IS/SE/SS ZC444: Artificial Intelligence

13 Logic, and ML

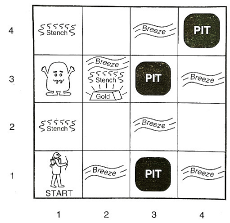
Dr. Kamlesh Tiwari
Associate Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

Nov 04, 2023 ONLINE WILP @ BITS-Pilani [July-Dec 2023]

http://ktiwari.in/ai

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	true
false	true	false	false	false	true	false	true	true	true	true	true	true
false	true	false	false	false	true	true	true	true	true	true	true	true
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

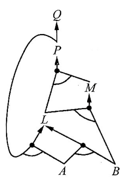
$$((A \wedge B) \rightarrow C) \leftrightarrow (A \rightarrow (B \rightarrow C))$$

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 $L_{1,1} \wedge breeze \rightarrow B_{1,1}$ if we know $L_{1,1}$ and *breeze* then $B_{1,1}$ is added in knowledge base [1]

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



- Applies Modus Ponens
$$\frac{\phi \quad \phi \rightarrow \psi}{\psi}$$
- An and-or tree gets constructed

[1] $L_{1,1}$: 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

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

2. Only one customer have purchased guitar

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

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** $\forall x \quad Feels(x, king)$ could be $Feels(Raju, king)$ substitution $\{x/\text{Raju}\}$ is done using some ground term.
- **Existential Elimination** $\exists x \quad Feels(x, king)$ could be $Feels(man, king)$ if man does not appear in knowledge base [2]
- **Existential Introduction** If $Feels(Raju, king)$ then we can say $\exists x \quad Feels(x, king)$

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?

