Artificial Intelligence HW 1 - 2020085014 강준서 (Total 3 pages)

1.1 (a) Non-Singular, full rank, $|A| \neq 0$ (determinant is non-zero), A^{-1} exist

(b) $AA^T = I$ (orthogonal matrix property)

$\downarrow \times A^{-1}$

$A^{-1}(AA^T) = A^{-1}I \longrightarrow \underline{A^T = A^{-1}}$

1.2 (a) $P(A,C|B) = P(A|B)P(C|B)$ ($\because$ conditional independence)

$\frac{P(A,C|B)}{P(C|B)} = P(A|B)$ $\left(\frac{P(A,C|B)}{P(C|B)} = \frac{P(A,B,C)}{P(B)} \cdot \frac{P(B)}{P(B,C)} = \frac{P(A,B,C)}{P(B,C)} = P(A|B,C) \right)$

$\downarrow$ chain rule

$\underline{P(A|B,C) = P(A|B)}$

(b) $P(X)$: Bernoulli distribution, X : boolean

$$E_x[X] = \sum_x P(x) \cdot X$$

$$= \sum_x \phi^x (1-\phi)^{1-x} \cdot X$$

$$= (1-\phi) \cdot 0 + \phi \cdot 1 = \underline{\phi}$$

(c) $H(P) = -E_{x \sim P(x)} \log P(x)$

$D_{KL}(P \| Q) = E_{x \sim P(x)} (\log P(x) - \log Q(x))$

$$H(P,Q) = -E_{x \sim P(x)} \log Q(x) = -E_{x \sim P(x)} \log P(x) + E_{x \sim P(x)} (\log P(x) - \log Q(x))$$

$$= \underline{H(P) + D_{KL}(P \| Q)}$$

2.1 (a) Arad → Zerind → Sibiu → Timisoara → Oradea → Fagaras → Rimnicu Vilcea → Lugoj → Bucharest → Pitesti

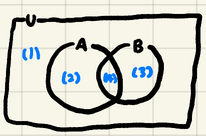
(b) Neamt → Iasi → Vaslui → Urziceni → Hirsova → Eforie → Bucharest → Giurgiu → Pitesti → Craiova → Drobeta → Mehadia → Lugoj → Timisoara → Arad → Zerind → Oradea → Sibiu

2.2 (a) Timisoara → Lugoj → Mehadia → Drobeta → Craiova → Pitesti → Bucharest

(b) Zerind → Arad → Oradea → Sibiu → Rimnicu Vilcea → Fagras → Pitesti → Bucharest

Zerind	Arad	Oradea	Sibiu	Rimnicu Vilcea	Fagras	Pitesti	Bucharest
0+374	75+366	71+380	215+253	295+193	314+176	392+100	493+0
374	441	451	468	488	490	492	493

3.1 (a)



$A = \{(2),(4)\}$, $B = \{(3),(4)\}$, $\neg A = \{(1),(3)\}$, $\neg B = \{(1),(2)\}$

$A \vee B = \{(2),(3),(4)\}$ $\underline{\neg A \wedge \neg B = \{(1)\}}$

$\underline{\neg(A \vee B) = \{(1)\}}$ $\therefore \neg(A \vee B) = \neg A \wedge \neg B$ always hold

(b)

A	B	A → B	B → A	A ↔ B
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

$A \to B = \{(T,T), (F,T), (F,F)\}$ $B \to A = \{(T,T), (T,F), (F,F)\}$

$\underline{A \leftrightarrow B} = \{(T,T), (F,F)\} = \underline{(A \to B) \wedge (B \to A)}$

3.2 (a) In case of model checking, we get the information we want by just making a truth table about all the propositions. So, it's easier to automatize and it's simple.

In contrast, propositional theorem proving only uses some propositions which we are "interested" in. Hence, it can be more efficent rather than model checking because we don't care about irrelevant propositions. (But not easy to automatize)

(b) $R_5 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$ biconditional elimination

$R_6 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$ And-Elimination

$R_7 : \neg B_{1,1} \rightarrow \neg (P_{1,2} \vee P_{2,1})$ contrapositive

$R_8 : \neg P_{1,2} \wedge \neg P_{2,1}$ Modus Ponens & De Morgan

$R_9 : \underline{\neg P_{2,1}}$ And-Elimination

(c) This will be attached at last page.

4.1 (a) JohnCalls and MaryCalls are conditionally independent given Alarm.

JohnCalls and Earthquake are conditionally independent given Alarm.

Burglary and MaryCalls are conditionally independent given Alarm.

(b) $P(E \mid A = true, J = true, M = true)$

$$P(e \mid a, j, m) = \alpha \sum_b P(b) P(e) P(a \mid b, e) P(j \mid a) P(m \mid a)$$

$$= \alpha P(e) P(j \mid a) P(m \mid a) \sum_b P(b) P(a \mid b, e)$$

$$= \underline{\alpha} P(e) \underline{P(j \mid a) P(m \mid a)} [P(b) P(a \mid b, e) + P(\neg b) P(a \mid \neg b, e)]$$

$$P(\neg e \mid a, j, m) = \underbrace{\alpha}_{\text{common part} \rightarrow \text{ignore}} P(\neg e) \underline{P(j \mid a) P(m \mid a)} [P(b) P(a \mid b, \neg e) + P(\neg b) P(a \mid \neg b, \neg e)]$$

$$\therefore P(e \mid a, j, m) = \frac{P(e)\ (0.001 \times 0.95 + 0.999 \times 0.29)}{P(e)(0.001 \times 0.95 + 0.999 \times 0.29) + P(\neg e)(0.001 \times 0.94 + 0.999 \times 0.001)}$$

$$= \frac{P(e) \times 0.29066}{P(e) \times 0.29066 + P(\neg e) \times 0.001939} = \underline{0.231 \ldots}$$

(c) $0.001 \times 0.002 \times 0.95 + 0.001 \times 0.998 \times 0.94 + 0.999 \times 0.002 \times 0.29 + 0.998 \times 0.999 \times 0.001$

$= \underline{0.002516442}$

(d) $$P(a \mid b, m) = \alpha \sum_e \sum_j P(b) P(e) P(a \mid b, e) P(j \mid a) P(m \mid a)$$

$$= \underline{\alpha P(b)} P(m \mid a) \sum_e P(e) P(a \mid b, e) \underbrace{\sum_j P(j \mid a)}_{=1}$$

$$P(\neg a \mid b, m) = \underbrace{\alpha P(b)}_{\text{common}} P(m \mid \neg a) \sum_e P(e) P(\neg a \mid b, e) \underbrace{\sum_j P(j \mid \neg a)}_{=1}$$

$$P(a \mid b, m) = \frac{0.7\ (0.002 \times 0.95 + 0.998 \times 0.94)}{0.7\ (0.002 \times 0.95 + 0.998 \times 0.94) + 0.01 \times (0.002 \times 0.05 + 0.998 \times 0.06)}$$

$$= \underline{0.99908 \ldots}$$

4.2 (a) $P(B = False, E = False, A = False, J = False, M = False)$

(b) The cases which we are interested in is when the alarm rang. So, we can reject 1, 5 in advance.

At remaining cases, {2, 3, 4}, a burglary had invaded only in 3.

$\therefore P(b \mid a) = \underline{\frac{1}{3}}$

HW1 3.2 (c)

B_{1,1}	B_{2,1}	P_{1,1}	P_{1,2}	P_{2,1}	P_{2,2}	R1	R2	R3	R4	KB
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE