CSCI567 Machine Learning (Fall 2025)
Instructor: Haipeng Luo

Exam Two (Sample)

Duration: 120 minutes

NAME _____

Stu ID _____

INSTRUCTIONS:

1. Please fill in your name and ID in the space above.
2. When the exam ends, you have 20 minutes to upload your solutions to Gradescope using your phone similarly to how you upload your homework (**do not use your phone for any reason before that!**). Make sure to match your solutions to the corresponding questions. **Do not continue working on the exam during these last 20 minutes.**
3. After finishing uploading, please also turn in your exam before leaving the room.
4. If you encounter any difficulty in uploading (or you are taking the exam in a test center that prohibits cell phones), simply turn in your exam and we will upload it for you.
5. This exam has a total of 15 pages (including this cover page). Check and make sure that you are not missing any pages. Additionally, there are 5 pages provided for scratch work at the end.
6. Any kind of cheating will be reported and lead to a 0 score for the entire exam.

1 Multiple-Choice Questions (30 points)

IMPORTANT: *Select ALL answers that you think are correct. You get 0.5 point for selecting each correct answer and similarly 0.5 point for not selecting each incorrect answer.* ***You MUST write down your answers on the line "_____" after the question****; only circling or crossing the letters will not count since they might be hard to recognize. If you truly think no answers are correct, write "****none****" on the line; otherwise, we will treat it as you giving up on this question and give you 0 point.*

(1) Which of the following is usually considered as an unsupervised learning problem? _____

(A) Predicting the price of a product based on historical data.
(B) Discovering communities in a social network.
(C) Visualizing genome data in a 2D space.
(D) Pre-training a language model with a huge corpus.

(2) Which of the following about the convergence of K-means is correct? _____

(A) K-means always converges after a finite number of iterations.
(B) K-means might run forever and never converge.
(C) K-means++ always converges after a finite number of iterations.
(D) K-means++ might run forever and never converge.

(3) Which of the following about Gaussian Mixture Model (GMM) and Expectation-Maximization (EM) algorithm is correct? _____

(A) GMM assumes that the data are generated via a mixture of Gaussian and is therefore not directly suitable for data with discrete features.
(B) EM might only find a local maximum of the total log likelihood of a GMM.
(C) Learning a GMM via EM gives not only the cluster (soft) assignments and centers, but also a way to generate new synthetic data.
(D) EM for GMM can be seen as a soft-version of the K-means algorithm.

(4) Which of the following about Principal Component Analysis (PCA) is correct? _____

(A) PCA is one way to reduce the dimensionality of data.
(B) The first principal component is the direction where the data have the least variance.
(C) PCA requires finding all eigenvectors of the covariance matrix.
(D) PCA requires finding all eigenvalues of the covariance matrix.

(5) Let $\boldsymbol{X}$ be an $N \times D$ data matrix where each row corresponds to a portrait, and $\boldsymbol{V} \in \mathbb{R}^{D \times p}$ be the top p eigenfaces. If one wants to compress the dataset while at the same time being able to approximately reconstruct the original portraits, which of the following should be stored? _____

(A) $\boldsymbol{V}$ (B) $\boldsymbol{XV}$ (C) $\boldsymbol{XVV}^\top$ (D) $\boldsymbol{X}^\top\boldsymbol{X}$

(6) Suppose that the covariance matrix of a dataset $\boldsymbol{X} \in \mathbb{R}^{N\times 4}$ is $\boldsymbol{X}^\top \boldsymbol{X} = \begin{pmatrix} 25 & 3 & 13 & -1 \\ 3 & 25 & -1 & 13 \\ 13 & -1 & 25 & 3 \\ -1 & 13 & 3 & 25 \end{pmatrix}$ and its top three eigenvalues are $40, 36$ and 16 respectively. How many principal components we need to pick when performing PCA if we want 80% of the variance explained? _____

