Exam Advanced Machine Learning
27 October 2022, 18.45–21.00

This exam consists of 5 problems, each consisting of several questions. All answers should be motivated, including calculations, formulas used, etc. The use of a calculator is not allowed.

Question 1: Short questions
Please provide an argument for your answer on the following questions.

(a) Suppose that you use linear regression to model a particular dataset. To test your linear regression model, you choose at random some records to be the training set, and choose at random some of the remaining records to be the test set. Now, let us increase the size of the training set gradually. Explain what will happen to the mean training error and the mean testing error.

The training error tends to increase. As more examples have to be fitted, it becomes harder to 'hit', or even come close, to all of them. The test error tends to decrease. As we take into account more examples when training, we have more information, and can come up with a model that better resembles the true behavior. More training examples lead to better generalization.

(b) Jason and Bob are discussing which structural assumptions in polynomial regression most affect the trade-off between underfitting and overfitting. Jason claims that the polynomial degree in the regression is more important for the trade-off, whereas Bob claims that the assumed variance in the Normal distribution of the error is more important. Who is right, and why?

Jason is right.

(c) Suppose that you are given a train set X_{train}, Y_{train} and a test set X_{test}, Y_{test}. You want to normalize your data before training your model. Argue if the following statement is true or false. The test data should be normalized with its own mean and variance before being fed to the network at test time because the test distribution might be different from the train distribution.

The statement is FALSE.

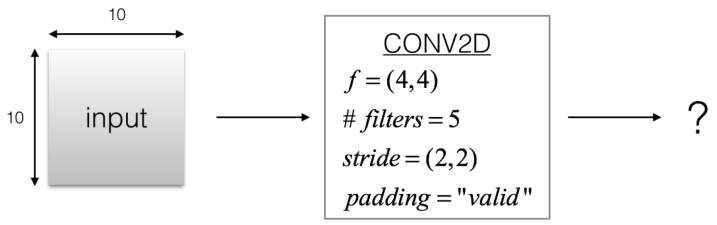
(d) You are doing full batch gradient descent using the entire training set (not stochastic gradient descent). Is it necessary to shuffle the training data? Explain your answer.

It is not necessary. Each iteration of full batch gradient descent runs through the entire dataset and, therefore, the order of the dataset does not matter.

(e) Weight sharing allows convolutional neural networks to deal with image data without using too many parameters. Does weight sharing increase the bias or the variance of a model?

Weight sharing increases the bias.

(f) Consider the figure below.



The input is of shape $(n_H, n_W, n_C) = (10, 10, 1)$. There are five 4×4 convolutional filters with 'valid' padding (i.e., zero padding) and a stride of $(2, 2)$. What is the output shape after performing the convolution step? Write your answer in the following format: (n_H, n_W, n_c).

$(n_H, n_W, n_c) = (4, 4, 5)$.

(g) You are consulting for a healthcare company. A patient may have any number of illnesses from a list of 70,000 known medical illnesses. The output of your recurrent neural network will therefore be a vector with 70,000 elements. Each element in this output vector represents the probability that the patient has the illness that maps to that particular element. Illnesses are not mutually exclusive, i.e., having one illness does not preclude you from having any other illnesses. Given this insight, what activation function would you use for your output unit? Explain your answer.

Sigmoid. In the softmax case, the presence of one disease would lower the probability of all other diseases. This contradicts our assumption that the diseases are not mutually exclusive.

Question 2: Neural networks

The following neural network in Figure 1 has 3 units. The neural network operates as a regular neural network. Each unit takes a linear combination of the units of the previous layer, adds a bias term, and then applies an activation function g to obtain the activated units (i.e., a_1, a_2, or a_3). Additionally, this network uses the standard error function $E = \frac{1}{2}(y - t)^2$, with t the target value and $y = a_3$, the output of the neural network.

