STATS 403 (Spring'25) Homework 1

❗ For each problem, please clearly show your reasoning and write all the steps.
G As data scientists, you should feel free to google it whenever you see something unfamiliar.
☺ Group discussion for the homework is highly encouraged, but you have to write your answer by yourself. Also, you are always welcome to discuss the problems with me.

Part I (40')

Problem 1. backpropagation

Consider the simple neural network in Fig 1, describe how to compute $\partial y/\partial x$ using backpropagation.

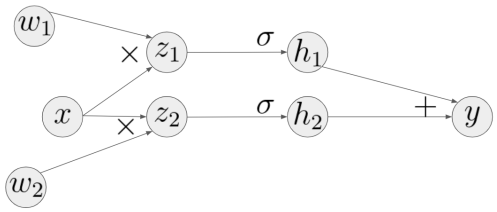

Figure 1: A simple neural network.

Problem 2. batch normalization

In "Lecture 2 Notebook 1", we defined a neural network with the following architecture: [input] $\to$ Linear(28$\times$28, 512) $\to$ ReLU $\to$ Linear(512, 512) $\to$ ReLU $*$ $\to$ Linear(512, 10) $\to$ ReLU $\to$ [output]

In this exercise, we try to produce Page 3 of Lecture 2's slides. First, use the above network to train 50 epochs. Plot the histogram of activation values of the second linear layer (the values of "$*$" above), versus epochs.

Then, add batch normalization as in "Lecture 2 Notebook 1". Plot the histogram of the activation after the second batch normalization, versus epochs.

Compare the two plots.

Problem 3. universal approximation

Let $f : [-1, 1]^n \to [-1, 1]$ be a Lipschitz function (that is, $|f(\boldsymbol{x}) - f(\boldsymbol{y})| \le L \|\boldsymbol{x} - \boldsymbol{y}\|$ for some constant L). Prove: given any $\epsilon > 0$, there is a neural network $N : [-1, 1]^n \to [-1, 1]$ with a sigmoid activation function, such that for every $\boldsymbol{x} \in [-1, 1]^n$, it holds that

$$|f(\boldsymbol{x}) - N(\boldsymbol{x})| \le \epsilon \, .$$

Problem 4. regularization

Suppose the objective function in training is $J(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{y})$, where $\boldsymbol{\theta}$ represents the parameters, $\boldsymbol{X}$ is the training data input, and $\boldsymbol{y}$ is target data output. One commonly used regularization is L^2 regularization, where the objective function is

$$\tilde{J}(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}) = J(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}) + \frac{\alpha}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} \ ,$$

whose gradient w.r.t. $\boldsymbol{w}$ is

$$\nabla_{\boldsymbol{w}}\tilde{J}(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}) = \nabla_{\boldsymbol{w}}J(\boldsymbol{w}; \boldsymbol{X}, \boldsymbol{y}) + \alpha\boldsymbol{w} \ .$$

1. Write the iteration step in the gradient descent method.

2. Assume the objective function is quadratic and $\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} J(\boldsymbol{w})$, then

   $$J(\boldsymbol{w}) = J(\boldsymbol{w}^*) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}^*)^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{w} - \boldsymbol{w}^*) \ .$$

   Show that the minimizer of $\tilde{J}$ is given by

   $$\tilde{\boldsymbol{w}} = (\boldsymbol{H} + \alpha\boldsymbol{I})^{-1}\boldsymbol{H}\boldsymbol{w}^* \ .$$

3. Show that $\tilde{\boldsymbol{w}}$ is a scaled version of $\boldsymbol{w}^*$ in the sense that along the direction of the i-th eigenvector of $\boldsymbol{H}$, $\boldsymbol{w}^*$ is scaled by a factor of

   $$\frac{\lambda_i}{\lambda_i + \alpha} \ .$$

Part II (40'+10')

Finish Problems 1, 2, 3, 4, 6 on the next page. If you also finish Problems 5 or 7, you will get 10 bonus points.

Part III (20'+5')

Finish the attached ipynb (either the PyTorch one or the TensorFlow one).

1 Convolution

1.1 (6 points)

Consider a rectangular signal which has value 1 for some finite duration $L > 1$, and value 0 everywhere else. First, compute by hand the convolution of two such rectangular signals. Then, write a Python function *convolve(x,h)* which computes the convolution of two signals of the same length (though not necessarily rectangular). Finally, graph the output of your function when passing two rectangular signals as input. Does it match the expectations of your original computation? Please include a copy of your code and output with your submission.