(A) 1 (B) 2 (C) 3 (D) 4

(7) In an HMM, which of the following quantities is equal to $P(X_{1:T} = x_{1:T})$? (α and β are forward and backward messages respectively, and a and b are model parameters as discussed in the lecture.) _____

(A) $\sum_s \alpha_s(1)\beta_s(1)$ (B) $\sum_s \alpha_s(T)\beta_s(T)$
(C) $\sum_{s,s'} \alpha_s(1)a_{s,s'}b_{s',x_2}\beta_{s'}(2)$ (D) $\sum_{s,s'} \alpha_s(T-1)a_{s,s'}b_{s',x_T}$

(8) Suppose that $z_1^*, \ldots, z_T^*$ is the output of the Viterbi algorithm given a sequence of observations $x_1, \ldots, x_T$. Which of the following is correct? _____

(A) $z_1^*, \ldots, z_T^*$ is a path of states sampled from the posterior distribution given $x_1, \ldots, x_T$.
(B) $z_1^*, \ldots, z_T^*$ is the expected path of states given $x_1, \ldots, x_T$.
(C) $z_1^*, \ldots, z_T^*$ is the most likely path of states given $x_1, \ldots, x_T$.
(D) Each z_t^* is the most likely state at time step t given $x_1, \ldots, x_T$.

(9) Which of the following about Recurrent Neural Networks (RNN) is correct? _____

(A) Doubling the number of input tokens of an RNN also roughly doubles the time it takes to process this input sequence.
(B) Doubling the number of input tokens of an RNN also roughly doubles its number of parameters.
(C) The number of input tokens and the number of output tokens in an RNN are always the same.
(D) When generating texts via softmax, the smaller the temperature parameter is, the more random the final outputs are.

(10) Which of the following about Transformers is correct? _____

(A) The encoders in a transformer summarize the input into a useful representation.
(B) Positional encoding addresses the issue that an attention head does not have positional information.
(C) Residual pathway mitigates the issue of vanishing gradients.
(D) The input of the self-attention head in an decoder is the final output of the encoder stack.

(11) Which of the following about multi-armed bandits is correct? _____

(A) Explore-then-exploit is the best way to handle exploration-exploitation trade-off.
(B) ϵ-greedy never stops exploration.
(C) Optimism in face of uncertainty is usually a good way to ensure adaptive exploration.
(D) If an arm always gives reward 0, UCB will eventually give up this arm and never select it again.

(12) Which of the following about dueling bandits and self-play is correct? _____

(A) Based on the self-play idea, one can directly apply two copies of UCB to solve dueling bandits.
(B) When using self-play to solve dueling bandits, a_t and b_t are always sampled from the exact same distribution.
(C) When using two copies of Exp3 to play against each other in dueling bandits, the faction of times where the Condorcet winner is not selected is negligible in the long run.
(D) Self-play is a general idea to approximately find a Nash equilibrium for a two-player zero-sum game.

(13) Which of the following about reinforcement learning is correct? _____

(A) Just like multi-armed bandits, a reinforcement learning algorithm also needs to properly trade off exploration and exploitation.
(B) Value iteration might not always converge.
(C) Experience replay is an effective way to reduce correlation and increase data efficiency.
(D) Policy gradient methods directly optimize the total reward over the policy space.

(14) Which of the following is the correct Value Iteration update? _____

(A) $V(s) \leftarrow \min_{a\in\mathcal{A}} \left(r(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s'|s,a)V(s')\right)$
(B) $V(s) \leftarrow \max_{a\in\mathcal{A}} \left(r(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s'|s,a)V(s')\right)$
(C) $V(s) \leftarrow \max_{a\in\mathcal{A}} \left(r(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s|s',a)V(s')\right)$
(D) $V(s) \leftarrow \max_{a\in\mathcal{A}} r(s,a) + \gamma \max_{a\in\mathcal{A}} \left(\sum_{s'\in\mathcal{S}} P(s'|s,a)V(s')\right)$