(a) We discussed several activation functions during the lecture. We did not discuss the Exponential Linear Unit (ELU) g_1 and the SoftPlus g_2 activation functions. These activation functions are given by

$$g_1(z) = \begin{cases} \alpha(e^z - 1), & \text{for } z < 0, \\ z, & \text{for } z \geq 0, \end{cases} \qquad \text{and} \qquad g_2(z) = \log_e(1 + e^z),$$

for some $\alpha \in \mathbb{R}$. Calculate the derivative of $g_i(z)$ with respect to z for both $i = 1, 2$.

$$g_1'(z) = \begin{cases} \alpha(e^z - 1) + \alpha = g_1(z) + \alpha, & \text{for } z < 0, \\ 1, & \text{for } z \geq 0, \end{cases} \qquad \text{and} \qquad g_2(z) = \frac{1}{1 + e^{-z}}.$$

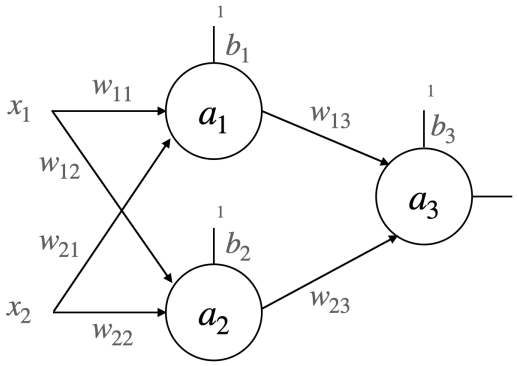


Figure 1: Neural network architecture.

(b) Discuss the advantages and disadvantages of the ELU and SoftPlus activation functions over the sigmoid, tanh, and ReLu activation functions.

Continuous activation function with non-zero derivatives.

Assume that node a_1 and node a_2 are activated by g_1 and node a_3 is activated by g_2.

(c) Calculate $\partial E/\partial w_{13}$.

$$\frac{\partial E}{\partial w_{13}} = \frac{\partial E}{\partial a_3}\frac{\partial a_3}{\partial z_3}\frac{\partial z_3}{\partial w_{13}} = (a_3 - t)\cdot g_2'(z_3)\cdot a_1.$$

(d) Calculate $\partial E/\partial w_{11}$.

$$\frac{\partial E}{\partial w_{11}} = \frac{\partial E}{\partial a_3}\frac{\partial a_3}{\partial z_3}\left[\frac{\partial z_3}{\partial a_1}\frac{\partial a_1}{\partial z_1}\frac{\partial z_1}{\partial w_{11}} + \frac{\partial z_3}{\partial a_2}\frac{\partial a_2}{\partial z_2}\frac{\partial z_2}{\partial w_{11}}\right] = \frac{\partial E}{\partial a_3}\frac{\partial a_3}{\partial z_3}\frac{\partial z_3}{\partial a_1}\frac{\partial a_1}{\partial z_1}\frac{\partial z_1}{\partial w_{11}}$$

$$\frac{\partial E}{\partial w_{11}} = (a_3 - t)\cdot g_2'(z_3)\cdot w_{13}\cdot g_1'(z_1)\cdot x_1.$$

(e) We have discussed the linear activation function during the lecture. This was given by $z = w_0 + \sum_i w_i x_i$. Now, consider the hard threshold

$$z = \begin{cases} 1, & \text{if } w_0 + \sum_i w_i x_i \geq 0, \\ 0, & \text{otherwise.} \end{cases}$$

Which of the following functions can be exactly represented by a neural network with one hidden layer which uses linear and/or hard threshold activation functions? For each case, justify your answer.

(i) polynomials of degree one

Yes

(ii) hinge loss, i.e., $h(x) = \max\{1 - x, 0\}$

No

(iii) polynomials of degree two
No
(iv) piecewise constant functions
Yes

Question 3: Graphical models
The following figure shows a graphical model over nine binary-valued variables $A, \ldots, I$. We do not know the parameters of the probability distribution associated with the graph.