1.2 (2 points)

If $y(n) = x(n) * h(n)$, show that $\sum_{n=-\infty}^{\infty} y(n) = (\sum_{n=-\infty}^{\infty} x(n))(\sum_{n=-\infty}^{\infty} h(n))$.

1.3 (4 points)

The equation for the discrete convolution of two signals is:

$$(x * h)[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \tag{1}$$

In the case of finite length signals, we assume both sides of the signal to be zero-padded.

Compute the convolution of the following signals. As a hint, you should expect your answer to be a vector written similarly to $x(n)$ and $h(n)$, with a length one less than the sum of the length of both inputs.

1. $x(n) = [1, 2, 4]$, $h(n) = [1, 1, 1, 1, 1]$
2. $x(n) = [1, 2, -1]$, $h(n) = x(n)$

1.4 (4 points)

Compute and plot the convolutions $x(n) * h(n)$ and $h(n) * x(n)$ for the pairs of signals shown in Figure 1.

2 Backpropagation (4 points)

In binary classification, we seek to associate an input x_n with its appropriate target value: $t_n = 0$ for class C_0, and $t_n = 1$ for class C_1. The cross-entropy error function is of the form

$$E(w) = -\sum_{n=1}^{N} \{t_n \ln y_n + (1 - t_n) \ln(1 - y_n)\} \tag{2}$$

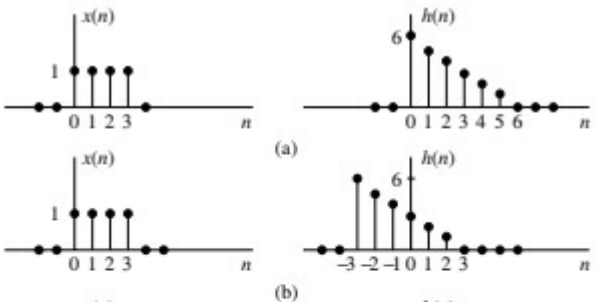

Figure 1: Signal Pairs for 2.4.4

Assuming a neural network with output units which use the logistic sigmoid activation function,

$$y_n = \sigma(a) = \frac{1}{1 + e^{-a}} \tag{3}$$

show the derivative of the above error function satisfies

$$\frac{\partial E}{\partial a_k} = y_k - t_k \tag{4}$$

Hint: show that $\frac{d\sigma}{da} = \sigma(1 - \sigma)$. *Use this relation to simplify the expressions for the derivatives of* $\ln(y)$ *and* $\ln(1 - y)$.

3 Jensen's Inequality (4 points)

Jensen's inequality states:

$$f(\sum_{i=1}^{M} \lambda_i x_i) \leq \sum_{i=1}^{M} \lambda_i f(x_i) \tag{5}$$

where $\lambda_i \geq 0$ and $\sum_i \lambda_i = 1$, for any set of points $\{x_i\}$.

Consider an M-state discrete random variable x, with entropy

$$H(x) = -\sum_{i=1}^{M} p(x_i) \ln p(x_i) \tag{6}$$

Show that the entropy of its distribution $p(x)$ satisfies:

$$H[x] \leq \ln M \tag{7}$$

4 Information Theory

4.1 Entropy (4 points)

1. For a biased coin with probability $P(head) = p$ and $P(tail) = 1 - p$, show that maximal entropy is achieved when the coin is fair.

   Hint: Write the expression of binary entropy as a function of p *and find its maximum.*

2. Plot $H(p)$ as a function of p, where p takes values from 0 to 1.

4.2 Mutual Information (8 points)

Assume a composer writes a melody by choosing for the first measure two notes from the 8 major scale notes spanning an octave, and for the second measure choosing the same two notes plus either the first or fifth scale degree. The purpose of the question is to consider how much information (bits) need to be sent if we already know something about how a sequence is generated, or how it may relate to another sequence (such as its own past).

1. If we treat the composer as an information source, and we know that they always use the above method for writing their melodies, how many bits are required to represent the first bar?

   Hint: Determine how many bits are required to represent initially the alphabet (the total number of possible notes in octave), which gives you # of bits per note without any prior information or compressions.

2. How many bits are required to represent the second bar?

   Hint: Consider the number of bits you need when you use the prior information from the first bar to encode the second bar.

3. If we represent the first bar as random variable X and the second bar as random variable Y, write the expressions for $H(X)$, $H(Y)$, and $H(Y|X)$.

