6.867: Exam 2, Fall 2017

Solutions

These are not the **only** acceptable answers. Some other answers also received credit.

Answer the questions in the spaces provided. Show your work neatly. **We will only grade answers that appear in the answer boxes or on answer lines.**

If a question seems vague or under-specified to you, make an assumption, write it down, and solve the problem given your assumption.

You may prepare and use both sides of one 8.5 inch x 11 inch sheet of paper upon which you may write/print anything you like. You may not use any electronic device or any other resource other than your two-sided sheet of paper.

Write your name on every page.

Come to the front if you need to ask a question.

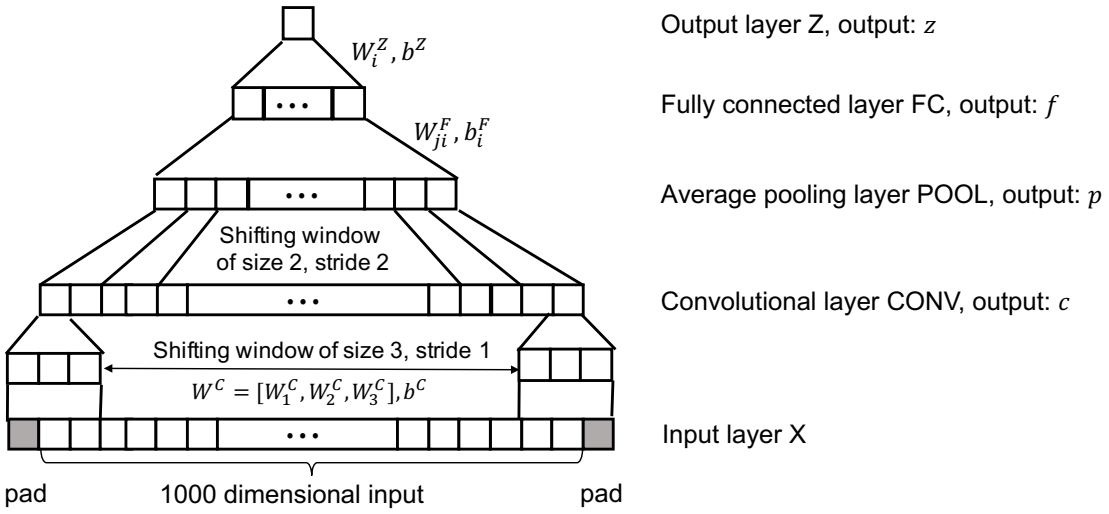
Name: _____ MIT Email: _____

Question	Points	Score
1	35	
2	30	
3	35	
Total:	100	

Name: _____

Neural Nets

1. (35 points) In this question we attempt to use a convolutional neural network (CNN) on time-series binary classification instead of 2D images, and do convolution in time instead of space.



The network takes in input of size 1×1000 (1 value per timestep for 1000 timesteps), and outputs a single value representing the probability of labeling $Y = 1$. The network consists of

- An input layer X, a 1×1000 vector.
- A convolutional layer CONV, with one feature map, a window width of 3, stride of 1 and padding of 1. ReLU is used for this layer. Denote the weight as $W^C = [W_1^C, W_2^C, W_3^C]$ and bias as b^C (scalar). Note that padding is applied on both ends.
- An average-pooling layer POOL that outputs the *average* value of all values in its window. It has window size of 2 and stride of 2. Note that a pooling layer has no trainable parameters.
- A fully connected layer FC with 20 units. ReLU is used for this layer. Denote the weight that connects the j-th unit in POOL to the i-th unit in FC by W_{ji}^F and the bias of the i-th unit in FC by b_i^F.
- An output layer Z outputing a single number. Sigmoid function σ is used for this layer. Denote the weight that connects the i-th unit in FC to Z by W_i^Z and the bias for Z by b^Z (scalar).

(a) (3 points) How many units are there in the CONV layer?

Solution: $(1000 + 2 \times 1) - 3 + 1 = 1000$

(b) (2 points) How many units are there in the POOL layer?

Solution: $1000/2 = 500$

Name: _____

Consider a training case with input X and label $Y = 1$. Let the outputs of the layers in the network be c, p, f and z for the CONV, POOL, FC and Z layers, respectively. Since we are using only one channel in the network, you may treat any of these variables as 1-D vectors and use subscript such as c_i, f_i, etc. to denote the i-th entry of the corresponding vector (i starts from 1). You may write your results with these variables without explicitly computing them.

