청년 AI/BigData아카데미 15기 [인공지능] 과목 시험

출제자: 백성빈 ([sungbin0515@postech.ac.kr](mailto:sungbin0515@postech.ac.kr))

2021년 9월 16일 ~ 2021년 9월 17일

* 시험 시간은 9월 16일 18:00 ~ 9월 17일 23:59 (29시간 59분) 입니다.
* 시험 문제는 제출 점수 문제를 제외하면 총 10문제(서브 문제 포함)이며, 난이도에 차이가 있습니다. (난이도 순서로 배열된 것은 아닙니다)
* **시험지 파일에** 답안을 작성하여 PLMS에 제출하시면 됩니다. **풀이는 적지 않아도 됩니다**.
  + 수식이나 표 등을 작성하여 스캔한 파일을 활용하는 등 풀이의 일부를 스캔 자료로 대체하는 것은 가능하나, 시험지를 출력하여 답안을 작성한 후 이를 스캔하는 등의 방법으로 **제출하지는 말아주세요.**
* 제출 제한시간을 초과할 경우 초과한 시간에 따라 아래와 같은 페널티가 적용됩니다:
  + 0분 이상 15분 이하: 페널티 없음
  + 15분 초과 1시간 이하: 10% 페널티
  + 1시간 초과 12시간 이하: 40% 페널티
  + 12시간 초과 24시간 이하: 80% 페널티
  + **24시간 초과: 미제출 처리 (0점 처리)**
* 답안은 한글, 영어, 수식, 의사 코드(Pseudo code)를 모두 사용하여 작성하실 수 있으며, 필요하다면 넷을 섞어도 무방합니다.
  + 수식이나 의사 코드를 작성하는데 어려움이 있을 경우, 종이에 수식을 그려 이를 촬영한 그림 파일을 사용해도 무방합니다.
  + 위 네 가지 양식 뿐만 아니라 다른 방식을 사용하는 것이 가능합니다. 하지만 해당 방식을 통해 작성한 답안은 채점자가 번역기 등 외부 도구 없이 해석할 수 있어야 합니다.
* 시험과 관련하여 질문이 있을 경우, [sungbin0515@postech.ac.kr](mailto:sungbin0515@postech.ac.kr) 로 메일을 보내주세요.
* **오픈북** 온라인 시험입니다. 이론 및 실습 강의, 인터넷 등 원하는 자료를 자유로이 열람하는 것이 가능합니다. (다른 수강생 분들의 답이나 풀이, 설명 등은 **제외**합니다)
  + 다른 수강생 분들과 답을 공유하거나, 외부 자료를 단순 복사 붙여넣기 하지 마세요. (외부 자료를 보면서 자신이 이해한대로 다시 쓰는 정도는 괜찮습니다) **단순 복사 붙여넣기(표절)가 적발될 경우, 해당 수강생 분의 점수를 0점 처리합니다.**

0. [제출 점수 부여, 40pts]

(a) [40pts] 여러분의 이름을 적어주세요.

B반 4조 김민경

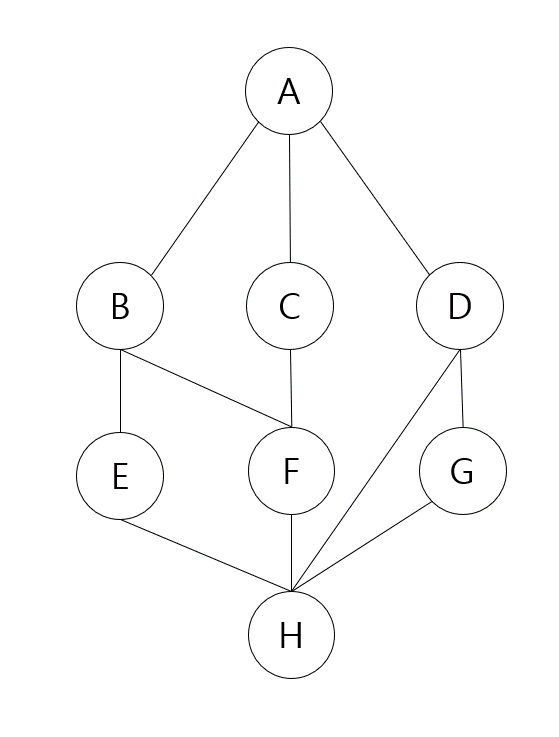
1. [Graph Search, 12pt].