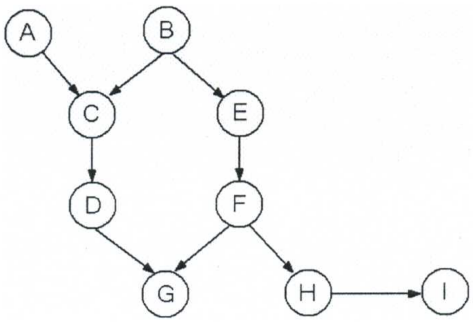


Figure 2: Graphical model.

(a) Write the expression for the joint probability $\mathbb{P}(A, B, C, D, E, F, G, H, I)$ of the network in its *reduced* factored form.
$\mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C|A,B)\mathbb{P}(D|C)\mathbb{P}(E|B)\mathbb{P}(F|E)\mathbb{P}(G|D,F)\mathbb{P}(H|F)\mathbb{P}(I|H)$

(b) Which of the following conditional independence assertions are true?

(i) $A \perp\!\!\!\perp B \,|\, G$
False
(ii) $A \perp\!\!\!\perp I$
True
(iii) $B \perp\!\!\!\perp H \,|\, E, G$
False
(iv) $\mathbb{P}(C \,|\, B, F) = \mathbb{P}(C \,|\, F)$
False

Question 4: Hidden Markov Models (HMMs)
Consider an HMM with latent states $Z_t \in \{1, 2, 3\}$, and observations $X_t \in \{A, B, C\}$. The initial distribution is given by $\pi = (\pi_1, \pi_2, \pi_3) = (1, 0, 0)$. The transition probabilities of the latent states and the emission probabilities are given by

$$A = \begin{pmatrix} 1/2 & 1/4 & 1/4 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \text{and} \qquad \varphi = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1/2 & 1/2 \end{pmatrix}.$$

Thus, e.g., $\mathbb{P}(Z_t = 2 \,|\, Z_{t-1} = 1) = 1/4$ and $\mathbb{P}(Z_t = 2 \,|\, Z_{t-1} = 2) = 1/2$. Similarly, states A and B are observed with probability $1/2$ in latent state 1, A and C in latent state 2, and states B and C in latent state 3.

(a) Calculate the probability $\mathbb{P}(Z_5 = 3)$.

$1 - \mathbb{P}(Z_5 = 1) - \mathbb{P}(Z_5 = 2) = 1 - 1/16 - 4 \times 1/32 = 13/16$

(b) Calculate the probability $\mathbb{P}(Z_5 = 3 \,|\, (X_1, \ldots, X_7) = (A, A, B, C, A, B, C))$.

0

(c) Write down the sequence $Z_1, \ldots, Z_7$ that maximizes the probability of observing the sequence $(X_1, \ldots, X_7) = (A, A, B, C, A, B, C)$.

(1, 1, 1, 2, 2, 3, 3) with posterior probability 1

(d) Suppose that you are training an HMM with a small number of latent states from a large number of observations. Explain if, in general, you can increase the training data likelihood by permitting more latent states.

To model any finite length sequence, we can increase the number of hidden states in an HMM to be the number of observations in the sequence and therefore (with appropriate parameter choices) generate the observed sequence with probability 1. Given a fixed number of finite sequences (say n), we would still be able to assign probability 1/n for generating each sequence. This is not useful, of course, but highlights the fact that the complexity of HMMs is not limited.

Question 5: Machine learning for blackjack

Armed with the power of Q-learning, you go to Holland Casino. You play a simplified version of blackjack where the deck is infinite and the dealer always has a fixed count of 15. The deck contains cards 2 through 10, J, Q, K, and A, each of which is equally likely to appear when a card is drawn. Each number card is worth the number of points shown on it, the cards J, Q, and K are worth 10 points, and A is worth 11.

