COMP4650/COMP6490 Document Analysis
2025 Semester 2

Computing Lab 3

Q1: Cross-Entropy Loss

(This is a theory question that does not require coding.)

Suppose we are training a linear classifier. For some training data point $(\mathbf{x},\ \mathbf{y})$ we have

$$\text{softmax}(W\mathbf{x}+\mathbf{b}) = \begin{bmatrix}0.1\\0.9\end{bmatrix},\ \mathbf{y} = \begin{bmatrix}1\\0\end{bmatrix}.$$

Compute the partial derivative of the loss of this data point with respect to the first bias $\frac{\partial L}{\partial b_1}$, where the loss L is the cross-entropy loss, i.e.

$$L = \sum_{i=1}^{2} y_i \log(\text{softmax}(W\mathbf{x}+\mathbf{b})_i) = \log(\text{softmax}(W\mathbf{x}+\mathbf{b})_1).$$

Q2: Back-propagation

(This is a theory question that does not require coding.)

Given a 2-layer neural network, $W^{[1]} = \begin{bmatrix}1 & 0\\-1 & 1\end{bmatrix}$, $\mathbf{b}^{[1]} = \begin{bmatrix}0\\0\end{bmatrix}$, $W^{[2]} = \begin{bmatrix}3 & 2\end{bmatrix}$, $b^{[2]} = 0$, training example $\mathbf{x} = \begin{bmatrix}2\\1\end{bmatrix}$, $y = 0$, and suppose the activation function is ReLU and the loss is the squared error $L(y,\hat{y}) = (y-\hat{y})^2$.

(a) Use back-propagation to compute the gradients $\nabla_{W^{[1]}} L$ and $\nabla_{W^{[2]}} L$.

(b) Perform a gradient descent update with learning rate $\alpha = 0.01$.

Q3: Pytorch Introduction

(This is a practice question that requires coding.)

Complete the PyTorch practical exercises in `lab3-pytorch_intro.ipynb`. To install PyTorch follow the instructions at `https://pytorch.org/get-started/locally/`. If you do not have a CUDA (Nvidia) GPU or ROCm (AMD) GPU on your machine then you should choose the `CPU` compute platform option.