UNIVERSITY COLLEGE LONDON

EXAMINATION FOR INTERNAL STUDENTS

MODULE CODE	:	**COMPM090**
ASSESSMENT PATTERN	:	**COMPM090B**
MODULE NAME	:	**Applied Machine Learning (Masters Level)**
DATE	:	**19 May 2016**
TIME	:	**2:30 pm**
TIME ALLOWED	:	**2 hours 30 mins**

This paper is suitable for candidates who attended classes for this module in the following academic year(s):

2015/16

TURN OVER

Applied Machine Learning, GI09, 2015-2016

Answer all THREE questions.

Marks for each part of each question are indicated in square brackets

Calculators are NOT permitted

1. a. Describe Forward Automatic Differentiation (AutoDiff) and give two procedures (one exact and the other an approximation) that compute the gradient of a subroutine $f(\mathbf{x})$ with respect to its arguments $\mathbf{x}$, giving time complexities of the approaches.

[5 marks]

b. Describe Reverse AutoDiff and explain its time complexity.

[7 marks]

c. Explain how to use Reverse AutoDiff to efficiently calculate the gradient with respect to θ_1, θ_2 of

$$\sum_{n=1}^{N} (y^n - sin(\theta_1 + \theta_2 x^n))^2$$

where (x^n, y^n) are the input-output values for the n^{th} datapoint. Your computation graph should have nodes representing elementary functions. Annotate your graph suitably and define the forward and backward passes explicitly.

[8 marks]

d. Consider a time series prediction problem in which, given a sequence of inputs $x_1, x_2, \ldots, x_t$, we make a prediction $\tilde{y}_t$ for the output at time t. To do this we define:

$$
\begin{aligned}
h_1 &= x_1 \\
h_t &= f(x_t, h_{t-1}, A) \qquad t > 1 \\
\tilde{y}_t &= g(h_t, B)
\end{aligned}
$$

where A and B are parameters and f and g are some (unspecified) functions. The objective is to find parameters A and B that minimise the loss

$$\sum_{t=1}^{T} (y_t - \tilde{y}_t)^2$$

Explain how to use Reverse AutoDiff to efficiently calculate the gradient of this loss function with respect to A and B.

[7 marks]

e. An input-output time-series (x_t, y_t), $t = 1, \ldots, T$ can be modelled by a recurrent LSTM (Long Short Term Memory) network. Explain the essential components of an LSTM network and what difficulties it tries to overcome (compared to standard recurrent networks).

[10 marks]

[Total 37 marks]

2. Principal Components Analysis (PCA) is a method to form a lower dimensional representation of data. For datapoints $\mathbf{x}^n, n = 1, \ldots, N$, define the matrix

$$\mathbf{X} = \left[\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^N\right]$$

That is, for datapoints $\mathbf{x}$ with dimension D, then $\mathbf{X}$ is $D \times N$ dimensional. The data is such that the mean is zero, that is

$$\sum_{n=1}^{N} \mathbf{x}^n = \mathbf{0}$$

K-dimensional PCA aims to find a representation

$$\mathbf{x}^n \approx \sum_{k=1}^{K} y_k^n \mathbf{b}^k$$

where $\mathbf{b}^1, \ldots, \mathbf{b}^K$ are 'basis' vectors and y_k^n are the coefficients.

a. Explain how to efficiently compute the basis vectors and coefficients in order to minimise the squared loss between the approximation and each $\mathbf{x}^n$, namely

$$\sum_{n=1}^{N} \left(\mathbf{x}^n - \sum_{k=1}^{K} y_k^n \mathbf{b}^k\right)^2$$

[8 marks]

b. Explain how Autoencoders can also be used to find low dimensional representations of data and explain how PCA relates to an Autoencoder.

