IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2023

MEng Honours Degree in Electronic and Information Engineering Part IV
MEng Honours Degree in Mathematics and Computer Science Part IV
MEng Honours Degrees in Computing Part IV
MSc in Advanced Computing
MSc in Artifical Intelligence
MSc in Computing Science (Specialist)
MRes in Artificial Intelligence and Machine Learning
for Internal Students of the Imperial College of Science, Technology and Medicine

*This paper is also taken for the relevant examinations for the*
*Associateship of the City and Guilds of London Institute*

# PAPER COMP70010=COMP97111

# DEEP LEARNING

Friday 24th March 2023, 10:00
Duration: 90 minutes

*Answer ALL TWO questions*

Paper contains 2 questions
Calculators required

1 Convolutions and Networks

a When training a vast feedforward neural network with 100 layers for a binary classification challenge, you employ a sigmoid activation for the last layer and combine *tanh* and *ReLU* activations for the remaining layers. You observe that the weights for some layers cease updating after the initial training epoch, even though the network hasn't fully adapted yet. Further investigation shows that the gradients for these specific layers either diminish to zero or nearly zero early in the training process. Additionally, the loss value remains within an acceptable range. What potential solutions could address this issue?

No explanation required, just state i), ii), iii) or iv).

i) Switch the ReLU activations with leaky ReLUs everywhere.

ii) Increase the size of your training set.

iii) Increase the learning rate.

iv) Add Batch Normalization before every activation.

b Which of the following techniques can be used to reduce model overfitting?

No explanation required, just state i), ii), iii) or iv).

i) Data augmentation

ii) Dropout

iii) Batch Normalization

iv) Using Adam instead of SGD

c Which of the following statements about convolutional layers in a convolutional neural network (CNN) are true? none, several, or all might be true.

i) Convolutional layers apply a set of learnable filters to the input data to produce output feature maps.

ii) Convolutional layers are well-suited for tasks that involve 2D spatial relationships, such as image processing.

iii) Padding refers to the addition of extra rows and columns of zeros around the input image before applying the filter.

iv) Stride determines the step size of the filter and affects the size and resolution of the output feature map.

d You are benchmarking runtimes for layers commonly encountered in CNNs. Which of the following would you expect to be the fastest (in terms of floating point operations)? Choose one.

i) Conv layer (convolution operation + bias addition)

ii) Average pooling

iii) Max pooling

iv) Batch Normalization

e You are designing a deep learning system to diagnose chest cancer through X-ray images. What do you think might be the most appropriate evaluation metric and why: Accuracy, Precision, Recall, F1 score.

f You are training a single-layer, feedforward neural network with a softmax activation function in the final layer to classify among 250 classes, with a cross-entropy loss training objective. Recall, the cross-entropy loss function:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = -\mathbf{y} \cdot \log(\hat{\mathbf{y}})$$

where $\mathbf{y}$ is the one-hot encoded label, and $\hat{\mathbf{y}}$ is the predicted probability distribution over labels. You decide to independently sample your initial weights from a Gaussian of mean 0, standard deviation 0.0001. You can assume perfect class balance in dataset.

You accidentally **set a 0 learning rate**. What would you expect your average loss after the first training epoch to be? Provide a brief explanation (1-2 sentences) as to why this is so. You do not have to calculate the exact numerical value - feel free to leave your answer as a fraction.

g The CTO of a major retail corporation has assigned you the responsibility of developing a deep learning model. The objective is to forecast customer preferences for various products. Your primary task is to design a classifier that, when presented with a product image, can determine its category among four options: clothing, electronics, home goods, or outdoor equipment. You've been given an extensive collection of product images, with each image representing a product that exclusively fits into one of the mentioned categories.

i) How can you determine an estimate of human-level performance for the task?