(a) [6pts] BFS (Breadth-first search)의 정의를 서술하세요**.**

**BFS는 너비 우선 탐색 알고리즘입니다. 루트 노드에서 시작하여 같은 폭에 있는 가까운 노드를 먼저 탐색하면서 탐색하는 방법입니다. 알고리즘 방식을 설명하면 처음에 큐에 시작 노드를 넣습니다. 그 뒤로 시작 노드를 꺼내고 연관된 노드들을 큐에 삽입합니다. 여기서 방문을 하지 않은 노드를 삽입하는 것입니다. 이 방식을 큐가 비어질 때까지 반복합니다. 그러면 순서를 알 수 있습니다. BFS는 최단경로를 구할 때 주로 사용합니다.**

(b) [6pts] 아래 단순 무방향 그래프를 BFS 알고리즘을 사용하여 탐색했을 때, 탐색하는 Node의 순서를 하단의 표에 작성하세요. Node **A부터 시작합니다.**

(p.s. 가능한 답이 여러 개 입니다. 그 중 **하나만** 적으면 됩니다. 모든 가능한 경우를 찾는 것이 아닙니다.)



|  |  |
| --- | --- |
| 순서 | 방문 노드 |
| 1 | **A** |
| 2 | B |
| 3 | C |
| 4 | D |
| 5 | E |
| 6 | F |
| 7 | H |
| 8 | G |

2. [Adversarial Search, 12pt].

(a) [6pts] Adversarial Search(적대적 탐색)를 사용하기 적합한 게임의 예시를 **두(2)개** 이상 제시하세요.

5개의 연속된 돌을 만들어야 하는 오목게임이 있습니다.

유령을 피해서 먹이를 많이 먹어야 하는 팩맨 게임이 있습니다.

(b) [6pts] Agent P는 카드 가져가기 게임을 Agent K와 함께 하고 있습니다. Agent P는 Adversarial Search를 활용하여 게임을 진행하기 위해, 아래와 같이 Search Tree를 작성했습니다. Game에 대한 설명을 읽고, Search Tree에 기재되지 않은 Terminal state Value (Terminal Value)를 계산하여 **하단의 표**에 작성하세요*. Value는 Agent P가 게임을 통해 획득한* ***점수*** *값입니다.*

(p.s. Agent P는 아군, Agent K는 적군이라고 생각하면 됩니다)

|  |
| --- |
| **카드 가져가기 게임**  자신이 가지고 있는 숫자 카드에 적힌 숫자의 합이 더 큰 사람이 승리한다.  각 플레이어는 자신과 상대의 카드를 볼 수 있다.  자신의 차례가 되면 플레이어는 아무도 가져가지 않은 카드 중 하나를 **보고** 가져갈 수 있다.  두 사람이 5번씩 가져가기를 하면 종료되며, 카드에 적힌 숫자의 합이 더 큰사람이 승리한다.  승리한 사람은 **자신의 숫자 카드에 적힌 숫자의 합에서 상대의 숫자 카드에 적힌 숫자의 합을 뺀 값만큼의 점수를** 받는다.  **현재 상황은** 아래와 같다:  Agent P가 가지고 있는 카드: [3] [4] [2] [6] (합: 15)  Agent K가 가지고 있는 카드: [1] [5] [7] [0] (합: 13)  아무도 가져가지 않은 카드: [8] [9] [10]  현재는 Agent P의 차례이다. |

