation belongs to class ω_1 and 0 if it belongs to class ω_2. A very large number of training examples, $N \to \infty$, are available to estimate the classifier parameter. Calculate the optimal value of the classifier parameter, a, using this criterion. [30%]

(c) A threshold of 0.5 on $y(x)$ is used to classify the data. Using the value of a estimated in Part (b) calculate the probability of misclassifying a sample, P_{e}, in terms of the Gaussian cumulative density function $F(x)$ where

$$F(x) = \int_{-\infty}^{x} \mathcal{N}(z;0,1)dz$$ [25%]

(d) Two additional classifiers are constructed using the same training criterion with data from different dimensions of the feature vector. These classifiers yield individual probability of errors $P_{\mathsf{e}}^{(2)}$ and $P_{\mathsf{e}}^{(3)}$.

(i) Derive an expression for the probability of error when combining all three classifiers in terms of the probability of errors of the individual systems. [15%]

(ii) Discuss whether you expect the probability of error obtained in Part (d)(i) to be achieved with these classifiers in practice. Justify your answer. [10%]

(iii) How could a classifier with the lowest probability of error be obtained for this task? [10%]

3 (a) Describe the concept of kernel functions, indicating why they are useful for a range of classification tasks. What is the role of slack variables in soft-margin support vector machines (SVMs)? [15%]

(b) A soft-margin SVM $y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b$ is trained on the dataset $\{(\mathbf{x}_n, t_n)\}_{n=1}^{N}$ with $\mathbf{x}_n \in \mathbb{R}^d$ and $t_n \in \{-1, 1\}$. Let ξ_n be the slack variable for the n-th data point. Let $\{\xi_n^\star\}_{n=1}^{N}$, $\mathbf{w}_\star$ and $b_\star$ denote the soft-margin solution obtained using the objective function $\frac{1}{2}||\mathbf{w}||^2 + C\sum_{n=1}^{N}\xi_n$ with penalty constant C.

(i) What constraints must be satisfied by the soft-margin solution? [10%]

(ii) What is the value of $\xi_n^\star$ whenever $(\mathbf{w}_\star^{\mathrm{T}}\mathbf{x}_n + b_\star)t_n \geq 1$? Justify your answer. [10%]

(iii) What is the value of $\xi_n^\star$ whenever $(\mathbf{w}_\star^{\mathrm{T}}\mathbf{x}_n + b_\star)t_n < 1$? Justify your answer. [10%]

(c) The classification dataset is shown in Fig. 1. Assume that you train a soft-margin SVM with a linear kernel and penalty constant C on the value of the slack variables.

(i) Draw, on the attached copy of Fig. 1 (left-hand-side one), the decision border obtained when $C \to \infty$. Highlight with a circle the data-points which are support vectors. Justify your answer. [10%]

(ii) Draw, on the attached copy of Fig. 1 (right-hand-side one), the decision border obtained when $C \approx 0$. Justify your answer. [10%]

(iii) Discuss under which scenarios the values of C given in Parts (c)(i) and (c)(ii) should be used. [10%]

Two copies of Fig. 1 are attached to the back of this paper. These should be detached and handed in with your answers.

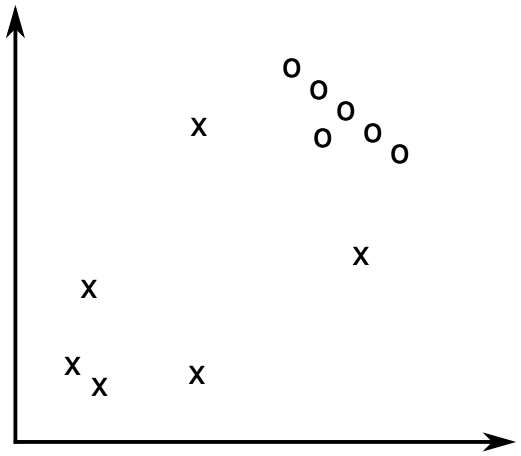


Fig. 1

(cont.

(d) We define the function $\ell(x) = [1 - x]_+$, $x \in \mathbb{R}$, where $[\cdot]_+$ returns the positive part of its argument, that is, $[z]_+ = z$ if $z > 0$ and $[z]_+ = 0$ if $z < 0$.

(i) Assume you want to solve the same soft max-margin problem as in Part (b). Use $\ell(x)$ and the solutions to Parts (b)(ii) and (b)(iii) to re-write the soft max-margin objective into a new objective function that does not include $\xi_1, \ldots, \xi_N$, only $\mathbf{w}$ and b, with solution again given by $\mathbf{w}_\star$ and $b_\star$ as in Part (b). [15%]

(ii) Would you be able to find $\mathbf{w}_\star$ and $b_\star$ by optimizing the new objective function obtained in Part (d)(i) by gradient descent? Justify your answer. [10%]

4 A hidden Markov model (HMM) is to be used to model a sequence of T, d-dimensional, vectors, $\mathbf{x}_1,\ldots,\mathbf{x}_T$. The HMM comprises 2 non-emitting states, $\mathtt{s}_1$ and $\mathtt{s}_N$, and $N-2$ emitting states, $\mathtt{s}_2,\ldots,\mathtt{s}_{N-1}$. The model always starts in non-emitting state $\mathtt{s}_1$ and finishes in state $\mathtt{s}_N$.