We use binary cross-entropy loss for network training

$$\text{cost}(z, Y) = -[Y \log(z) + (1 - Y) \log(1 - z)] \tag{1}$$

(c) (6 points) Write $\partial \text{cost}/\partial f_i$ for this training case in terms of network parameters and final output z. Recall that $Y = 1$ for this training case.
Hint: Recall that the gradient of a sigmoid function $\sigma(x)$ is given by $\sigma'(x) = \sigma(x)(1-\sigma(x))$,

Solution:

$$\begin{aligned}\frac{\partial \text{cost}}{\partial f_i} &= -\frac{\partial \log(z)}{\partial f_i} = -\frac{1}{z}\frac{\partial (z)}{\partial f_i} = -\frac{1}{z} \cdot z(1-z) \cdot \frac{\partial (\sum_{i'=1}^{20} W_{i'}^Z f_{i'} + b^Z)}{\partial f_i} \\ &= (z-1)W_i^Z\end{aligned}$$

(d) (5 points) Write $\partial f_i/\partial p_j$ for this training case in terms of f and network parameters.
Hint: The gradient of ReLU activation can be written as $\text{ReLU}'(x) = s(\text{ReLU}(x))$, where $s(\cdot)$ is a step function that is 1 when the input is positive and 0 otherwise.

Solution:

$$\begin{aligned}\frac{\partial f_i}{\partial p_j} &= \frac{\partial \text{ReLU}(\sum_{j'} W_{j'i}^F p_{j'} + b_i^F)}{\partial p_j} \\ &= s(f_i)\frac{\partial (\sum_{j'} W_{j'i}^F p_{j'} + b_i^F)}{\partial p_j} \\ &= s(f_i) W_{ji}^F\end{aligned}$$

(e) (5 points) Compute $\partial p_j/\partial c_k$ for this training case. Remember, the indexing starts at 1.

Solution: For average pooling, $\frac{\partial p_j}{\partial c_k} = 0.5$ if $k = 2j$ or $k = 2j - 1$, otherwise it's 0.

(f) (8 points) Given all the answers above, compute $\partial cost/\partial W_2^C$ for this training case. For clarity, you should leave your answers in summation form (using two summations or three summations are acceptable), and you need to specify the lower bound and upper bound of any summation you use in your answers.

Solution: For a conv layer with window width of 3 and padding 1, we have

$$c_k = \mathrm{ReLU}(W_1^C X_{k-1} + W_2^C X_k + W_3^C X_{k+1} + b^C)$$

where $X_0 = X_{1001} = 0$ are the padded zeros. Therefore, we always have $\frac{\partial c_k}{\partial W_2^C} = s(c_k)X_k$.

$$\begin{aligned}
\frac{\partial cost}{\partial W_2^C} &= \sum_{i=1}^{20}\sum_{j=1}^{500}\sum_{k=1}^{1000} \frac{\partial cost}{\partial f_i}\frac{\partial f_i}{\partial p_j}\frac{\partial p_j}{\partial c_k}\frac{\partial c_k}{\partial W_2^C} \\
&= \sum_{i=1}^{20}\sum_{j=1}^{500} (z-1)W_i^Z \cdot s(f_i)W_{ji}^F \cdot 0.5\left(\frac{\partial c_{2j-1}}{\partial W_2^C} + \frac{\partial c_{2j}}{\partial W_2^C}\right) \\
&= \sum_{i=1}^{20}\sum_{j=1}^{500} (z-1)W_i^Z \cdot s(f_i)W_{ji}^F \cdot 0.5(s(c_{2j-1})X_{2j-1} + s(c_{2j})X_{2j})
\end{aligned}$$

(g) (6 points) Check true or false for each of the following statements.

- **T**( ) **F**( ) Convolutional neural networks can be seen as special cases of fully-connected feedforward neural networks.
- **T**( ) **F**( ) Although feedforward neural networks must include non-linear activation functions to be effective, non-linear activation functions are not necessary in RNNs.
- **T**( ) **F**( ) If batch normalization is applied to the output layer, it reduces internal covariate shift.