그림 카드 가져가기 게임의 규칙

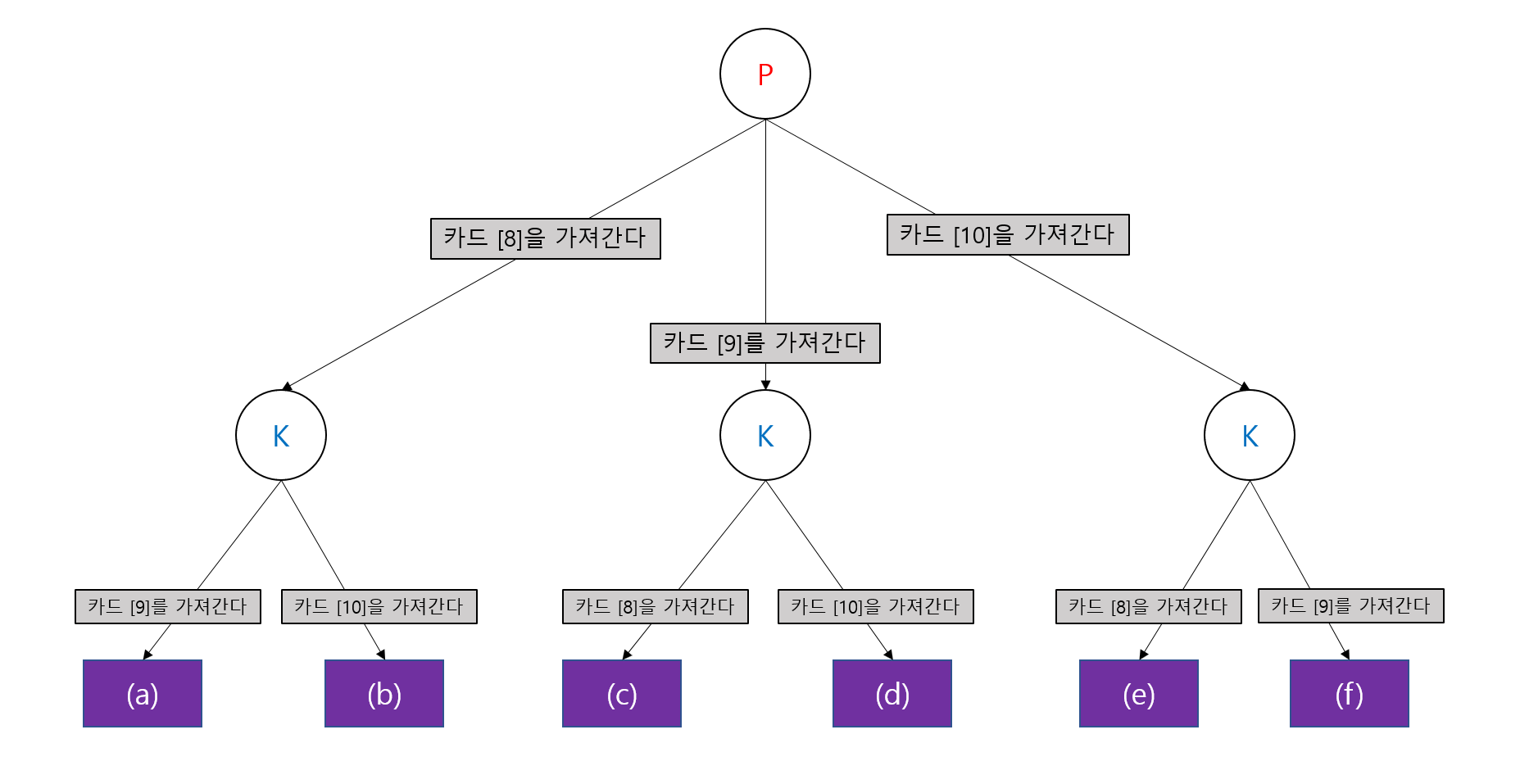


그림 . Search Tree

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 번호 | (a) | (b) | (c) | (d) | (e) | (f) |
| Terminal State Value | 1 | 0 | 3 | 1 | 4 | 3 |

3. [Markov Decision Process, 12pt]

(a). [6pts] Markov Decision Process를 구성하는 **모든** 요소의 이름과 그 설명을 작성하세요.

상태(set of state s) – 선택이 가능한 상태들의 집합을 의미합니다.

행동(set of actions a) – 상태집합에서 취할 수 있는 행동의 집합을 의미합니다.

보상(reward function R) – 상태가 s이고 행동 a를 취했을 때 받을 수 있는 보상의 값을 의미합니다.

정책(transition function T(s,a,s’)) – state에 action이 주어질 때 s’가 나올 확률을 의미합니다.

(b). [6pts] 아래 게임을 MDP로 작성한 내용을 참고하여, Optimal policy를 찾아서 서술하세요.

(**Deterministic한 Policy를 작성하시면 됩니다**. 즉, 현재 State에 따라 다음에 취할 Action이 **하나로 정해지는** Policy를 작성해 주세요.)

|  |
| --- |
| **미니 블랙잭**  숫자 1,2가 적힌 카드가 **무한히** 많은 카드 더미에서 카드를 한장씩 뽑는다. 각 숫자가 나올 확률은 동일하다.  매 턴마다 **카드를 한 장 뽑거나, 카드 뽑기를 멈출 수 있다.**  카드 뽑기를 멈출 경우, 플레이어에겐 아래와 같은 보상이 주어진다:  뽑은 카드에 적힌 숫자의 합이 5**보다 작을 경우**: 숫자의 합  뽑은 카드에 적힌 숫자의 합이 5**보다 크거나 같을 경우**: 0 |

State space = {S0, S1, S2, S3, S4, END, BUSTED}

* S0 ~ S4: 뽑은 카드에 적힌 숫자의 합이 0~4인 상태 (e.g. S2: 숫자의 합이 2인 상태)  
  BUSTED: 뽑은 카드에 적힌 숫자의 합이 5 이상인 상태 (점수를 받을 수 없는 상태)  
  END: 카드 뽑기를 중단하기를 선언한 상태

