IS-ZC444: ARTIFICIAL INTELLIGENCE

Lecture-12: Logical Agents, Bayesian Learning

Dr. Kamlesh Tiwari
Assistant Professor
Department of Computer Science and Information Systems,
BITS Pilani, Pilani, Jhunjhunu-333031, Rajasthan, INDIA

Nov 08, 2020 FLIPPED (WILP @ BITS-Pilani Jul-Nov 2020)

Recap: Propositional Logic

Propositions or **declarative sentences** can be true or false. Complex sentences could be formed by using $\neg$, $\wedge$, $\vee$ $\rightarrow$

Natural Deduction applies proof rules on sentences ϕ_1, ϕ_2, ..., ϕ_n to get new ψ; we denote it as $\phi_1, \phi_2, ..., \phi_n \vdash \psi$

$$\frac{\phi \qquad \psi}{\phi \wedge \psi} \wedge i \qquad \frac{\phi \wedge \psi}{\phi} \wedge e_1 \qquad \frac{\phi \wedge \psi}{\psi} \wedge e_2$$

$$\frac{\phi}{\neg\neg\phi} \neg\neg i \qquad \frac{\neg\neg\phi}{\phi} \neg\neg e \qquad \frac{\phi \qquad \phi \rightarrow \psi}{\psi} \rightarrow e$$

$$\frac{\phi \rightarrow \psi \qquad \neg\phi}{\neg\phi} MT \qquad \frac{\boxed{\begin{array}{c}\phi \\ \vdots \\ \psi\end{array}}}{\phi \rightarrow \psi} \rightarrow i$$

Box nesting is important

Logical Equivalences

$(\alpha \wedge \beta)$	=	$(\beta \wedge \alpha)$	Commutativity of $\wedge$
$(\alpha \vee \beta)$	=	$(\beta \vee \alpha)$	Commutativity of $\vee$

Logical Equivalences

$$
\begin{array}{lcll}
(\alpha \wedge \beta) & = & (\beta \wedge \alpha) & \text{Commutativity of } \wedge \\
(\alpha \vee \beta) & = & (\beta \vee \alpha) & \text{Commutativity of } \vee \\
(\alpha \wedge \beta) \wedge \gamma & = & \alpha \wedge (\beta \wedge \gamma) & \text{Associativity of } \wedge \\
(\alpha \vee \beta) \vee \gamma & = & \alpha \vee (\beta \vee \gamma) & \text{Associativity of } \vee
\end{array}
$$

Logical Equivalences

$(\alpha \wedge \beta)$	$=$	$(\beta \wedge \alpha)$	Commutativity of $\wedge$
$(\alpha \vee \beta)$	$=$	$(\beta \vee \alpha)$	Commutativity of $\vee$
$(\alpha \wedge \beta) \wedge \gamma$	$=$	$\alpha \wedge (\beta \wedge \gamma)$	Associativity of $\wedge$
$(\alpha \vee \beta) \vee \gamma$	$=$	$\alpha \vee (\beta \vee \gamma)$	Associativity of $\vee$
$\neg\neg\alpha$	$=$	α	Double negation elimination
$\alpha \rightarrow \beta$	$=$	$\neg\beta \rightarrow \neg\alpha$	Contraposition
$\alpha \rightarrow \beta$	$=$	$\neg\alpha \vee \beta$	Implication Elimination
$\alpha \leftrightarrow \beta$	$=$	$(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$	Biconditional Elimination

Logical Equivalences

$(\alpha \land \beta)$	$\equiv$	$(\beta \land \alpha)$	Commutativity of $\land$
$(\alpha \lor \beta)$	$\equiv$	$(\beta \lor \alpha)$	Commutativity of $\lor$
$(\alpha \land \beta) \land \gamma$	$\equiv$	$\alpha \land (\beta \land \gamma)$	Associativity of $\land$
$(\alpha \lor \beta) \lor \gamma$	$\equiv$	$\alpha \lor (\beta \lor \gamma)$	Associativity of $\lor$
$\neg\neg\alpha$	$\equiv$	α	Double negation elimination
$\alpha \to \beta$	$\equiv$	$\neg\beta \to \neg\alpha$	Contraposition
$\alpha \to \beta$	$\equiv$	$\neg\alpha \lor \beta$	Implication Elimination
$\alpha \leftrightarrow \beta$	$\equiv$	$(\alpha \to \beta) \land (\beta \to \alpha)$	Biconditional Elimination
$\neg(\alpha \land \beta)$	$\equiv$	$\neg\alpha \lor \neg\beta$	De Morgen
$\neg(\alpha \lor \beta)$	$\equiv$	$\neg\alpha \land \neg\beta$	De Morgen

Logical Equivalences

$(\alpha \wedge \beta)$	$=$	$(\beta \wedge \alpha)$	Commutativity of $\wedge$
$(\alpha \vee \beta)$	$=$	$(\beta \vee \alpha)$	Commutativity of $\vee$
$(\alpha \wedge \beta) \wedge \gamma$	$=$	$\alpha \wedge (\beta \wedge \gamma)$	Associativity of $\wedge$
$(\alpha \vee \beta) \vee \gamma$	$=$	$\alpha \vee (\beta \vee \gamma)$	Associativity of $\vee$
$\neg\neg\alpha$	$=$	α	Double negation elimination
$\alpha \rightarrow \beta$	$=$	$\neg\beta \rightarrow \neg\alpha$	Contraposition
$\alpha \rightarrow \beta$	$=$	$\neg\alpha \vee \beta$	Implication Elimination
$\alpha \leftrightarrow \beta$	$=$	$(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$	Biconditional Elimination
$\neg(\alpha \wedge \beta)$	$=$	$\neg\alpha \vee \neg\beta$	De Morgen
$\neg(\alpha \vee \beta)$	$=$	$\neg\alpha \wedge \neg\beta$	De Morgen
$\alpha \wedge (\beta \vee \gamma)$	$=$	$(\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$	Distribution of $\wedge$ on $\vee$
$\alpha \vee (\beta \wedge \gamma)$	$=$	$(\alpha \vee \beta) \wedge (\alpha \vee \gamma)$	Distribution of $\vee$ on $\wedge$

Soundness and Completeness

- Soundness: doing right
- Completeness: full coverage

There are 10 defective bulbs in a box of 25.

- Mr. A gives me 10 bulbs none of them is defective
- Mr. B gives me 20 bulbs; 5 of them is defective

Soundness and Completeness

- Soundness: doing right
- Completeness: full coverage

There are 10 defective bulbs in a box of 25.

- Mr. A gives me 10 bulbs none of them is defective
- Mr. B gives me 20 bulbs; 5 of them is defective

A is sound
B is complete

Evaluate a legal system “guilty until proven innocent” and “innocent until proven guilty”

Soundness and Completeness

- Soundness: doing right
- Completeness: full coverage

There are 10 defective bulbs in a box of 25.

- Mr. A gives me 10 bulbs none of them is defective
- Mr. B gives me 20 bulbs; 5 of them is defective

A is sound
B is complete

Evaluate a legal system “guilty until proven innocent” and “innocent until proven guilty”

What we want? both.

CNF, IMPL_FREE and NNF

Conjunctive normal form[1], implication free[2] and negative normal form[3]

Find CNF(NNF(IMPL_FREE(A)))

Where $A = \neg p \wedge q \rightarrow p \wedge (r \rightarrow q)$

[1] everything is conjunctions of disjunction

[2] no $\rightarrow$

[3] no double negation

CNF, IMPL_FREE and NNF

Conjunctive normal form[1], implication free[2] and negative normal form[3]

Find CNF(NNF(IMPL_FREE(A)))

Where $A = \neg p \wedge q \rightarrow p \wedge (r \rightarrow q)$

$$\neg(\neg p \wedge q) \vee (p \wedge (\neg r \vee q))$$
$$(p \vee \neg q) \vee (p \wedge (\neg r \vee q))$$
$$(p \vee \neg q \vee p) \vee (p \wedge \neg q \wedge \neg r \vee q)$$

[1] everything is conjunctions of disjunction

[2] no $\rightarrow$

[3] no double negation

Horn Clause

Formula that can be generated by H

$$\begin{aligned} P &::= \bot \mid \top \mid p \mid q \mid r \mid \ldots \\ A &::= P \mid P \wedge A \\ C &::= A \rightarrow P \\ H &::= C \mid C \wedge H \end{aligned} \tag{1}$$

Horn Clause

Formula that can be generated by H

$$
\begin{aligned}
P &::= \bot \mid \top \mid p \mid q \mid r \mid \ldots \\
A &::= P \mid P \wedge A \\
C &::= A \rightarrow P \\
H &::= C \mid C \wedge H
\end{aligned}
\tag{1}
$$

Satisfiability

1. It marks $\top$ if it occurs in that list.
2. If there is a conjunct $P_1 \wedge P_2 \wedge \cdots \wedge P_{k_i} \rightarrow P'$ of ϕ such that all P_j with $1 \leq j \leq k_i$ are marked, mark P' as well and go to 2. Otherwise (= there is no conjunct $P_1 \wedge P_2 \wedge \cdots \wedge P_{k_i} \rightarrow P'$ such that all P_j are marked) go to 3.
3. If $\bot$ is marked, print out 'The Horn formula ϕ is unsatisfiable.' and stop. Otherwise, go to 4.
4. Print out 'The Horn formula ϕ is satisfiable.' and stop.

Horn Clause

Formula that can be generated by H

$$\begin{aligned} P &::= \bot \mid \top \mid p \mid q \mid r \mid \ldots \\ A &::= P \mid P \wedge A \\ C &::= A \to P \\ H &::= C \mid C \wedge H \end{aligned} \tag{1}$$

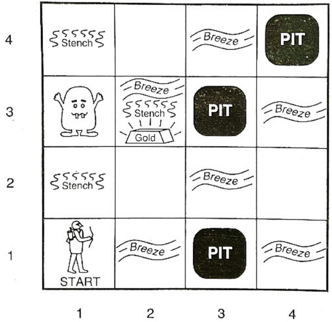
Satisfiability

1. It marks $\top$ if it occurs in that list.
2. If there is a conjunct $P_1 \wedge P_2 \wedge \cdots \wedge P_{k_i} \to P'$ of ϕ such that all P_j with $1 \leq j \leq k_i$ are marked, mark P' as well and go to 2. Otherwise (= there is no conjunct $P_1 \wedge P_2 \wedge \cdots \wedge P_{k_i} \to P'$ such that all P_j are marked) go to 3.
3. If $\bot$ is marked, print out 'The Horn formula ϕ is unsatisfiable.' and stop. Otherwise, go to 4.
4. Print out 'The Horn formula ϕ is satisfiable.' and stop.

(a) $(p \wedge q \wedge w \to \bot) \wedge (t \to \bot) \wedge (r \to p) \wedge (\top \to r) \wedge (\top \to q) \wedge (u \to s) \wedge (\top \to u)$
(b) $(p \wedge q \wedge w \to \bot) \wedge (t \to \bot) \wedge (r \to p) \wedge (\top \to r) \wedge (\top \to q) \wedge (r \wedge u \to w) \wedge (u \to s) \wedge (\top \to u)$
(c) $(p \wedge q \wedge s \to p) \wedge (q \wedge r \to p) \wedge (p \wedge s \to s)$
(d) $(p \wedge q \wedge s \to \bot) \wedge (q \wedge r \to p) \wedge (\top \to s)$
(e) $(p_5 \to p_{11}) \wedge (p_2 \wedge p_3 \wedge p_5 \to p_{13}) \wedge (\top \to p_5) \wedge (p_5 \wedge p_{11} \to \bot)$
(f) $(\top \to q) \wedge (\top \to s) \wedge (w \to \bot) \wedge (p \wedge q \wedge s \to \bot) \wedge (v \to s) \wedge (\top \to r) \wedge (r \to p)$

Recall Wumpus World

- **Performance** gold +100, death -100, step -1, arrow -10
- **Environment** smell around wumpus, breeze around pit
- **Actuator** turn left/right, forward, grab, release, shoot
- **Sensor** breeze, glitter, smell, bump, scream



Single Agent, Deterministic, Static, Discrete, !Observable & !Episodic

- $P_{x,y}$ if there is a pit in $[x, y]$
- $W_{x,y}$ if wumpus is in $[x, y]$
- $B_{x,y}$ if breeze is in $[x, y]$
- $S_{x,y}$ if stench is in $[x, y]$

We know R_1: $\neg P_{1,1}$, R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$,
R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$, R_4: $\neg B_{1,1}$, R_5: $B_{2,1}$

Model Checking for Inference

- Seven symbols $P_{1,1}$, $B_{1,1}$, $P_{1,2}$, $P_{2,1}$, $B_{2,1}$, $P_{2,2}$, $P_{3,1}$ have $2^7 = 128$ models. In three of these knowledge base is true.

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	R_1	R_2	R_3	R_4	R_5	KB
false	*false*	*false*	*false*	*false*	*false*	*false*	*true*	*true*	*true*	*true*	*false*	*false*
false	*false*	*false*	*false*	*false*	*false*	*true*	*true*	*true*	*false*	*true*	*false*	*false*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
false	*true*	*false*	*false*	*false*	*false*	*false*	*true*	*true*	*false*	*true*	*true*	*false*
false	*true*	*false*	*false*	*false*	*false*	*true*	*true*	*true*	*true*	*true*	*true*	<u>*true*</u>
false	*true*	*false*	*false*	*false*	*true*	*false*	*true*	*true*	*true*	*true*	*true*	<u>*true*</u>
false	*true*	*false*	*false*	*false*	*true*	*true*	*true*	*true*	*true*	*true*	*true*	<u>*true*</u>
false	*true*	*false*	*false*	*true*	*false*	*false*	*true*	*false*	*false*	*true*	*true*	*false*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
true	*true*	*true*	*true*	*true*	*true*	*true*	*false*	*true*	*true*	*false*	*true*	*false*

In all those three $\neg P_{1,2}$ is true, hence there is no pit in [1,2].
On the other hand $P_{2,2}$ is true on two and false in one so it is not confirmed whether there is pit in [2,2] or not.

Validity and Satisfiability

- **Validity**: sentence is true in all models (tautologies)

$$A \vee \neg A$$
$$A \vee B \rightarrow A \vee B$$

- **Satisfiability**: sentence is true in some models

$$A \vee \neg B$$
$$A \rightarrow B$$

Determine whether following sentence is valid or satisfiable

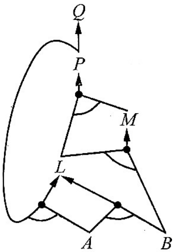
$$((A \wedge B) \rightarrow C) \leftrightarrow (A \rightarrow (B \rightarrow C))$$

Forward Chaining

Determines if a single proposition symbol *q* is entailed by the knowledge? (data driven reasoning)

- It begins from known facts and adds conclusions of the implication whose all the premises are known
- for $L_{1,1} \wedge \textit{breeze} \rightarrow B_{1,1}$ if we know $L_{1,1}$ and *breeze* then $B_{1,1}$ is added in knowledge base [4]

$P \Rightarrow Q$
$L \wedge M \Rightarrow P$
$B \wedge L \Rightarrow M$
$A \wedge P \Rightarrow L$
$A \wedge B \Rightarrow L$
A
B



- Applies Modus Ponens

$$\frac{\phi \qquad \phi \rightarrow \psi}{\psi}$$

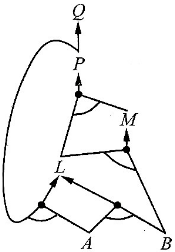
[4] $L_{1,1}$: location is [1,1]

Forward Chaining

Determines if a single proposition symbol q is entailed by the knowledge? (data driven reasoning)

- It begins from known facts and adds conclusions of the implication whose all the premises are known
- for $L_{1,1} \wedge \textit{breeze} \rightarrow B_{1,1}$ if we know $L_{1,1}$ and *breeze* then $B_{1,1}$ is added in knowledge base [4]

$P \Rightarrow Q$
$L \wedge M \Rightarrow P$
$B \wedge L \Rightarrow M$
$A \wedge P \Rightarrow L$
$A \wedge B \Rightarrow L$
A
B



- Applies Modus Ponens

$$\frac{\phi \quad \phi \rightarrow \psi}{\psi}$$

- An and-or tree gets constructed

[4] $L_{1,1}$: location is [1,1]

Backward Chaining

- Works backward from query
- If query Q is known to be true, then no work is needed.
- Otherwise, find those implications whose conclusion is Q
- If all the premises of one of those implications can be proven true (by backward chaining) then Q is true

$$P \Rightarrow Q$$
$$L \wedge M \Rightarrow P$$
$$B \wedge L \Rightarrow M$$
$$\boxed{A \wedge P \Rightarrow L}$$
$$A \wedge B \Rightarrow L$$
$$A$$
$$B$$

- test(Q) is it true ?
- test(P) is it true ?
- test($L \wedge M$) ?
- ((test($A \wedge B$) or test($A \wedge P$)) and test($B \wedge L$) ? we know A and B so we have L this gives M
- Therefore P and hence Q