(a) Briefly describe the conditional independence assumptions for a HMM, and the associated graphical model. [15%]

(b) Write an expression for the log-likelihood of the sequence, $\mathbf{x}_1,\ldots,\mathbf{x}_T$, in terms of the state transition matrix $\mathbf{A}$, and the state output probability distributions. [15%]

(c) The transition probabilities (and model structure) are assumed known, but the state output distributions are to be trained using Maximum Likelihood (ML). The auxiliary function (considering only a single sequence) to estimate the parameters using Expectation Maximisation (EM) can be expressed as

$$\mathscr{Q}(\theta,\hat{\theta}) = C + \sum_{t=1}^{T}\sum_{j=2}^{N-1} P(q_t=\mathtt{s}_j|\mathbf{x}_1,\ldots,\mathbf{x}_T;\theta)\log\left(p(\mathbf{x}_t|\mathtt{s}_j;\hat{\theta})\right)$$

where C is independent of the values of the parameters to be estimated. The following variables are to be used in the EM process

$$\alpha_j(t) = \log\left(p(\mathbf{x}_1,\ldots,\mathbf{x}_t,q_t=\mathtt{s}_j;\theta)\right);\quad \beta_j(t) = \log\left(p(\mathbf{x}_{t+1},\ldots,\mathbf{x}_T|q_t=\mathtt{s}_j;\theta)\right)$$

(i) Briefly describe the meaning of each of the terms in the auxiliary function above, and how this expression can be used to find the parameter values. [15%]

(ii) How can $\alpha_j(t)$ and $\beta_j(t)$ be used to find $P(q_t=\mathtt{s}_j|\mathbf{x}_1,\ldots,\mathbf{x}_T;\theta)$? [15%]

(iii) If the output probability distribution has the form

$$p(\mathbf{x}_t|\mathtt{s}_j;\theta) = \mathscr{N}(\mathbf{x}_t;\mu_j,\Sigma_j)$$

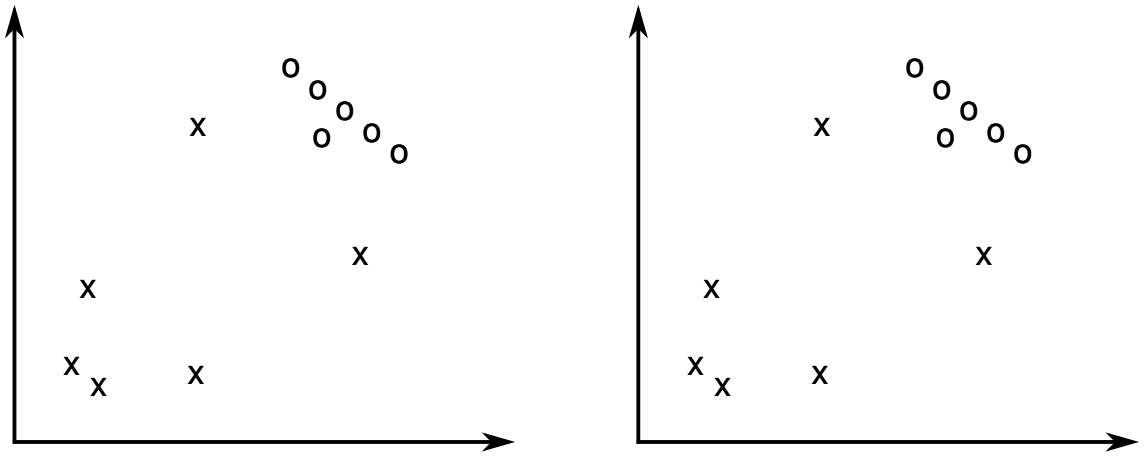
where μ_j and Σ_j are the mean and diagonal covariance matrix for state $\mathtt{s}_j$, derive the update formula for the mean of the Gaussian for state $\mathtt{s}_j$. [25%]

(d) The set of output probability distributions of Part (c)(iii) is to be replaced by a single deep neural network that predicts the mean vector and covariance matrix for each of the output distributions. Briefly discuss the topology of the network, focusing on the input and the output, that could be used in this situation. Do you expect this form of model to perform better than the output distribution in Part (c)(iii)? [15%]

END OF PAPER

Version MJFG/3 Candidate Number:

EGT3
ENGINEERING TRIPOS PART IIB
Monday 23 April 2018, Module 4F10, Question 3.



Extra copies of Fig. 1. Left, copy for Question 3 (c)(i). Right, copy for Question 3 (c)(ii).