Action = {Draw, Stop}

* Draw: 카드 한장을 더 뽑는다  
  Stop: 카드 뽑기를 멈춘다

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s | a | s' | T(s,a,s') | R(s,a,s') |
| S0 | Draw | **S1** | 0.5 | 0 |
| S0 | Draw | **S2** | 0.5 | 0 |
| S1 | Draw | **S2** | 0.5 | 0 |
| S1 | Draw | **S3** | 0.5 | 0 |
| S2 | Draw | **S3** | 0.5 | 0 |
| S2 | Draw | **S4** | 0.5 | 0 |
| S3 | Draw | **S4** | 0.5 | 0 |
| S3 | Draw | **BUSTED** | 0.5 | 0 |
| S4 | Draw | **BUSTED** | 1 | 0 |
| S0 | Stop | **END** | 1 | 0 |
| S1 | Stop | **END** | 1 | 1 |
| S2 | Stop | **END** | 1 | 2 |
| S3 | Stop | **END** | 1 | 3 |
| S4 | Stop | **END** | 1 | 4 |
| BUSTED | Stop | **END** | 1 | 0 |

답은 하단의 표에 작성하시면 됩니다. 각 State별로 여러분의 Optimal Policy가 선택할 Action을

**Draw, Stop 중 하나를 선택하여** 작성하면 됩니다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| State | S0 | S1 | S2 | S3 | S4 |
| Policy | **Draw** | **Draw** | **Draw** | **Stop** | **Stop** |

4. [Probabilities in BNs, 12pt]

아래 제시된 특정 확률 변수에 대한 Bayes’ Net과 확률 분포 표를 활용하여 문제를 풀어주세요.

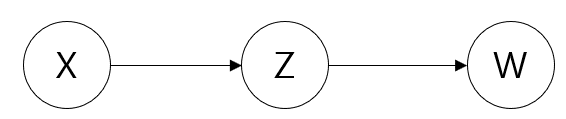


그림 Bayes' Net

|  |  |
| --- | --- |
| **X** | **P(X)** |
| +x | 0.4 |
| -x | 0.6 |

|  |  |  |
| --- | --- | --- |
| **X** | **Z** | **P(Z|X)** |
| +x | +z | 0.2 |
| +x | -z | 0.4 |
| -x | +z | 0.4 |
| -x | -z | 0.2 |

|  |  |  |
| --- | --- | --- |
| **Z** | **W** | **P(W|Z)** |
| +z | +w | 0.4 |
| +z | -w | 0.2 |
| -z | +w | 0.2 |
| -z | -w | 0.4 |

(a) [6pts] Bayes’ nets (Network)의 **정의**를 서술하세요. (구성 요소를 서술하는 것이 아닙니다)

확률 모델은 joint 확률을 구하는 것이 목적이지만 이것을 더 효율적이고 효과적으로 계산하기 위해서 독립인 사건을 이용하여 계산하는 것입니다.

(b) [6pts] 그림 3의 Bayes’ Net과 확률 분포 표를 바탕으로 아래 확률 함수의 값을 구하세요.

(Hint: Conditional Independence, Chain Rule)

1. **P (+x, -z, +w) = 0.032**
2. **P (-x, +z, -w) = 0.048**

5. [Independence in a BN, 12pt]

아래 제시된 특정 확률 변수에 대한 Bayes’ Net을 바탕으로 문제를 풀어주세요

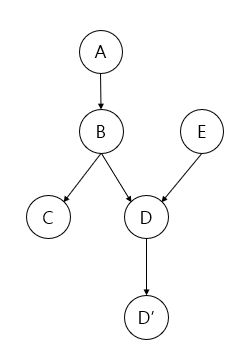


그림 Bayes' Net

(a) [6pts] D-separation을 통해 Bayes’ Net에서 어떤 것을 확인할 수 있는지 **하나** 서술하세요. (정의를 서술하는 것이 아닙니다)

그래프를 보고 의존(dependency)을 알 수 있습니다.

(b) [6pts] 아래 수식에서 중괄호 안에 들어갈 수 있는 확률 변수의 조합을 하나 기술하세요. 그림 4의 Bayes’ Net을 기준으로 합니다.

답이 두개 이상의 확률 변수의 조합일 경우, 각 확률 변수 사이를 콤마(쉼표)로 구분해주세요.

(Hint: D-separation)

**Ex.**  : {B}

1. **: D,B**
2. **: D**

수고하셨습니다!