First Order Logic (Predicate Logic)

- We have **constants**, **variables**, **predicates** and **functions**
- Here $P(x)$ could means $\forall x$ we have $P(x)$ or $\exists x$ such that $P(x)$
- Variable x has a domain from where it gets values
- $\forall x, \exists y P(x, y)$ is not always same as $\exists y, \forall x P(x, y)$
- When we say $\exists$ a predicate then it is higher order logic

Examples

1. Not every customer have purchased milk and bread

First Order Logic (Predicate Logic)

- We have **constants**, **variables**, **predicates** and **functions**
- Here $P(x)$ could means $\forall x$ we have $P(x)$ or $\exists x$ such that $P(x)$
- Variable x has a domain from where it gets values
- $\forall x, \exists y P(x, y)$ is not always same as $\exists y, \forall x P(x, y)$
- When we say $\exists$ a predicate then it is higher order logic

Examples

1. Not every customer have purchased milk and bread

$$\exists c \;\; Cust(c) \wedge [\neg shop(milk, c) \vee \neg shop(bread, c)]$$

2. Only one customer have purchased guitar

First Order Logic (Predicate Logic)

- We have **constants**, **variables**, **predicates** and **functions**
- Here $P(x)$ could means $\forall x$ we have $P(x)$ or $\exists x$ such that $P(x)$
- Variable x has a domain from where it gets values
- $\forall x, \exists y P(x, y)$ is not always same as $\exists y, \forall x P(x, y)$
- When we say $\exists$ a predicate then it is higher order logic

Examples

1. Not every customer have purchased milk and bread

$$\exists c \;\; Cust(c) \wedge [\neg shop(milk, c) \vee \neg shop(bread, c)]$$

2. Only one customer have purchased guitar

$$\exists x \;\; [Cust(x) \wedge shop(G, x) \wedge \forall y [\neg(x = y) \wedge Cust(y) \Rightarrow \neg shop(G, y)]]$$

3. Only one customer have purchased guitar and pen
4. Highest purchase in forenoon is more than afternoon.

Inference in First Order Logic

- **Universal Elimination** $\forall x \; Feels(x, king)$ could be $Feels(Raju, king)$ substitution $\{x/\text{Raju}\}$ is done using some ground term.
- **Existential Elimination** $\exists x \; Feels(x, king)$ could be $Feels(man, king)$ if *man* does not appear in knowledge base [5]
- **Existential Introduction** If $Feels(Raju, king)$ then we can say $\exists x \; Feels(x, king)$

1. It is crime for Magadh to sell formula to a hostile country
2. Country Bhind, an enemy of Magadh have purchased some formula from Dara
3. Dara is from Magadh
4. Question: Is Dara a criminal?

[5] *man* is a name of person who feels like king

Prolog

- A logic programming language [6]
- Compile as ['a.pl'].
- If :- and , or ; not not
- write('hello'), nl

```
warm_blood(penguin).
warm_blood(human).
produce_milk(penguin).
produce_milk(human).
have_feather(penguin).
have_hair(human).
mammal(X) :-
warm_blood(X),
produce_milk(X),

have_hair(X).
```

```
?- mammal(penguin)
no
?- mammal(X).
X = human.
```

- ```
  is_even(X) :-
  Y is X//2, X =:= 2*Y.
  ```
- `write('what is your name/ '), read(X), write('Hi '),write(X).`