Solution: T F F

Name: _____

Questions and Answers

2. (30 points) Consider an online survey based on a collection of yes-or-no questions (e.g., "is 6.867 at MIT a class on machine learning?"), where each question is given to a potentially different subset of people. Each person's response is recorded as a "yes" (+1) or a "no" (-1). The survey responses are *noisy* in that they do not always match the *true* answer, which we assume always exist. Furthermore, different people have different likelihoods of correctly answering the questions, depending on their knowledge/skill levels. *In the following, we guide you through a sequence of steps for developing an unsupervised method that infers true answers from noisy survey responses, while accounting for personal differences in response accuracy.*

Model description. Let N be the number of questions and M be the number of people. We use $[n]$ to denote the set $\{1, \ldots, n\}$. In addition, we introduce the following notation:

- $E_{ij} \in \{0, 1\}$: whether question $i \in [N]$ was given to person $j \in [M]$ ($E_{ij} = 1$) or not ($E_{ij} = 0$),
- $A_{ij} \in \{-1, +1\}$: the response of person j to question i for every (i, j) where $E_{ij} = 1$,
- $T_i \in \{-1, +1\}$: the true answer of question $i \in [N]$,
- $\pi_j \in [0, 1]$: the probability of person $j \in [M]$ correctly answering any given question ("response accuracy")

In our model, both T_i's and π_j's are latent variables, and only A_{ij}'s are observed. Note that E_{ij}'s are known constants. We use bold symbols to denote whole sets of variables; i.e., $\mathbf{A} = \{A_{ij} : E_{ij} = 1\}$, $\mathbf{T} = \{T_1, \ldots, T_N\}$, and $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_M\}$.

For your convenience, we also provide the following notation:

- $P(i) = \{j : E_{ij} = 1\}$: the set of people who were given question $i \in [N]$,
- $Q(j) = \{i : E_{ij} = 1\}$: the set of questions given to person $j \in [M]$,
- $S = \{(i, j) : E_{ij} = 1\}$: all pairs of (i, j) where question $i \in [N]$ was given to person $j \in [M]$.

We assume a model that defines the joint distribution as

$$p(\mathbf{A}, \mathbf{T}, \boldsymbol{\pi}) = \prod_{i=1}^{N} p(T_i) \prod_{j=1}^{M} p(\pi_j) \prod_{(i,j) \in S} p(A_{ij} | T_i, \pi_j),$$

where $p(T_i)$ and $p(\pi_j)$ are both uniform distributions for all i and j, and

$$p(A_{ij} | T_i, \pi_j) = \begin{cases} \pi_j & \text{if } A_{ij} = T_i, \\ 1 - \pi_j & \text{otherwise,} \end{cases}$$

for each $(i, j) \in S$.

(a) (7 points) Write down a closed-form expression for the objective function maximized by the *maximum a posteriori* (MAP) inference of $\boldsymbol{\pi}$ and $\mathbf{T}$.
Your answer should be in terms of A_{ij}'s, T_i's, and π_j's and not include any $p(\cdot)$ terms. You may want to use the indicator function $\mathbf{1}\{\cdot\}$ that takes the value of one if the predicate inside the braces is true, and zero otherwise.

Solution: MAP estimation maximizes the posterior probability given as

$$\begin{aligned} p(\mathbf{T}, \boldsymbol{\pi} | \mathbf{A}) &\propto p(\mathbf{T}) p(\boldsymbol{\pi}) p(\mathbf{A} | \mathbf{T}, \boldsymbol{\pi}) \\ &\propto \prod_{(i,j) \in S} p(A_{ij} | T_i, \pi_j) \\ &= \prod_{(i,j) \in S} (\mathbf{1}\{A_{ij} = T_i\} \pi_j + \mathbf{1}\{A_{ij} \neq T_i\}(1 - \pi_j)). \end{aligned}$$

Any equivalent solution that leads to the same optimal solution (e.g., with constant factor difference or log-transformation) is accepted.

While solving the MAP inference problem in (a) is generally difficult, we can still use techniques from class to tackle this problem as follows.

(b) (7 points) *Known response accuracies, unknown true answers.* Suppose $\boldsymbol{\pi}$ is known. Write down a closed-form expression for the optimal T_i^* that maximizes the objective in (a) as a function of A_{ij}'s and π_j's. You may use the indicator function $\mathbf{1}\{\cdot\}$.

