CS221 Exam 2

Spring 2021

Please read all of the following information before starting the exam:

- This test has 18 questions on 14 pages for a total of 80 possible points.
- **You will have 120 minutes to complete and submit the exam.** The exam has been designed to take around 80 minutes, and the extra 40 minutes are provided to allow for submission upload. **We will not be accepting any email submissions for this exam, so it is important that you get your work submitted on time.**
- Note that different questions are worth different amounts of points. Budget your time accordingly!
- Keep your answers precise and concise. We may award partial credit so show all your work clearly and in order.
- Don't spend too much time on one problem. Read through all the problems carefully and do the easier ones first.
- **If you are unsure about a problem statement when taking the exam, state your assumptions within your answer.** We will take all reasonable assumptions into account when grading.
- This exam is open-book; you may use any inanimate resources, including the course website.
- Being subject to the provisions of the Honor Code means in part that you must observe the rules established for this exam, which are: you may consult only inanimate sources. You may not consult or collaborate with anyone about the questions. Such collaboration is a violation of the Honor Code.
- Good luck!

Problem	Part	Max Score	Score
1	a	16	
	b	6	
	c	4	
	Total	**26**	
2	a	15	
	b	10	
	c	9	
	Total	**34**	
3	a	8	
	b	12	
	Total	**20**	

Total Score: $\boxed{} + \boxed{} + \boxed{} = \boxed{}$

0. Honor Code (*0 points*) Please write or type down the honor code below and sign your name. Your exam will not be graded if this question is not completed.

"I will not consult or collaborate with anyone about the questions. Such collaboration is a violation of the Honor Code."

1. Art Galleries (*26 points*)

Some artists have finished new paintings and are trying to display them. Luckily, some galleries are looking for new paintings to display. It is your job to match artists and galleries taking into account the preferences of the artists, the preferences of the galleries, and the capacity of each gallery.

Here is the formal art gallery matching problem setup:

1. There are m artists $A_1, ..., A_m$ who each have a single painting they would like displayed.
2. There are n galleries $G_1, ..., G_n$ that have space to display paintings.
3. Each artist A_i specifies arbitrary non-negative preferences $PA_1^{(i)}, ..., PA_n^{(i)} \geq 0$ for each of the n galleries. A large preference value of $PA_j^{(i)}$ means that artist A_i really wants their painting to be displayed in gallery G_j, and a preference value of 0 for $PA_j^{(i)}$ means that artist A_i does not want their painting to be displayed in gallery G_j.
4. Each gallery G_i specifies arbitrary non-negative preferences $PG_1^{(i)}, ..., PG_m^{(i)} \geq 0$ for each of the m artists. A large preference value of $PG_j^{(i)}$ means that gallery G_i really wants to display artist A_j's painting, and a preference value of 0 for $PG_j^{(i)}$ means that gallery G_i does not want to display artist A_j's painting

The art gallery matching process has the following requirements:

1. Each gallery G_i can have a maximum of 1 painting displayed.
2. Each artist must be matched to exactly one gallery for which they have specified a *positive* preference (assume each artist has at least one such preference) and for which the chosen gallery specifies a *positive* preference for the artist (assume each gallery has at least one such preference).

a. (*16 points*) We can model the art gallery matching process as a CSP. Our CSP should find the assignment with the *maximum weight* as determined by the product of the preference weights of the artists and galleries all together. There are two possible formulations of this CSP - one with m variables, one for each artist $A_1, ..., A_m$, and one with n variables, one for each gallery $G_1, ..., G_n$.

Finish the specification of this CSP for each of the formulations by stating the domains of each variable and the factors needed. You may define any notation/helper functions to help you concisely express your answers below.

Formulation 1: Artists as Variables (For this formulation, you should use only unary and binary factors.)