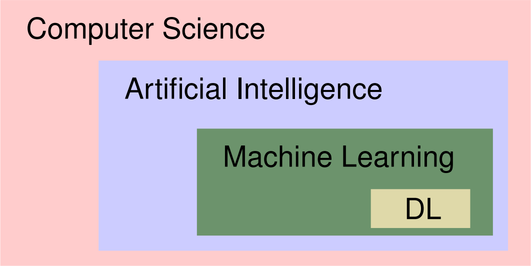
Many more things are possible

[6] http://www.swi-prolog.org/
  ```

Machine Learning

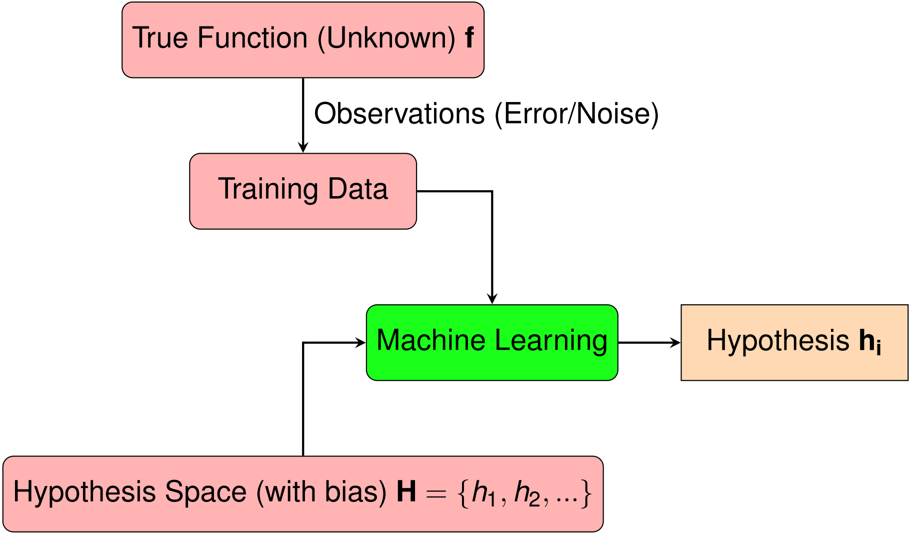
For some problems we don't precisely know either 1) how to solve, or 2) difficult to specify solution procedure

Then we go for **Machine Learning** (ML)



http://ktiwari.in/ml

The Flow of ML



Probability of observing a dataset

Assume you are flipping a biased coin where $p(H) = 0.4$. What is the probability that you see this dataset $D =< H, H, T, T, H, H >$

- $p(H) = 0.4$
- $p(T) = 1 - p(H) = 1 - 0.4 = 0.6$

Probability of observing a dataset

Assume you are flipping a biased coin where $p(H) = 0.4$. What is the probability that you see this dataset $D =< H, H, T, T, H, H >$

- $p(H) = 0.4$
- $p(T) = 1 - p(H) = 1 - 0.4 = 0.6$
- If **all the trails are independent** then $p(D|\theta)$

$$= p(H) \times p(H) \times p(T) \times p(T) \times p(H) \times p(H)$$

$$= 0.4^4 \times 0.6^2 = 0.009216$$

Probability of observing a dataset

Assume you are flipping a biased coin where $p(H) = 0.4$. What is the probability that you see this dataset $D =< H, H, T, T, H, H >$

- $p(H) = 0.4$
- $p(T) = 1 - p(H) = 1 - 0.4 = 0.6$
- If **all the trails are independent** then $p(D|\theta)$

$$= p(H) \times p(H) \times p(T) \times p(T) \times p(H) \times p(H)$$

$$= 0.4^4 \times 0.6^2 = 0.009216$$

Note: Order of elements in the data set do not matter in the trial. So $p(< H, H, H, H, T, T >)$ is same (in fact any other permutation)

What is θ

It is the parameter. For our case it represents $p(H) = 0.4$

Hypothesis

X	Y
10	0
11	0
12	0
13	1
14	0
15	1
16	0
17	1
18	1

h_1	h_2	...
0	1	...
0	0	...
0	1	...
1	0	...
1	1	...
1	0	...
1	1	...
1	0	...
1	1	...

- In this example h_1, h_2, ... are hypothesis.
- **Hypothesis** is a function that aims to provide value of the Y
- Can you identify h_1 and h_2
- Represent H as candidate set of hypothesis, $i.e. h_i \in H$
- Size of H is at least 2^m

Bayesian Learning

It is based on assumption that quantities of interest are governed by probability distribution

- Notation
  - $P(h)$: initial probability that hypothesis h holds
  - $P(D)$: probability that data D will be observed
  - $P(D|h)$: probability of observing data D given some world in which hypothesis h holds
  - $P(h|D)$: probability of holding hypothesis h when data D is observed

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

Maximum a posteriori (MAP)

- Choose a hypothesis that maximizes $P(h|D)$

$$h_{MAP} = \operatorname*{argmax}_{h \in H} P(h|D)$$

Maximum a posteriori (MAP)

- Choose a hypothesis that maximizes $P(h|D)$

$$
\begin{aligned}
h_{MAP} &= \underset{h \in H}{\operatorname{argmax}}\, P(h|D) \\
&= \underset{h \in H}{\operatorname{argmax}}\, \frac{P(D|h)P(h)}{P(D)}
\end{aligned}
$$

Maximum a posteriori (MAP)

- Choose a hypothesis that maximizes $P(h|D)$

$$
\begin{aligned}
h_{MAP} &= \underset{h \in H}{\operatorname{argmax}} P(h|D) \\
&= \underset{h \in H}{\operatorname{argmax}} \frac{P(D|h)P(h)}{P(D)} \\
&= \underset{h \in H}{\operatorname{argmax}} P(D|h)P(h) \qquad (2)
\end{aligned}
$$

Maximum a posteriori (MAP)

- Choose a hypothesis that maximizes $P(h|D)$

$$
\begin{aligned}
h_{MAP} &= \underset{h \in H}{\operatorname{argmax}}\, P(h|D) \\
&= \underset{h \in H}{\operatorname{argmax}}\, \frac{P(D|h)P(h)}{P(D)} \\
&= \underset{h \in H}{\operatorname{argmax}}\, P(D|h)P(h)
\end{aligned} \tag{2}
$$

- Because $P(D)$ is independent of h

Maximum a posteriori (MAP)

- Choose a hypothesis that maximizes $P(h|D)$

$$\begin{aligned} h_{MAP} &= \operatorname*{argmax}_{h \in H} P(h|D) \\ &= \operatorname*{argmax}_{h \in H} \frac{P(D|h)P(h)}{P(D)} \\ &= \operatorname*{argmax}_{h \in H} P(D|h)P(h) \end{aligned} \tag{2}$$

- Because $P(D)$ is independent of h
- If all the hypothesis are equally probable, we may further simplify called *maximum likelihood (ML)*

$$h_{ML} = \operatorname*{argmax}_{h \in H} P(D|h) \tag{3}$$

For our current example

X	Y
10	0
11	0
12	0
13	1
14	0
15	1
16	0
17	1
18	1

h_1	h_2	...
0	1	...
0	0	...
0	1	...
1	1	...
1	1	...
1	0	...
1	1	...
1	0	...
1	1	...

For our current example

X	Y
10	0
11	0
12	0
13	1
14	0
15	1
16	0
17	1
18	1

h_1	h_2	...
0	1	...
0	0	...
0	1	...
1	1	...
1	1	...
1	0	...
1	1	...
1	0	...
1	1	...

- Let bias for h_1 and h_2 be 2/50 and 6/50
- Since h_1 and h_2 are correct with probability 7/9 and 3/9 respectively
- Posterior is (7/9)*(2/50) and (3/9)*(6/50)
- Normalized probabilities are 0.4375 and 0.5625 respectively
- So MAP hypothesis corresponds to?

For our current example