You decide to use cross entropy loss to train your network. Recall that the cross-entropy loss for a single example is defined as follows:

$\mathcal{L}_{CE} = (\hat{y}, y) = -\sum_{i=1}^{n_y} y_i \log(\hat{y}_i)$ where $\hat{y} = (\hat{y}_1, \hat{y}_2, ..., \hat{y}_n)^T$ represents the predicted probability distribution over the classes and $y = (y_1, y_2, ..., y_n)^T$ represents the ground truth vector, which is zero everywhere except for the correct class (*e.g.*, $y = (1, 0, 0, 0)^T$ for clothing, and $y = (0, 0, 1, 0)^T$ for home goods).

ii) Suppose you are given an example image of an electronics product. If the model correctly predicts the resulting probability distribution as $\hat{y} = (0.05, 0.3, 0.45, 0.2)$, what is the value of the cross-entropy loss? You can give an answer in terms of logarithms.

iii) Your initial model achieves a 90% accuracy on the training dataset. What issues might the model be facing, and what potential solution(s) would you suggest?

iv) You have modified the model architecture and found that using a softmax classifier leads to good results. The last layer of the network calculates logits $z = (z_1, z_2, ..., z_{ny})$ which are passed through a softmax activation. The model achieves $100\%$ accuracy on the training data, but the training loss does not reach 0. Explain why the cross-entropy loss can never be 0 when using a softmax activation.

v) The model performs well on the training set but only has an accuracy of $85\%$ on the internal validation set, indicating overfitting. You intend to use L1 or L2 regularization to address this issue, but you are informed that some examples in the data may be mislabelled. In this situation, which form of regularization would you prefer and why?

h A Siamese network is a type of neural network architecture that consists of two or more identical subnetworks (called "twins") that share the same weights and architecture. Siamese networks are often used for tasks that involve comparing or matching inputs, such as verification, identification, and similarity learning.

The outputs of the twin networks are usually joined later on by more layers. Let's assume we have a two layer Siamese neural network, as defined below:

$$
\begin{aligned}
z_1 &= W_1 x^{(i)} + b_1 \\
a_1 &= ReLU(z_1) \\
z_2 &= W_1 x'^{(i)} + b_1 \\
a_2 &= ReLU(z_2) \\
a &= a_1 - a_2 \\
z_3 &= W_2 a + b_2 \\
\hat{y}^{(i)} &= \sigma(z_3) \\
\mathcal{L}^{(i)} &= y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \\
J &= -\frac{1}{m} \sum_{i=1}^{m} \mathcal{L}^{(i)}
\end{aligned}
$$

Note that $(x^{(i)}, x'^{(i)})$ represents a pair of single input examples, and are each of shape $D \times 1$ . Further $y^{(i)}$ is a single output label and is a scalar. There are $m$ examples in our dataset. We use $D_{a1}$ nodes in our first hidden layers; *i.e.*, $z_1$'s and $z_2$'s shape is $D_{a1} \times 1$. Note that the first two layers share the same weights.

i) What are the shapes of $W_1, b_1, W_2, b_2$? If we were vectorizing across multiple examples, what would the shapes of X and Y be instead?

ii) Derive $\frac{\partial J}{\partial z_3}$ formally and write $\delta_1^i = ....$ You can simplify the equation in terms of $\hat{y}^{(i)}$.

iii) Derive $\frac{\partial z_3}{\partial a}$ formally and write $\delta_2^i = ....$

iv) Derive $\frac{\partial a}{\partial z_2}$ formally and write $\delta_3^i = ....$

*The eigth parts carry, respectively, 10%, 15%, 5%, 5%, 5%, 10%, 25%, and 25% of the marks.*

2 Generative models, RNNs and Attention

a Consider a VAE with the graphical model visualised below:

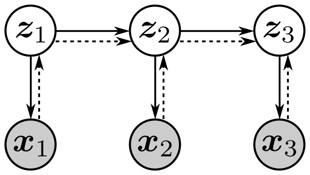

Fig. 1: Graphical model visualisation of $p$ (solid arrows) and $q$ (dashed arrows).

i) Write down the factorisation structure of $p(x_1, x_2, x_3, z_1, z_2, z_3)$ and $q(z_1, z_2, z_3|x_1, x_2, x_3)$ (in terms of product of conditional distributions).

ii) Consider a data distribution $p_{\text{data}}(x)$ and a VAE with the generative model $p_\theta(x|z)p(z)$ and encoder $q_\phi(z|x)$. Assume the neural networks are flexible enough so that the global optimum $q^*(z|x), p^*(z|x)$ of the VAE objective can be achieved. What is the result for $\int p^*(x|z)p(z)dz$?

iii) Which of the following statements are TRUE for VAEs? (Write the indices of your choice in your answer booklet.)