- **Variables (Already given):** We have m variables for the artists $A_1, ..., A_m$

- **Domains (how large is each and what are the values?):**

- **Factors (Use only unary and binary factors. State the arity of each and write them as functions from variables to scalars):**

Formulation 2: Galleries as Variables

- **Variables (Already given):** We have n variables for the galleries $G_1, ..., G_n$
- **Domains (how large is each and what are the values?):**

- **Factors (state the arity of each and write them as functions from variables to scalars):**

b. (*6 points*) Imagine a small setting with 3 artists A_1, A_2, A_3 and 3 galleries G_1, G_2, G_3.
The artist preferences are below:

	G_1	G_2	G_3
$PA^{(1)}$	0	3	0
$PA^{(2)}$	1	2	4
$PA^{(3)}$	3	2	1

The gallery preferences are below:

	A_1	A_2	A_3
$PG^{(1)}$	2	4	0
$PG^{(2)}$	1	4	5
$PG^{(3)}$	3	2	1

Assume that we are modeling the problem using the **first formulation** of the CSP and are using the artists as variables.
Apply the CSP you designed to this small setting and enforce arc-consistency amongst its variables. In particular, write out each variable and its **final** domain after arc consistency has been enforced. For example, if you have a variable X_i with a domain $\{a, b, c\}$, after enforcing arc-consistency, you should write $X_i : \{a, b, c\}$

c. (*4 points*) We will now return to the generalized version of the art gallery matching problem using the first formulation.

Circle all of the options below that would be applicable techniques for our art gallery matching CSP if we want a solution that is guaranteed to be the maximum weight solution.

- Least Constrained Value

- Most Constrained Variable
- Iterated Conditional Modes
- Backtracking Search

2. Bayesian networks (*34 points*)

a. (*15 points*) **True or False**

For the following statements about the Bayesian network with binary variables (in domain $\{0, 1\}$) as shown in Figure 1, please answer whether it's true or false, and justify your answer with 1-2 sentences.

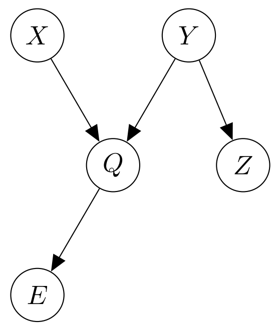

Figure 1: The Bayesian network for part (a).

(i) (3 points) **True/False:** If $Q = X$ OR Y, then the inequality $P(X = 1|Q = 1, Y = 1) \leq P(X = 1|Q = 1)$ always holds.

Justification:

(ii) (3 points) **True/False:** When doing inference for $P(Q = 1|E = 1)$ by converting the Bayesian network to a Markov network, we can remove variable Z.

Justification:

(iii) (3 points) **True/False:** When doing inference for $P(Q = 1|E = 1)$ by converting the Bayesian network to a Markov network, we can remove variable Y.

Justification:

(iv) (3 points) **True/False:** When doing inference for $P(Q = 1|E = 1)$ by converting the Bayesian network to a Markov network, the resulting Markov network has 3 unary factors and 2 binary factors.

Justification:

(v) (3 points) **True/False:** The Markov network constructed for inferring $P(Y = 1|Q = 1)$ is the same as $P(X = 1|Q = 1)$, as long as we remove all redundant variables.

Justification:

b. (*10 points*) **Gibbs Sampling**

Consider a Bayesian network with binary variables X, Q, E_A, E_B (Figure 2a). In order to compute the probability $P(Q = 1|E_A = 1, E_B = 0)$, you decide to first convert the Bayesian network into a Markov network (we did this for you, as shown in Figure 2b), and then use Gibbs sampling to do the inference.

The following parameters are known to you:

$$
\begin{array}{ll}
P(X = 1) = p & \\
P(Q = 1|X = 1) = \theta_1 & P(Q = 1|X = 0) = \theta_0 \\
P(E_A = 1|Q = 1) = \lambda_{A1} & P(E_A = 1|Q = 0) = \lambda_{A0} \\
P(E_B = 1|Q = 1) = \lambda_{B1} & P(E_B = 1|Q = 0) = \lambda_{B0}
\end{array}
$$

(a) The Bayesian network for part (b). (b) The Markov network converted from the Bayesian network in Figure 2a.

Figure 2: Part (b).