Solution: Let

$$f(a) = \prod_{j \in P(i)} (\mathbf{1}\{A_{ij} = a\} \pi_j + \mathbf{1}\{A_{ij} \neq a\}(1 - \pi_j))$$

for $a \in \{-1, +1\}$.

$$T_i^* = 2\mathbf{1}\{f(+1) \geqslant f(-1)\} - 1$$

(c) (7 points) *Known true answers, unknown response accuracies.* Suppose $\mathbf{T}$ is known. Write down a closed-form expression for the optimal π_j^* that maximizes the objective in (a) as a function of A_{ij}'s and T_i's. You may use the indicator function $\mathbf{1}\{\cdot\}$.

Solution: Note that

$$\prod_{i \in Q(j)} (\mathbf{1}\{A_{ij} = T_i\} \pi_j + \mathbf{1}\{A_{ij} \neq T_i\}(1 - \pi_j))$$

is the term in the MAP objective relevant for finding π_j^*. This can be alternatively expressed as

$$\pi_j^C (1 - \pi_j)^{|Q(j)| - C},$$

where $C = \sum_{i \in Q(j)} \mathbf{1}\{A_{ij} = T_i\}$. This is equivalent to the likelihood of observing C successes under a multinomial distribution with success probability π_j and $|Q(j)|$ trials. The ML estimate for π_j in this case, which corresponds to our π_j^*, is given as

$$\pi_j^* = \frac{C}{|Q(j)|}.$$

(d) (7 points) Using the results from (b) and (c), write down a pseudocode for an algorithm for the MAP estimation problem in (a) that jointly optimizes $\mathbf{T}$ and $\boldsymbol{\pi}$ given only $\mathbf{A}$. Make sure every step of the algorithm is clearly defined.

Solution:

1. Initialize $\boldsymbol{\pi}$ to random values in $[0, 1]$
2. Repeat until convergence (i.e., when the values of $\mathbf{T}$ and $\boldsymbol{\pi}$ stop changing):
   (a) Fix $\boldsymbol{\pi}$ and update $\mathbf{T}$ according to (b)
   (b) Fix $\mathbf{T}$ and update $\boldsymbol{\pi}$ according to (c)

(e) (2 points) Is your algorithm in (d) *guaranteed* to find a globally optimal solution? Check yes or no and provide a brief justification.

◯ Yes. Explanation: _____

_____

◯ No. Explanation: _____

_____

Solution: No. MAP inference in (a) is a non-convex problem with potentially many local optima (i.e., solutions where updating either $\boldsymbol{\pi}$ or $\mathbf{T}$ does not lead to a better solution). The inference algorithm in (d) finds a locally optimal solution, but it may not be globally optimal.

Name: _____

Quora LDA

3. (35 points) In this problem you will investigate how we can use variants of latent Dirichlet allocation (LDA) to learn a topic model for Quora, a popular question and answer website. Our corpus consists of M documents where each document consists of two sections. Each document 'm' has a question section with $N_m^{(q)}$ words and an answer section with $N_m^{(a)}$ words. We will be fitting a model with K topics.

   As with standard LDA, we assume a bag of words model, with a total vocabulary size of V, where there is a single global topic distribution θ_m for each document m (describing the theme of the whole document, including both question and answer) and where the topics for each word are conditionally independent of each other given θ_m. However, we recognize that *the distribution of words from a topic in the question section is likely to be very different from the distribution of words from the same topic in the answer section*. We will use this assumption to develop a variant of LDA, which we will refer to as Q/A LDA. We will also assume that all ϕ_k, the topic distributions over words, are known. The generative process for standard LDA is provided below for reference.