(15) Policy gradient methods directly find the optimal policy by applying (stochastic) gradient descent to the objective function $R(\pi_\rho)$. Which of the following is equal to $\nabla_\rho R(\pi_\rho)$? _____

(A) $\mathbb{E}_\tau \left[\sum_{h=1}^{H} \nabla_\rho \log \pi_\rho(a_h|s_h) R(\tau)\right]$
(B) $\mathbb{E}_\tau \left[\sum_{h=1}^{H} \nabla_\rho \log \pi_\rho(a_h|s_h) \left(R(\tau) - 10\right)\right]$
(C) $\mathbb{E}_\tau \left[\sum_{h=1}^{H} \nabla_\rho \log \pi_\rho(a_h|s_h) \left(R(\tau) - b(s_{1:h}, a_{1:h-1})\right)\right]$ for any function b
(D) $\mathbb{E}_\tau \left[\sum_{h=1}^{H} \nabla_\rho \log \pi_\rho(a_h|s_h) \left(\sum_{h'=h}^{H} r(s_{h'}, a_{h'}) - V_\theta(s_h)\right)\right]$ for some target network θ

2 HMM (8 points)

Recall that a hidden Markov model with a state space S and an observation space O is parameterized by:

- initial state distribution $P(Z_1 = s) = \pi_s$,
- transition distribution $P(Z_{t+1} = s' \mid Z_t = s) = a_{s,s'}$,
- emission distribution $P(X_t = o \mid Z_t = s) = b_{s,o}$.

Imagine a speech sample of length-T generated by this HMM. Unfortunately, due to low quality of the device that collects the speech, you only have partial observations of this sequence. Follow the steps below to infer the hidden states.

2.1 Let $\Omega \subset [T]$ be a subset of time steps, and suppose that we are given a sequence of partial observations x_t for $t \in \Omega$. Further let $\Omega_{\leq t}$ be the shorthand for $\{\tau \in \Omega : \tau \leq t\}$ and similarly $\Omega_{\geq t}$ be the shorthand for $\{\tau \in \Omega : \tau \geq t\}$. Redefine the forward and backward messages as

$$\alpha_s(t) = P(Z_t = s, X_k = x_k, \forall k \in \Omega_{\leq t}) \quad \text{and} \quad \beta_s(t) = P(X_k = x_k, \forall k \in \Omega_{\geq t+1} \mid Z_t = s).$$

When the parameters of the HMM are known, computing these messages is similar to the original forward and backward procedure. Take the forward message as an example. Recall the original forward procedure:

Algorithm 1: Original forward procedure

Input: observations $x_1, \ldots, x_T$
Initialization: for each state $s \in S$, compute $\alpha_s(1) = \pi_s b_{s,x_1}$
for $t = 2, \ldots, T$ **do**
  for each state $s \in S$, compute

$$\alpha_s(t) = b_{s,x_t} \sum_{s' \in S} a_{s',s} \alpha_{s'}(t-1)$$

Fill in the missing details in the modified forward procedure below. No reasoning is needed. (4 points)

Algorithm 2: Modified forward procedure

Input: observations x_t for $t \in \Omega$
Initialization: for each state $s \in S$, compute
for $t = 2, \ldots, T$ **do**
  for each state $s \in S$, compute

2.2 Now suppose that you want to infer the hidden state Z_t for a particular time step t, given all the observations x_k for $k \in \Omega$. In other words, we are interested in finding for a state $s \in S$, the probability

$$P(Z_t = s \mid X_k = x_k, \forall k \in \Omega).$$

Express this probability using the modified $\alpha_s(t)$ and $\beta_s(t)$ from the last question (as well as the model parameters). Show your derivation, which can use the propositional sign, but express your final answer WITHOUT using it. (4 points)