At each turn, you may either *hit* or *stay*. If you choose to *hit*, you receive no immediate reward and are dealt an additional card. If you *stay*, you receive a reward of 0 if your current point total is exactly 15, $+10$ if it is higher than 15 but not higher than 21, and -10 otherwise (i.e., lower than 15 or larger than 21). After taking the *stay* action, the game enters a terminal state *end* and ends. A total of 22 or higher is refered to as a *bust*; from a bust, you can only choose the action *stay*.

As your state space, you take the set $\{0, 2, \ldots, 21, \text{bust}, \text{end}\}$ indicating point totals, "bust" if your point total exceeds 21, and "end" for the end of the game.

(a) Suppose you have performed k iterations of value iteration. Compute $V_{k+1}(12)$ given the partial table below for $V_k(s)$. Give your answer in terms of the discount γ as a variable. Note: do not worry about whether the listed V_k values could actually result from this MDP!

$V_{k+1}(12) = 1/13 \times (8 \times 10\gamma + 5 \times (-10)\gamma) = 30/13\gamma$. There are 8 cards (2 through 9) that will take us to a state with return 10, and 5 cards (10 through Ace) that will take us to the *bust* state.

(b) You suspect that the cards do not actually appear with equal probability and decide to use Q-learning instead of value iteration. Given the partial table of initial Q-values below, update the partial table of Q-values after the following episode occurred. Assume a learning rate of 0.5 and a discount factor of $\gamma = 1$. The initial portion of the episode has been omitted.

How are the values updated? Here's a sample one: $Q(19, hit) = 0.5 \times -2 + 0.5 \times (0 + 1 \times \max(-6, 8)) = 3$.

s	$V_k(s)$
13	2
14	10
15	10
16	10
17	10
18	10
19	10
20	10
21	10
bust	-10
end	0

s	a	$Q(s,a)$
19	hit	-2
19	stay	5
20	hit	-4
20	stay	7
21	hit	-6
21	stay	8
bust	stay	-8

Episode

s	a	r	s	a	r	s	a	r
19	hit	0	21	hit	0	bust	stay	-10

s	a	$Q(s,a)$
19	hit	3
19	stay	
20	hit	
20	stay	
21	hit	-7
21	stay	
bust	stay	-9

(c) Unhappy with your experience with basic Q-learning, you decide to featurize your Q-values, representing them in the form $\sum_i w_i f_i(s,a)$ for some feature functions $f_i(s,a)$.

Consider the two feature functions

$$f_1(s,a) = \begin{cases} 0, & \text{if } a = \text{stay}, \\ +1, & \text{if } a = \text{hit and } s \geq 15, \\ -1, & \text{if } a = \text{hit and } s < 15. \end{cases} \quad \text{and} \quad f_2(s,a) = \begin{cases} 0, & \text{if } a = \text{stay}, \\ +1, & \text{if } a = \text{hit and } s \geq 18, \\ -1, & \text{if } a = \text{hit and } s < 18. \end{cases}$$

For which of the following partial policy tables (i)–(v) is it possible to represent Q-values in the form $w_1 f_1(s,a) + w_2 f_2(s,a)$ that imply that policy unambiguously (i.e., without having to break ties)?

Table (iii).

(i)

s	$\pi(s)$
14	hit
15	hit
16	hit
17	hit
18	hit
19	hit

(ii)

s	$\pi(s)$
14	stay
15	hit
16	hit
17	hit
18	stay
19	stay

(iii)

s	$\pi(s)$
14	hit
15	hit
16	hit
17	hit
18	stay
19	stay

(iv)

s	$\pi(s)$
14	hit
15	hit
16	hit
17	hit
18	hit
19	stay

(v)

s	$\pi(s)$
14	hit
15	hit
16	hit
17	stay
18	hit
19	stay

partial grade	1	2	3	4	5
(a)	1	2	1	2	1
(b)	1	2	4	2	2
(c)	1	2		2	3
(d)	1	2		1	
(e)	1	3			
(f)	1				
(g)	1				

Final grade is: (sum of partial grades) / 4.0 + 1.0