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

Prolog

- A logic programming language [3]
- 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).
  ```

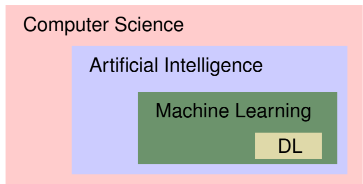
Many more things are possible

[3] 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

Machine Learning: Tasks

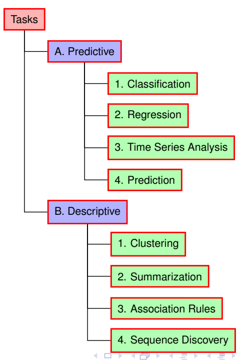
Two broad categories of machine learning models are *Predictive* and *Descriptive*. Some of the related tasks are

Predictive

focuses towards new data item or expanding beyond known facts

Descriptive

focus to understand the available data



Types of Learning

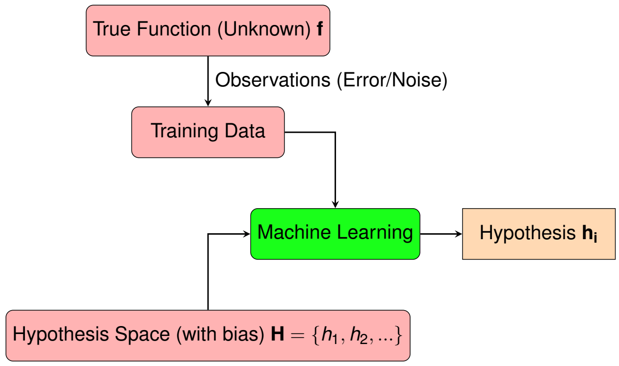
- **Supervised:** "right answers" are provided for sufficient training examples. Computer tells "right answers" for new input. Performance measure. (Classification and regression)
- **Unsupervised:** "right answers" are NOT provided and the computer tries to make sense of the data. How good the spread of items is. (clustering and association rule)
- **Semi-supervised:** "right answers" are provided for few training examples only
- **Active:** computer can ask questions. Needs less training. Opposite is passive learning
- **Lazy:** learner do not consolidate the findings.
- **Reinforced:** hit and trial method to minimize cost. (game playing)
- **Transfer:** Learning a task B to do A. (cycle riding for bike riding)
- **Deep:** processing like human brain

Applications of ML

In many domains including finance, robotics, bioinformatics, vision, natural language, *etc.*

- Spam filtering
- Speech/handwriting recognition
- Object detection/recognition
- Weather prediction
- Stock market analysis
- Search engines (e.g, Google)
- Ad placement on websites
- Adaptive website design
- Credit-card fraud detection
- Webpage clustering (e.g.,Google News)
- Machine Translation (e.g., Google Translate)
- Recommendation systems (e.g., Netflix, Amazon)
- Classifying DNA sequences
- Automatic vehicle navigation
- Performance tuning of computer systems
- Predicting good compilation flags for programs
- .. and many more

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 $D =< H, H, T, T, H, H >$

- $p(H) = 0.4$
- $p(T) = 1 - p(H) = 1 - 0.4 = 0.6$
- 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 $P(h|D)$

$$\begin{aligned} h_{MAP} &= \underset{h \in H}{\operatorname{argmax}} P(h|D) \\ &= \underset{h \in H}{\operatorname{argmax}} \frac{P(D|h)P(h)}{P(D)} \\ &= \underset{h \in H}{\operatorname{argmax}} P(D|h)P(h) \qquad (1) \end{aligned}$$

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

$$h_{ML} = \underset{h \in H}{\operatorname{argmax}} P(D|h) \qquad (2)$$

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

Bayes Optimal Classifier

Switching the question, from "which is most probable hypothesis?" to "what is the most probable classification of the new instance?"

Is it possible to do better then MAP?

Example: Let posterior probabilities of three hypotheses h_1, h_2, h_3 given the training data are 0.4, 0.3, and 0.3 (obviously h_1 is MAP)

- Let classification of a new instance x is positive by h_1 and negative by h_2 and h_3
- By taking all hypotheses into account, the probability that x is positive is 0.4, and negative is 0.6
  - *Most probable classification is negative and it differs from MAP*

Bayes optimal classification:

$$\underset{v_j \in V}{\text{argmax}} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D)$$

where classification v_j is from V and $P(v_j|D)$ is the correct classification

Bayes Optimal Classifier

$$\boxed{\text{argmax}_{v_j \in V} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D)}$$

$V = \{\oplus, \ominus\}$

$P(h_1\|D)$=0.4	$P(\ominus\|h_1)$=0	$P(\oplus\|h_1)$=1
$P(h_2\|D)$=0.3	$P(\ominus\|h_2)$=1	$P(\oplus\|h_2)$=0
$P(h_3\|D)$=0.3	$P(\ominus\|h_3)$=1	$P(\oplus\|h_3)$=0

Therefore,

$$\sum_{h_i \in H} P(\oplus|h_i)P(h_i|D) = 0.4 \qquad \sum_{h_i \in H} P(\ominus|h_i)P(h_i|D) = 0.6$$

and

$$\text{argmax}_{v_j \in \{\oplus,\ominus\}} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D) = \ominus$$

This type of classifier is called a Bayes optimal classifier, or Bayes optimal learner.

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

$$\begin{aligned} v_{MAP} &= \underset{v_j \in V}{\text{argmax}} \frac{P(a_1, a_2, ..., a_n|v_j)P(v_j)}{P(a_1, a_2, ..., a_n)} \\ &= \underset{v_j \in V}{\text{argmax}} P(a_1, a_2, ..., a_n|v_j)P(v_j) \end{aligned} \quad (3)$$

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 $< Rainy, Hot, High, Strong >$

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

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...

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**

Thank You!

Thank you very much for your attention!

Queries ?

(Reference[4])

[4] 1) Book - *AIMA*, ch-07/08, Russell and Norvig. 2) Book - *Logic in CS*, ch-01/02, Mitchel Huth and Mark Ryan.