3 RNNs and Transformers (16 points)

3.1 Consider the following mini RNN which deals with only 4 possible different characters. Suppose that before we feed the one-hot representations of these characters to the RNN, we want to first covert them into 10-dimensional word embeddings that are directly learned from training this RNN. How many parameters are there in this RNN then? Show your calculation and feel free to ignore all bias terms. (5 points)

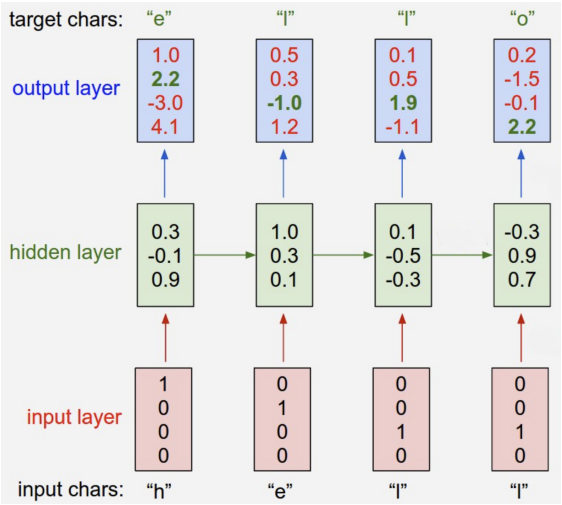


Figure 1: A mini RNN

3.2 Consider a **decoder** in a transformer.

(1) For a 3-token input from the training data, the table below shows some partial values of the 3×3 matrix obtained after applying softmax to the attention score matrix plus a causal mask in a self-attention head, that is, softmax $\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top + \boldsymbol{M}}{\sqrt{d_k}}\right)$. Fill in the missing values in this table (no reasoning needed). (3 points)

0.1		
	0.6	0.1

(2) Continuing from the last question, suppose that the outputs of this self-attention head are 3 5-dimensional vectors, which are then fed into an encoder-decoder attention head where the query vectors and value vectors are of dimension 10 and 20 respectively. It is also known that the final outputs of the encoder stack are 4 15-dimensional vectors. Answer the following questions (no reasoning needed for the first two questions, but show your calculation for the last one).

- What is the dimension of the $\boldsymbol{Q}$, $\boldsymbol{K}$, $\boldsymbol{V}$ matrices in this encoder-decoder head? (3 points)
- What is the dimension of the attention score matrix in this encoder-decoder head? (1 points)
- How many parameters are there in this encoder-decoder head (ignoring bias terms)? (4 points)

4 Multi-Armed Bandits (8 points)

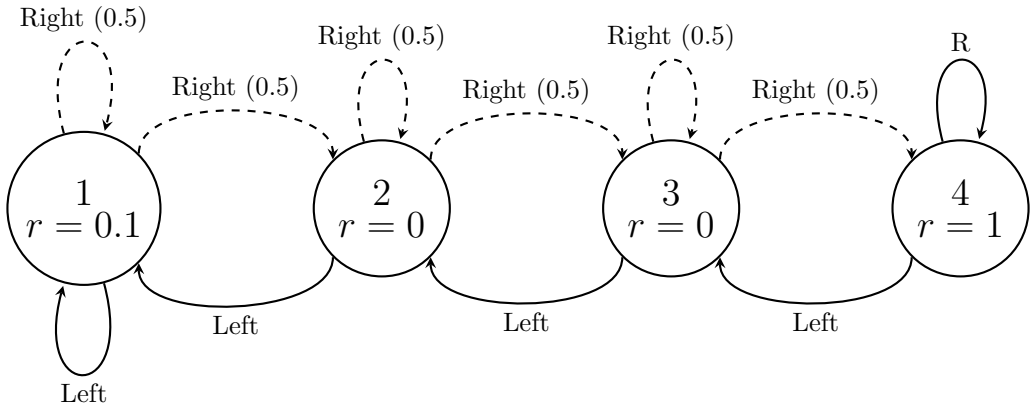
After using both the UCB algorithm and the Exp3 algorithm to pick the restaurant for lunch over the past two months, Alice finally decides that USC Cafe is her favorite and will have lunch there most of the time in the future. However, the restaurant recently has a new combo option in their menu that allows customers to pick two proteins (that could potentially be the same) from the following four choices: beef, chicken, pork, and tofu. Alice thinks that now it is a good time to try out a dueling bandit algorithm to see what she likes most. In particular, she will use two copies of the Exp3 algorithm (each with learning rate $\eta = 0.1$) to decide what to order for this combo.