X	Y
10	0
11	0
12	0
13	1
14	0
15	1
16	0
17	1
18	1

h_1	h_2	...
0	1	...
0	0	...
0	1	...
1	1	...
1	1	...
1	0	...
1	1	...
1	0	...
1	1	...

- Let bias for h_1 and h_2 be 2/50 and 6/50
- Since h_1 and h_2 are correct with probability 7/9 and 3/9 respectively
- Posterior is (7/9)*(2/50) and (3/9)*(6/50)
- Normalized probabilities are 0.4375 and 0.5625 respectively
- So MAP hypothesis corresponds to? h_2
- Can ML hypothesis?

For our current example

X	Y
10	0
11	0
12	0
13	1
14	0
15	1
16	0
17	1
18	1

h_1	h_2	...
0	1	...
0	0	...
0	1	...
1	1	...
1	1	...
1	0	...
1	1	...
1	0	...
1	1	...

- Let bias for h_1 and h_2 be 2/50 and 6/50
- Since h_1 and h_2 are correct with probability 7/9 and 3/9 respectively
- Posterior is (7/9)*(2/50) and (3/9)*(6/50)
- Normalized probabilities are 0.4375 and 0.5625 respectively
- So MAP hypothesis corresponds to? h_2
- Can ML hypothesis? it is h_1

For our current example

X	Y
10	0
11	0
12	0
13	1
14	0
15	1
16	0
17	1
18	1

h_1	h_2	...
0	1	...
0	0	...
0	1	...
1	1	...
1	1	...
1	0	...
1	1	...
1	0	...
1	1	...

- Let bias for h_1 and h_2 be 2/50 and 6/50
- Since h_1 and h_2 are correct with probability 7/9 and 3/9 respectively
- Posterior is (7/9)*(2/50) and (3/9)*(6/50)
- Normalized probabilities are 0.4375 and 0.5625 respectively
- So MAP hypothesis corresponds to? h_2
- Can ML hypothesis? it is h_1

- **Brute-force MAP learning algorithm:** Evaluates posterior probability for all and returns the one with maximum
- **Consistent Learner:** learning algorithm is consistent learner if it provides a hypothesis that commits zero error

Naive Bayes Classifier

Bayes classifier is a highly practical Bayesian learning method

- In some domains, its performance found to be comparable to neural network and decision tree
- The Bayesian approach to classify a new instance is to assign the most probable target value describing the instance
  $v_{MAP} = \text{argmax}_{v_j \in V} P(v_j | a_1, a_2, ..., a_n)$
- We can use Bayes theorem to rewrite this expression as

$$\begin{aligned} v_{MAP} &= \underset{v_j \in V}{\text{argmax}} \frac{P(a_1, a_2, ..., a_n | v_j) P(v_j)}{P(a_1, a_2, ..., a_n)} \\ &= \underset{v_j \in V}{\text{argmax}} P(a_1, a_2, ..., a_n | v_j) P(v_j) \end{aligned} \tag{4}$$

Naive Bayes has assumption is that the attribute values are conditionally independent given the target value

Naive Bayes Classifier

If attribute values are conditionally independent given the target value

- Under this assumption,
- Given a target value, the probability of observing the conjunction $< a_1, a_2, ..., a_n >$ is just the product of the probabilities.

$$P(a_1, a_2, ..., a_n | v_j) = \Pi_i P(a_i | v_j)$$

Naive Bayes classifier

is the one which

$$\underset{v_j \in V}{\text{argmax}} \, P(v_j) \Pi_i P(a_i | v_j)$$

Example: Naive Bayes Classification

Given the data

Day	Outlook	Temperature	Humidity	Wind	Play
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rainy	Mild	High	Weak	Yes
D5	Rainy	Cool	Normal	Weak	Yes
D6	Rainy	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rainy	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rainy	Mild	High	Strong	No

Determine classification for $< \mathit{Rainy}, \mathit{Hot}, \mathit{High}, \mathit{Strong} >$

Example: Naive Bayes Classification

Day	Outlook	Temperature	Humidity	Wind	Play
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D6	Rainy	Cool	Normal	Strong	No
D8	Sunny	Mild	High	Weak	No
D14	Rainy	Mild	High	Strong	No

Day	Outlook	Temperature	Humidity	Wind	Play
D3	Overcast	Hot	High	Weak	Yes
D4	Rainy	Mild	High	Weak	Yes
D5	Rainy	Cool	Normal	Weak	Yes
D7	Overcast	Cool	Normal	Strong	Yes
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rainy	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes

P(Yes) = 9/14

P(No) = 5/14

Example: Naive Bayes Classification

Day	Outlook	Temperature	Humidity	Wind	Play
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D6	Rainy	Cool	Normal	Strong	No
D8	Sunny	Mild	High	Weak	No
D14	Rainy	Mild	High	Strong	No

Day	Outlook	Temperature	Humidity	Wind	Play
D3	Overcast	Hot	High	Weak	Yes
D4	Rainy	Mild	High	Weak	Yes
D5	Rainy	Cool	Normal	Weak	Yes
D7	Overcast	Cool	Normal	Strong	Yes
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rainy	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes

Outlook

	Yes	No
Sunny	2/9	

P(Yes) = 9/14

P(No) = 5/14

Example: Naive Bayes Classification

Day	Outlook	Temperature	Humidity	Wind	Play
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D6	Rainy	Cool	Normal	Strong	No
D8	Sunny	Mild	High	Weak	No
D14	Rainy	Mild	High	Strong	No

Day	Outlook	Temperature	Humidity	Wind	Play