4. What is the mutual information $I(X, Y)$ between the first and second bar? *Hint: Think about how many bits of information are already contained in X in order to compose Y.*

5 ELBO

5.1 Latent Variables (10 points)

1. The basic idea in the Variational Method and Expectation-Maximization is to maximize the ELBO instead of likelihood $P(X|\theta)$. With this "trick", we can find the optimal model shown in Class. For this problem, we provide a slightly different formulation, where the expectation E is done over $q(z)$ instead of $q(z|x)$. Both versions of ELBO are used in practice, depending on whether the objective is to construct an encoder Q that is more sensitive to input X or not.

   Prove that the log likelihood $\ln p(X|\theta)$ can be decomposed as ELBO + KL$(q||p)$, where

   $$\text{ELBO} = \mathcal{L}(q, \theta) = \sum_z q(Z) \ln \frac{p(X, Z|\theta)}{q(Z)} \tag{8}$$

   and

   $$\text{KL}(q||p) = -\sum_z q(Z) \ln \frac{p(Z|X, \theta)}{q(Z)} \tag{9}$$

   with q representing the distribution of the latent variables Z and KL$(q||p)$ representing the Kullback-Leibler divergence between the distributions q and p.

   Hint: Use definitions of conditional probability (Bayes' rule) so that the sum of the ELBO and KL expressions cancel the dependency of the probability on the variable Z.

2. ELBO and EM are closely related. One is approximating the likelihood using VAE through differentiable programming (i.e. gradient descent), and the other is an iterative solution. While VAE uses gradient descent to find the approximate distribution q, the EM method uses an old estimate of $p(x|z, \theta)$ as the approximation.

   In the EM algorithm, the expectation of the complete-data log likelihood evaluated for some general parameter value θ is given as

   $$Q(\theta, \theta^{old}) = \sum_z p(Z|X, \theta^{old}) \ln p(X, Z|\theta) \tag{10}$$

   where $p(Z|X, \theta^{old})$ is the posterior distribution of the latent variables estimated in the E step using the current parameter values θ^{old}. Show that the ELBO is the same as $Q(\theta, \theta^{old})$, up to an entropy term in q, for $q = p(z|x, \theta)$.

   Hint: entropy is defined as $\sum q \ln q$.

3. Argue that the best approximate distribution q for a given parameter θ is the posterior distribution $p(Z|X, \theta)$, and use this argument to explain the purpose of the E step in the EM algorithm.

4. Show that maximizing the ELBO is equivalent to maximizing $Q(\theta, \theta^{old})$. *Hint: Verify which part of the equation depends on the model parameter θ.*

6 Gaussian Approximation (4 points)

Suppose that $p(x)$ is some fixed distribution and that we wish to approximate it using a Gaussian distribution $q(x) = \mathcal{N}(x|\mu, \Sigma)$. By writing down the form of the KL divergence KL$(p||q)$ for a Gaussian $q(x)$ and then differentiating, show that minimization of KL$(p||q)$ with respect to μ and Σ leads to the result that μ is given by the expectation of x under $p(x)$ and that Σ is given by the covariance.

7 Probabilistic PCA (10 points)

In probabilistic PCA, the observations are assumed to be generated from latent variable z with added noise ϵ according to

$$x = Wz + \mu + \epsilon \tag{11}$$

1. Using expressions for the mean and covariance of x, prove that

   $$E[x] = \mu \tag{12}$$

   and

$$cov[x] = WW^T + \sigma^2 \tag{13}$$

In your explanation, use the assumption that

$$z \sim N(\vec{0}, I) \tag{14}$$

Explain where you use this assumption.

2. It can be shown that
$$p(z|x) = \mathcal{N}(z|M^{-1}W^T(x-\mu), M^{-1}\sigma^{-2}) \tag{15}$$
where M is defined as
$$M = W^TW + \sigma^2 I \tag{16}$$
Show that as $\sigma \to 0$, the posterior mean for z given x becomes
$$E(z|x) = (W^TW)^{-1}W^T(x-\mu) \tag{17}$$

3. In non-probabilistic (regular) PCA, the goal is to approximate a vector x that has D dimensions by combination from a smaller set of basis vectors $w_1, w_2, ...w_M$ with $M < D$. Arranging the basis vectors as columns of a matrix W, consider $\hat{x} = Wz$ to be a low dimensional approximation to x. Explain why z is M-dimensional, and show that for a fixed W, the optimal z is found by the pseudo-inverse of W given by
$$(W^TW)^{-1}W^T \tag{18}$$
Compare this to the maximum likelihood result for probabilistic PCA, and describe your observation.

Hint: to find a pseudo-inverse, write $x = Wz + error$. Then, write an expression for MSE and minimize with respect to z, showing that optimal z is given by $(W^TW)^{-1}W^Tx$.