4.1 Describe in one sentence how Alice should order on the first day. (2 points)

4.2 Suppose that Alice ends up picking chicken and tofu on the first day and finds herself in favor of tofu. Describe how Alice should order on the second day and explain why. (6 points)

5 Reinforcement Learning (10 points)

Consider the following "river-swim" MDP from HW4. For this question, you only need the facts that this MDP has four states (indexed by 1,2,3,4), two actions (Left and Right), and a discount factor $\gamma = 0.9$.



Now, consider running the Q-learning algorithm with ϵ exploration probability on this MDP. Suppose that at some point, the Q table has the following values:

$Q(s, a)$	$s = 1$	$s = 2$	$s = 3$	$s = 4$
$a =$ Left	0.1	0	0	0.9
$a =$ Right	0.3	0.1	0.2	1

5.1 If the agent is currently at state $s = 2$, how should they pick the next action? (4 points)

5.2 Suppose that the agent ended up picking $a = \text{Right}$, received reward 0, and transitioned to state $s' = 3$. How should we update the Q table when using a learning rate $\alpha = 0.2$? Fill in the table below with the updated Q values and also show your calculation below the table. (6 points)

$Q(s, a)$	$s = 1$	$s = 2$	$s = 3$	$s = 4$
$a = \text{Left}$				
$a = \text{Right}$				

6 K-Means and Kernel (12 points)

Consider applying K-means++ to cluster a dataset of N samples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^D$. Instead of doing this directly, we apply a feature map $\boldsymbol{\phi} : \mathbb{R}^D \to \mathbb{R}^M$ to each example, and then apply K-means++. With a corresponding kernel function $k(\cdot, \cdot)$ for this feature map, you need to follow the steps below to show that this algorithm can be efficiently implemented (that is, without operating in space $\mathbb{R}^M$).

6.1 First, recall that in the initialization step of K-means++, we need to randomly select K centers based on squared L2 distances, which requires calculating $\|\boldsymbol{\phi}(\boldsymbol{x}_n) - \boldsymbol{\phi}(\boldsymbol{x}_m)\|_2^2$ for some $n, m \in [N]$. Express this quantity using the kernel function only (show your derivation). (2 points)

6.2 Suppose that $\mathcal{S} \subset [N]$ contains a nonempty subset of examples belonging to the same cluster in some iteration of K-means++. In the next iteration, one needs to compute the squared distance between an arbitrary example $\boldsymbol{\phi}(\boldsymbol{x}_n)$ and the center of this cluster $\boldsymbol{\mu} = \frac{1}{|\mathcal{S}|} \sum_{m \in \mathcal{S}} \boldsymbol{\phi}(\boldsymbol{x}_m)$, that is, $\|\boldsymbol{\phi}(\boldsymbol{x}_n) - \boldsymbol{\mu}\|_2^2$. Once again, express this quantity using the kernel function only (show your derivation). (4 points)

6.3 Based on your answers from the last two questions, fill in the missing details in Algorithm 3. More specifically,

- complete the for loop in Line 2 which finds the initial center indices $n_2, \ldots, n_K \in [N]$;
- complete the for loop in Line 5 which finds the new partition $\mathcal{S}'_1, \ldots, \mathcal{S}'_K$ based on $\mathcal{S}_1, \ldots, \mathcal{S}_K$.