D3	Overcast	Hot	High	Weak	Yes
D4	Rainy	Mild	High	Weak	Yes
D5	Rainy	Cool	Normal	Weak	Yes
D7	Overcast	Cool	Normal	Strong	Yes
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rainy	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes

Outlook

P(Yes) = 9/14

P(No) = 5/14

	Yes	No
Sunny	2/9	3/5
Overcast	4/9	0/5
Rainy	3/9	2/5

Example: Naive Bayes Classification

- P(Yes) = 9/14 P(No) = 5/14

Outlook

	Yes	No
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Humidity

	Yes	No
High	3/9	4/5
Low	6/9	1/5

Wind

	Yes	No
Strong	3/9	3/5
Weak	6/9	2/5

Temperature

	Yes	No
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Example: Naive Bayes Classification

For $x =< Rainy, Hot, High, Strong >$

P(Yes)

- $P(x|Yes) \times P(Yes)$
- $P(Rainy|Yes) \times P(Hot|Yes) \times P(High|Yes) \times P(Strong|Yes) \times P(Yes)$
- $3/9 \times 2/9 \times 3/9 \times 3/9 \times 9/14$
- 0.005291...

Example: Naive Bayes Classification

For $x = < Rainy, Hot, High, Strong >$

P(Yes)

- $P(x|Yes) \times P(Yes)$
- $P(Rainy|Yes) \times P(Hot|Yes) \times P(High|Yes) \times P(Strong|Yes) \times P(Yes)$
- $3/9 \times 2/9 \times 3/9 \times 3/9 \times 9/14$
- 0.005291...

P(No)

- $P(x|No) \times P(No)$
- $P(Rainy|No) \times P(Hot|No) \times P(High|No) \times P(Strong|No) \times P(No)$
- $2/5 \times 2/5 \times 4/5 \times 3/5 \times 5/14$
- 0.027428...

So the classification of x is **No**

Probability

- $P(x, y) = P(x) \times P(y|x)$
- **Independence** of x and y implies $P(y|x) = P(y)$
- Then $P(x, y) = P(x) \times P(y)$
- Bayes Rule

$$P(x|y) = \frac{P(x, y)}{P(y)} = \frac{P(y|x) \times P(x)}{P(y)}$$

- **Marginal:** distribution of a single variable x can be obtained from a given joint distribution $p(x, y)$ by

$$p(x) = \sum_{y} p(x, y)$$

- The process of computing a marginal from a joint distribution is called marginalisation.

$$p(x_1, ..., x_{i-1}, x_{i+1}, ..., x_n) = \sum_{x_i} p(x_1, x_2, ..., x_n)$$

Modeling Independencies

One morning Tracey leaves her house and realities that her grass is wet. Is it due to overnight rain or did she forget to turn off the sprinkler last night? Next she notices that the grass of her neighbor, Jack, is also wet.

Modeling Independencies

One morning Tracey leaves her house and realities that her grass is wet. Is it due to overnight rain or did she forget to turn off the sprinkler last night? Next she notices that the grass of her neighbor, Jack, is also wet.

(R=1) $\rightarrow$ rain last night,
(S=1) $\rightarrow$ sprinkler on last night,
(J=1) $\rightarrow$ Jack's grass is wet,
(T=1) $\rightarrow$ Traceya's Grass is wet

- Model of Traceya's world involves probability distribution on T, J, R, S that has $2^4 = 16$ states

Conditional Independence

- We may assume that Traceya's grass is wet depends only directly on whether or not it has been raining and whether or not her sprinkler was on so $p(T|J,R,S) = p(T|R,S)$
- Assume that Jack's grass is wet is influenced only directly by whether or not it has been raining $p(J|R,S) = p(J|R)$
- Furthermore, we assume the rain is not directly influenced by the sprinkler $p(R|S) = p(R)$
- Therefore, our model becomes

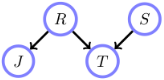
$$\begin{aligned} p(T,J,R,S) &= p(T|J,R,S)p(J|R,S)p(R|S)p(S) \\ &= p(T|R,S)p(J|R)p(R)p(S) \end{aligned}$$

Conditional Independence

- We may assume that Traceya's grass is wet depends only directly on whether or not it has been raining and whether or not her sprinkler was on so $p(T|J,R,S) = p(T|R,S)$
- Assume that Jack's grass is wet is influenced only directly by whether or not it has been raining $p(J|R,S) = p(J|R)$
- Furthermore, we assume the rain is not directly influenced by the sprinkler $p(R|S) = p(R)$
- Therefore, our model becomes

$$\begin{aligned} p(T,J,R,S) &= p(T|J,R,S)p(J|R,S)p(R|S)p(S) \\ &= p(T|R,S)p(J|R)p(R)p(S) \end{aligned}$$

- Number of values we need to specify is 4+2+1+1=8
- We can represent these conditional independencies as



Belief network

How to represent these conditional independencies?

- **Belief network** is a distribution of the form

$$p(x_1, x_2, ..., x_n) = \Pi_{i=1}^{n} p(x_i | pa(x_i))$$

where $pa(x_i)$ represent the parental variables of variable x_i

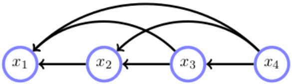
Belief network

How to represent these conditional independencies?

- **Belief network** is a distribution of the form

$$p(x_1, x_2, ..., x_n) = \Pi_{i=1}^{n} p(x_i | pa(x_i))$$

  where $pa(x_i)$ represent the parental variables of variable x_i