   Standard LDA generative process:

   1. For each document m choose $\theta_m \sim \mathrm{Dir}(\alpha)$
   2. For each word $w_{m,j}$ at position j of document m:
      (a) Choose a topic $z_{m,j} \sim \mathrm{Multi}(\theta_m)$
      (b) Choose a word $w_{m,j} \sim \mathrm{Multi}(\phi_{z_{m,j}})$

   In the following questions we will use bold notation to refer to a set of variables. For example $\mathbf{w}$ refers to all $w_{m,j}$. For some of the parts below it may be useful to recall that the Dirichlet density on n variables $(x_1, x_2, \ldots x_n)$ is defined as:

$$p(x_1, \ldots, x_n \,;\, \alpha_1, \ldots \alpha_n) = \frac{\Gamma(\sum_{i=1}^{n} \alpha_i)}{\prod_{i=1}^{n} \Gamma(\alpha_i)} \, x_1^{\alpha_1 - 1} x_2^{\alpha_2 - 1} \ldots x_n^{\alpha_n - 1} \tag{2}$$

   where $\sum_{i=1}^{n} x_i = 1$ and $x_i \geqslant 0 \;\; \forall i$.

   (a) (6 points) Provide the generative process for the new Q/A LDA model. You should use two different word-topic distributions–$\phi_k^{(q)}$ for the question section, and $\phi_k^{(a)}$ for the answer section, for each topic k.

   Solution:

   1. For each document m choose $\theta_m \sim \mathrm{Dir}(\alpha)$
   2. For each word m, j at position j of the questions section of document m:
      (a) Choose a topic $z_{m,j} \sim \mathrm{Multi}(\theta_m)$
      (b) Choose a word $w_{m,j} \sim \mathrm{Multi}(\phi_{z_{m,j}}^{(q)})$

3. For each word m, j at position j of the answers section of document m:
   (a) Choose a topic $z_{m,j} \sim \text{Multi}(\theta_m)$
   (b) Choose a word $w_{m,j} \sim \text{Multi}(\phi^{(a)}_{z_{m,j}})$

Solutions randomly generating the lengths of the question and answer sections (as opposed to treating them as known) were also accepted.

Next we will derive the Gibbs sampler for Q/A LDA. To get you started we have provided an expression for the full joint probability distribution of the standard LDA model:

$$p(\mathbf{w}, \mathbf{z}, \theta; \alpha, \boldsymbol{\phi}) = \left[\prod_{m=1}^{M} \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K} \prod_{k=1}^{K} \theta_{m,k}^{A_{m,k}+\alpha-1}\right]\left[\prod_{k=1}^{K}\prod_{v=1}^{V} \phi_{k,v}^{B_{k,v}}\right]$$

where:

- $A_{m,k} = \sum_j^{N_m} \mathbf{1}(z_{m,j} = k)$ is the number of words in document m assigned to topic k
- $B_{k,v} = \sum_m^M \sum_j^{N_m} \mathbf{1}(z_{m,j} = k)\mathbf{1}(w_{m,j} = v)$ is the number of words v assigned to topic k
- We have also assumed all elements of the vector $\boldsymbol{\alpha}$ are equal (ie. $\boldsymbol{\alpha} = [\alpha, \ldots, \alpha]^T$) for brevity and you may do the same.

(b) (5 points) Provide an expression for the joint distribution of the Q/A LDA model $p(\mathbf{w}, \mathbf{z}, \theta; \alpha, \boldsymbol{\phi}^{(q)}, \boldsymbol{\phi}^{(a)})$. You will need to use the variables $B^{(q)}_{k,v}$ and $B^{(a)}_{k,v}$ which are defined as $B_{k,v}$ above, but for words in the questions section and answers section respectively.

Solution:

$$p(\mathbf{w}, \mathbf{z}, \theta; \alpha, \boldsymbol{\phi}^{(q)}, \boldsymbol{\phi}^{(a)}) =$$
$$\left[\prod_{m=1}^{M} \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K} \prod_{k=1}^{K} \theta_{m,k}^{A_{m,k}+\alpha-1}\right]\left[\prod_{k=1}^{K}\prod_{v=1}^{V} (\phi_{k,v}^{(q)})^{B^{(q)}_{k,v}}\right]\left[\prod_{k=1}^{K}\prod_{v=1}^{V} (\phi_{k,v}^{(a)})^{B^{(a)}_{k,v}}\right]$$

(c) (5 points) Provide an expression for the Gibbs sampling update distribution for $z_{m,j}$ i.e. $p(z_{m,j} = k \mid w_{m,j} = v, \mathbf{z}_{\neg m,j}, \mathbf{w}_{\neg m,j}, \theta; \alpha, \boldsymbol{\phi}^{(q)}, \boldsymbol{\phi}^{(a)})$ where $\mathbf{z}_{\neg m,j}$ denotes the variable $\mathbf{z}$ without the (m, j) term and similarly for $\mathbf{w}_{\neg m,j}$.