Note that we have pre-computed the Gram matrix $\boldsymbol{M}$ as an input, so you can directly use $M_{n,m}$ whenever you need $k(\boldsymbol{x}_n, \boldsymbol{x}_m)$. (6 points)

Algorithm 3: K-means++ with kernel

Input: dataset $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ with Gram matrix $\boldsymbol{M}$
Output: a partition of the dataset $\mathcal{S}_1, \ldots, \mathcal{S}_K \subset [N]$
Initialize:
1 Uniformly at random select an index $n_1 \in [N]$ as the first center.
2 **for** $k = 2, \ldots, K$ **do**
3 Set $\mathcal{S}_k = \{n_k\}$ for all $k \in [K]$
Repeat until convergence:
4 Set $\mathcal{S}'_k = \emptyset$ for all $k \in [K]$ ▷ initialize the partition as empty sets
5 **for** $n = 1, \ldots, N$ **do**
6 Set $\mathcal{S}_k = \mathcal{S}'_k$ for all $k \in [K]$ ▷ overwrite $\mathcal{S}_1, \ldots, \mathcal{S}_K$ with the new partition.

7 EM for Clustering (16 points)

Suppose that Bob runs a website with N registered users and has collected the number of active days for each user in the past m days: $x_1, \ldots, x_N \in \{0, 1, \ldots, m\}$ (where x_n is the number of active days for user n). He suspects that there are two types of users: *casual* users and *engaged* users, and wants to use a clustering algorithm to help find these two clusters. He realizes that K-means probably does not work well for this one-dimensional discrete data. Instead, inspired by the Gaussian Mixture Model, he comes up with the following Binomial Mixture Model with three parameters $\omega, \theta_c, \theta_e \in [0, 1]$. Specifically, he assumes that the data are generated independently in the following way for each n:

- First, a hidden variable indicating the type of the user $z_n \in \{c, e\}$ is drawn so that $z_n = c$ (casual user) with probability ω and $z_n = e$ (engaged user) with probability $1 - \omega$.
- Then, the number of active days x_n for this user is drawn from a binomial distribution with parameter θ_{z_n}, that is, the probability of x_n being $r \in \{0, 1, \ldots, m\}$ is

$$\mathrm{Bin}(r, m, \theta_{z_n}) = \binom{m}{r} \theta_{z_n}^r (1 - \theta_{z_n})^{m-r}. \tag{1}$$

  (This is the same as assuming that on each of the m days, the user is active with probability θ_{z_n}.)

Follow the steps below to help Bob derive the EM algorithm for learning this Binomial Mixture Model.

7.1 E-step 1: fixing the parameter $\omega, \theta_c, \theta_e$, derive the posterior distribution $q_n(z) \triangleq p(z_n = z \mid x_n; \omega, \theta_c, \theta_e)$ for $z \in \{c, e\}$. Show your derivation, which can use the propositional sign, but express your final answer WITHOUT using it. Feel free to use the binomial mass function $\mathrm{Bin}(\cdot, \cdot, \cdot)$ from Eq. (1) without plugging in its definition. (5 points)

7.2 E-step 2: write down the expected complete log-likelihood $Q(\omega, \theta_c, \theta_e)$. Express it in terms of q_n, x_n (for $n = 1, \ldots, N$), m, and the model parameters. This time, you have to plug in definition of the binomial mass function from Eq. (1) (which helps you solve the next question). (3 points)

7.3 M-step: find the maximizer of $Q(\omega, \theta_c, \theta_e)$ by setting its gradient to $\mathbf{0}$. Express it in terms of q_n, x_n (for $n = 1, \ldots, N$), and m. Show your derivation. (8 points)

You can use this page for scratch work

You can use this page for scratch work

You can use this page for scratch work

You can use this page for scratch work

You can use this page for scratch work