(i) [6 points] Suppose the initial values of the variables are set to $X = 1, Q = 0, E_A = 1, E_B = 0$. What is the probability $P(Q = 1|\text{everything else})$ for sampling variable Q in the first Gibbs update? Show your work.

(ii) [4 points] Using the Gibbs sampling algorithm above, can we also estimate $P(X = 1|E_A = 1, E_B = 0)$? Justify your answer with 1-2 sentences.

c. (*9 points*) **Learning**

Now suppose the parameters θ_0, θ_1 are unknown to you. Suppose we have a tiny dataset $\mathcal{D}$ which is assumed to be samples of the Bayesian network shown in Figure 2a:

Sample ID	X	Q	E_A	E_B
1	1	0	0	1
2	1	1	0	1
3	0	1	1	1
4	1	0	1	0

Table 1: A tiny dataset $\mathcal{D}$ containing 4 samples.

(i) [5 points] From dataset $\mathcal{D}$, what is the maximum likelihood estimation (MLE) value for θ_0 and θ_1?

(ii) [4 points] Now let's consider a more challenging scenario – your computer is hacked, and the X column in your dataset $\mathcal{D}$ is lost forever. In that case, which of the following algorithms can be used for estimating θ_0 and θ_1? Select all that apply.

A. Count and normalize

B. Particle filtering

C. Expectation-Maximization

D. Gibbs sampling

E. Laplace smoothing

F. Forward-backward algorithm

G. None of the above

3. Logic (*20 points*)

a. (*8 points*) **Translating**

Translate each of the following sentences into first order logic using only the predicates listed below:

- Teacher(x): x is a teacher.
- Student(x): x is a student.
- Test(x): x is a test.
- Passed(x, y): x passed y.

(i) [2 point] Some students are also teachers.

(ii) [3 points] All students have failed a test.

(iii) [3 points] There is a test that every student has passed.

b. (*12 points*) **Knowledge Base**

Imagine we are building a knowledge base of propositions in first order logic and want to make inferences based on what we know. We will deal with a simple setting, where we only have three objects in the world: Alice, Carol, and Bob. Our predicates are as follows:

- Employee(x): x is an employee.
- Boss(x): x is a boss.
- Works(x): x works.
- Paid(x): x gets paid.

The knowledge base we have constructed consists of the following propositions:

1. Boss(Carol)
2. Employee(Bob)
3. Paid(Carol) $\land$ Works(Carol)
4. Paid(Alice)
5. $\forall x$ (Employee(x) $\leftrightarrow \neg$ Boss(x))
6. $\forall x$ (Employee(x) $\rightarrow$ Works(x))
7. $\forall x$ ((Paid(x) $\land \neg$ Works(x)) $\rightarrow$ Boss(x))

(i) [2 Point] We know from class that one technique we can use to perform inference with our knowledge base is to propositionalize the statements of first-order logic into statements of propositional logic. Practice this by propositionalizing statement (6) from our knowledge base.

(ii) [3 Points] If we translated the statement "Anyone who is not a boss either works or does not get paid" into first-order logic and added it to our knowledge base, how would the size of the set of valid models representing our knowledge base change, and why?

(iii) [7 Points] Using only our original knowledge base (not including the statement from part (ii)), we want to answer the question "Does everyone work?" We first translate the sentence "everyone works" into first order logic as statement f. Determine the answer to our query by considering the following questions of satisfiability:

① [3 points] Is KB $\cup \neg f$ satisfiable? Answer yes/no. If yes, fill in the following table with T for true and F for false to show that there is a satisfying model.

x	Employee(x)	Boss(x)	Works(x)	Paid(x)
Alice				
Bob				
Carol				

② [3 points] Is KB $\cup f$ satisfiable? Answer yes/no. If yes, fill in the following table with T for true and F for false to show that there is a satisfying model.

x	Employee(x)	Boss(x)	Works(x)	Paid(x)
Alice				
Bob				
Carol				

③ [1 points] Based on your answers to the previous two parts, does our knowledge base entail f, contradict f, or is f contingent? And what should the answer to our original question "Does everyone work?" be?