1. Gaussian distribution $q_\phi(z|x)$ is the best choice for the encoder
2. A VAE, when compared with a GAN, is less likely to suffer from mode collapse due to training objective design
3. We can parameterise the variance of Gaussian $q_\phi(z|x)$ as $\sigma = \text{LeakyReLU}(\text{NN}_\phi(x))$
4. A VAE should always use a continuous latent variable $z$
5. Test log-likelihood can be low even when reconstruction error is small

b Consider designing a generator $G$ for image super-resolution tasks, where the goal is to use this $G$ network to output a high resolution image given a low resolution image input. We assume we have two *separate* (i.e., not paired) datasets: $\mathcal{D}_L = \{x_n\}_{n=1}^N$ of low resolution images, and $\mathcal{D}_H = \{y_m\}_{m=1}^M$ of high resolution images.

i) Provide your design for a GAN adversarial loss that makes the outputs of $G$ look like realistic high resolution images.

ii) Given an input to $G$ (a low resolution image), we would also like the output of $G$ (a high resolution image), when downsampled, to be consistent with the input. Provide your design for a loss to ensure this consistency.

c Consider the following RNN to process the input sequence $\boldsymbol{x}_{1:T}$:

$$\boldsymbol{h}_t = \sigma(W_h \boldsymbol{h}_{t-1} + W_x \boldsymbol{x}_t + \boldsymbol{b}_h), \quad t = 1, ..., T$$

with the convention that $\boldsymbol{h}_0 = \mathbf{0}$, and $\sigma(\cdot)$ is the activation function. Assume the final output is $\boldsymbol{y} = \boldsymbol{h}_T$ and there is a loss function $\mathcal{L}(\boldsymbol{y})$ applied to it.

i) What is the fully expanded form of $\frac{d\mathcal{L}(\boldsymbol{y})}{dW_x}$?

ii) Assume $\sigma(\cdot)$ is identity mapping, $\boldsymbol{h}_t \in \mathbb{R}^{3\times 1}$ and $W_h$ is

$$W_h = \begin{pmatrix} 1.03 & 0.65 & -0.5 \\ -0.08 & -0.31 & 1.37 \\ 0.59 & 1.45 & 0.19 \end{pmatrix}.$$

The singular values of this matrix are approximately $(1.98, 1.38, 0.58)$. State whether gradient vanishing/explosion would happen when $T \to +\infty$ and explain in 1-2 sentences why.

iii) Which of the following solutions can help address RNN's gradient vanishing/explosion problem in practice? (Write the indices of your choice in your answer booklet.)

1. Use batch normalisation during training
2. Normalise the row vectors in $W_h$ to have $\ell_2$ norm 1
3. Use momentum gradient descent
4. Truncated back-propagation through time
5. Use softplus activations

d Attention mechanisms & Transformers:

i) Consider scaled dot-product hard attention: compute its output using the following inputs:

$$Q = \begin{pmatrix} 2 & -3 \\ -4 & 1 \\ 5 & -2 \end{pmatrix}, \quad K = \begin{pmatrix} 1 & 4 \\ 0 & -3 \end{pmatrix}, \quad V = \begin{pmatrix} 2 & 4 & 3 & -2 \\ -4 & 3 & 1 & -5 \end{pmatrix}.$$

ii) Which of the following statements are TRUE? (Write the indices of your choice in your answer booklet.)

1. Dot-product attention is permutation equivariant
2. To use position encodings the max. & min. indices need to be known
3. For text generation, a Transformer is an auto-regressive model
4. Given the total sum of head widths fixed, using more heads in multi-head attention makes the module more run-time efficient
5. The sinusoid encoding with fixed output dimensionality is a one-to-one mapping

*The four parts carry, respectively, 30%, 30%, 25%, and 15% of the marks.*