- Represented as a directed graph, with an arrow pointing from a parent variable to child variable, a belief network corresponds to a Directed Acyclic Graph (DAG)



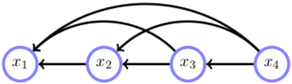
Belief network

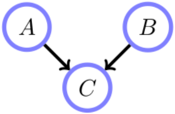
How to represent these conditional independencies?

- **Belief network** is a distribution of the form

$$p(x_1, x_2, ..., x_n) = \Pi_{i=1}^{n} p(x_i | pa(x_i))$$

  where $pa(x_i)$ represent the parental variables of variable x_i
- Represented as a directed graph, with an arrow pointing from a parent variable to child variable, a belief network corresponds to a Directed Acyclic Graph (DAG)





$$p(A, B, C) = p(C|A, B)p(A)p(B)$$

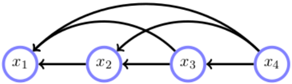
Belief network

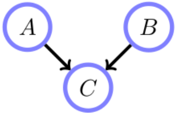
How to represent these conditional independencies?

- **Belief network** is a distribution of the form

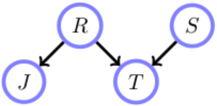
$$p(x_1, x_2, ..., x_n) = \Pi_{i=1}^{n} p(x_i | pa(x_i))$$

  where $pa(x_i)$ represent the parental variables of variable x_i
- Represented as a directed graph, with an arrow pointing from a parent variable to child variable, a belief network corresponds to a Directed Acyclic Graph (DAG)





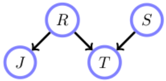
$$p(A, B, C) = p(C|A, B)p(A)p(B)$$



Example

One morning Tracey realises that her grass is wet and the grass of her neighbour, Jack, is also wet. Let the prior probabilities be $p(R{=}1) = 0.2$ and $p(S{=}1) = 0.1$. We set $p(J{=}1|R{=}1) = 1$, $p(J{=}1|R{=}0) = 0.2$, $p(T{=}1|R{=}1, S{=}0) = 1$, $p(T{=}1|R{=}1, S{=}1) = 1$, $p(T{=}1|R{=}0, S{=}1) = 0.9$, $p(T{=}1|R{=}0, S{=}0) = 0$

Using following Beleif Network; calculate

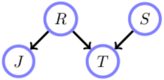


1. Probability that the sprinkler was *on* overnight, given that Traceya's grass is wet.

Example

One morning Tracey realises that her grass is wet and the grass of her neighbour, Jack, is also wet. Let the prior probabilities be $p(R{=}1) = 0.2$ and $p(S{=}1) = 0.1$. We set $p(J{=}1|R{=}1) = 1$, $p(J{=}1|R{=}0) = 0.2$, $p(T{=}1|R{=}1, S{=}0) = 1$, $p(T{=}1|R{=}1, S{=}1) = 1$, $p(T{=}1|R{=}0, S{=}1) = 0.9$, $p(T{=}1|R{=}0, S{=}0) = 0$

Using following Beleif Network; calculate

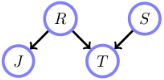


1. Probability that the sprinkler was *on* overnight, given that Traceya's grass is wet. $p(S{=}1|T{=}1) = 0.3382$

Example

One morning Tracey realises that her grass is wet and the grass of her neighbour, Jack, is also wet. Let the prior probabilities be $p(R{=}1) = 0.2$ and $p(S{=}1) = 0.1$. We set $p(J{=}1|R{=}1) = 1$, $p(J{=}1|R{=}0) = 0.2$, $p(T{=}1|R{=}1, S{=}0) = 1$, $p(T{=}1|R{=}1, S{=}1) = 1$, $p(T{=}1|R{=}0, S{=}1) = 0.9$, $p(T{=}1|R{=}0, S{=}0) = 0$

Using following Beleif Network; calculate

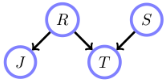


1. Probability that the sprinkler was *on* overnight, given that Traceya's grass is wet. $p(S{=}1|T{=}1) = 0.3382$ How? on next slide
2. Probability that Traceya's sprinkler was *on* overnight, given that her grass is wet and that Jack's grass is also wet.

Example

One morning Tracey realises that her grass is wet and the grass of her neighbour, Jack, is also wet. Let the prior probabilities be $p(R{=}1) = 0.2$ and $p(S{=}1) = 0.1$. We set $p(J{=}1|R{=}1) = 1$, $p(J{=}1|R{=}0) = 0.2$, $p(T{=}1|R{=}1, S{=}0) = 1$, $p(T{=}1|R{=}1, S{=}1) = 1$, $p(T{=}1|R{=}0, S{=}1) = 0.9$, $p(T{=}1|R{=}0, S{=}0) = 0$

Using following Beleif Network; calculate



1. Probability that the sprinkler was *on* overnight, given that Traceya's grass is wet. $p(S{=}1|T{=}1) = 0.3382$ How? on next slide
2. Probability that Traceya's sprinkler was *on* overnight, given that her grass is wet and that Jack's grass is also wet. $p(S{=}1|T{=}1, J{=}1) = 0.1604$

Example: Probability of $p(S{=}1|T{=}1)$

$$
\begin{aligned}
p(S=1|T=1) &= \frac{p(S=1,T=1)}{p(T=1)} \\
&= \frac{\sum_{J,R} p(S=1,J,R,T=1)}{\sum_{J,R,S} p(T=1,J,R,S)} \qquad (5) \\
&= \frac{\sum_{J,R} p(J|R)p(T=1|R,S=1)p(R)p(S=1)}{\sum_{J,R,S} p(J|R)p(T=1|R,S)p(R)p(S)} \\
&= \frac{\sum_{J,R} p(T=1|R,S=1)p(R)p(S=1)}{\sum_{J,R,S} p(T=1|R,S)p(R)p(S)} \qquad (6) \\
&= \frac{0.9\times 0.8\times 0.1 + 1\times 0.2\times 0.1}{.9\times .8\times .1 + 1\times .2\times .1 + 0\times .8\times .9 + 1\times .2\times .9} \\
&= \boxed{0.3382}
\end{aligned}
$$

Uses given belief network in (5) and proof in (6)

Thank You!

Thank you very much for your attention!

Queries ?

(Reference[7])

[7] 1) Book - *AIMA*, ch-07, Russell and Norvig. 2) Book - *Logic in CS*, ch-01, Mitchel Huth and Mark Ryan. 2) Book - *Bayeian Reasoning and Machine Learning*, ch-04, David Barber.