Solution:

$$p(z_{m,j} = k \mid \ldots) \propto \begin{cases} \theta_{m,k}\phi^{(q)}_{k,v}, & \text{if } z_{m,j} \text{ is in question section} \\ \theta_{m,k}\phi^{(a)}_{k,v}, & \text{otherwise.} \end{cases}$$

(d) (5 points) Provide an expression for the Gibbs sampling update distribution for θ_m i.e. $p(\theta_m \mid \theta_{\neg m}, \mathbf{w}, \mathbf{z}; \alpha, \boldsymbol{\phi}^{(q)}, \boldsymbol{\phi}^{(a)})$ where $\theta_{\neg m}$ denotes the variable θ except the m term.

Solution:

$$p(\theta_m \mid \dots) = \mathrm{Dir}(A_{m,1} + \alpha, \dots, A_{m,K} + \alpha)$$

(e) One limitation of Q/A LDA is that we make a bag of words assumption, meaning we ignore the dependence of consecutive words. To address this you decide to combine your Q/A LDA model with a bi-gram model. This means each generated word $w_{m,j}$ is dependent on both its topic $z_{m,j}$ and the previous word $w_{m,j-1}$.

1. (6 points) Provide the generative process for generating a single document with this bi-gram extension, given that the document has a questions section and an answers section. *Hint*: You will have to redefine the parameters $\boldsymbol{\phi}^{(q)}$ and $\boldsymbol{\phi}^{(a)}$ that define a word-topic distribution.

   Solution: For each word w_j at position j of the questions section:

   (a) Choose a topic $z_j \sim \mathrm{Multi}(\theta)$

   (b) Choose a word $w_j \sim \mathrm{Multi}(\phi^{(q)}_{z_j, w_{j-1}})$

   For each word w_j at position j of the answers section:

   (a) Choose a topic $z_j \sim \mathrm{Multi}(\theta)$

   (b) Choose a word $w_j \sim \mathrm{Multi}(\phi^{(a)}_{z_j, w_{j-1}})$

   Where $\phi^{(q)}_{k,v}$ and $\phi^{(a)}_{k,v}$ are the distributions over words for topic k when the previous word is v for the questions and answers sections respectively.

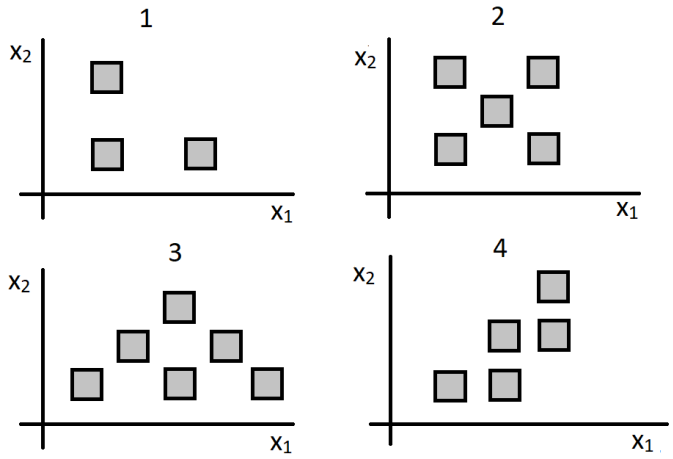
2. (4 points) A drawback of this bi-gram extension is that it increases the number of parameters. If we are using K topics, a vocabulary of size V, and the question and answer sections have distinct word-topic distributions, how many $\boldsymbol{\phi}$ parameters (where a parameter is defined as a scalar) will be required in this bi-gram model?

   Solution: $2KV^2$. It is also correct to say $2KV(V-1)$ since probabilities sum to 1. Solutions using $V+1$ instead of V (for a start token) were also accepted.

The last part of this question is not directly related to the LDA models from the previous parts of this question.

(f) (4 points) The following figure shows 4 distributions over the variables x_1 and x_2, where the shaded regions have non-zero probability. Select the distributions for which Gibbs sampling is **NOT** ergodic. (A process is *ergodic* if it always reaches a unique stable distribution that is independent of the initial state.)

Name: _____



☐ 1 √ **2** √ **3** ☐ 4