[5 marks]

c. Consider an Autoencoder with structure $\mathbf{x} \to \mathbf{h} \to \tilde{\mathbf{x}}$, trained to minimise the squared loss

$$\sum_{n=1}^{N} (\tilde{\mathbf{x}}^n - \mathbf{x}^n)^2$$

with $\mathbf{h}^n = f(\mathbf{A}\mathbf{x}^n)$ and $\tilde{\mathbf{x}}^n = \mathbf{B}\mathbf{h}^n$ for matrices $\mathbf{A}$, $\mathbf{B}$ and a non-linear function f.

For K-dimensional $\mathbf{h}$, is this non-linear procedure in principle more powerful than K-dimensional PCA, in the sense that it has a lower squared loss? Explain fully your answer.

[6 marks]

d. For N datapoints $\mathbf{x}^1, \ldots, \mathbf{x}^N$, explain how it is possible to obtain essentially perfect reconstructions of these datapoints using an Autoencoder with N units in the bottleneck layer.

[3 marks]

e. When training a deep Autoencoder (say more than 8 layers) explain why it is important to initialise the parameters of Autoencoders carefully. Suggest a criterion to initialise the parameters and explain the motivation behind this approach.

[5 marks]

f. PCA can be considered a form of matrix factorisation. An alternative matrix factorisation method is probabilistic latent semantic analysis (PLSA) (also called non-negative matrix factorisation). This takes a positive matrix $\mathbf{X}$ whose entries all sum to 1:

$$\sum_{ij} X_{ij} = 1, \qquad 0 \leq X_{ij} \leq 1$$

and forms an approximation based on

$$X_{ij} \approx \sum_{k=1}^{H} U_{ik} V_{kj}$$

for matrices U and V non-negative entries and $\sum_i U_{ik} = 1$ and $\sum_k V_{kj} = 1$.

i. In the lectures, we compared the application of PCA and PLSA on a set of face images. Explain what are the typical characteristics of the 'eigenfaces' compared with the 'plsa' faces.

[4 marks]

ii. Derive an algorithm to find U and V based on an interpretation of X, U and V in terms of probability distributions.

[8 marks]

[Total 39 marks]

3. a. For input-output training points $(\mathbf{x}^n, y^n)$, $n = 1, \ldots, N$, where each input $\mathbf{x}^n$ is a vector and each output y^n is a scalar, the squared loss of a linear regression model is

$$E(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} \left(y^n - \mathbf{w}^\mathsf{T} \mathbf{x}^n \right)^2$$

i. Compute the gradient and Hessian of this objective function and show that $E(\mathbf{w})$ is convex.

[4 marks]

ii. Explain what Stochastic Gradient Descent is and how it could be used to find the $\mathbf{w}$ that minimises $E(\mathbf{w})$.

[4 marks]

iii. In the case that the input vectors are sparse (only a fraction f of the elements of each $\mathbf{x}^n$ are non-zero), explain what computational savings this has when implementing gradient descent.

[4 marks]

iv. Explain how Conjugate Gradients could be used to find the $\mathbf{w}$ that minimises $E(\mathbf{w})$ and what computational savings can be made when the input vectors $\mathbf{x}^n$ are sparse.

[4 marks]

b. Consider a multi-class classification problem with input vector $\mathbf{x}^n$ and corresponding class label $c^n \in \{1, \ldots, C\}$. The softmax log likelihood objective is to maximise

$$L(\mathbf{w}_1, \ldots, \mathbf{w}_C) \equiv \sum_{n=1,\ldots,N} \log p(c^n | \mathbf{x}^n)$$

where

$$p(c^n | \mathbf{x}^n) = \frac{e^{\mathbf{w}_{c^n}^\mathsf{T} \mathbf{x}^n}}{\sum_{c=1}^{C} e^{\mathbf{w}_c^\mathsf{T} \mathbf{x}^n}}$$

i. Calculate the gradient vectors

$$\frac{\partial}{\partial \mathbf{w}_c} L$$

[4 marks]

ii. Show that $L(\mathbf{w}_1, \ldots, \mathbf{w}_C)$ is jointly concave.

[4 marks]

[Total 24 marks]

END OF PAPER