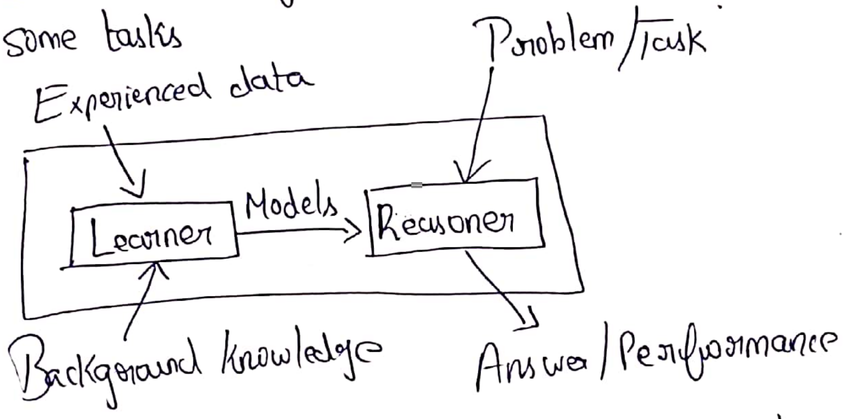
Machine Learning (ML)

Learning: It is the ability to Improve once behaviour based on experience.

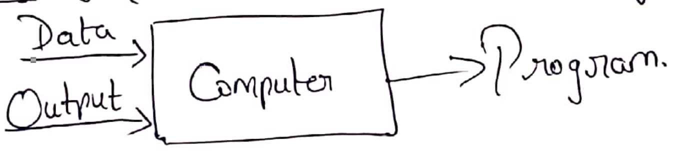
Machine Learning: Explores algorithms that can,

* Learn from data / build a model from data.
* Use the model for prediction, decision making or solving some tasks.

Learner:



Machine Learning: (System adapts to changing environment and brings useful information)



Definition Of ML:

→ A computer program is said to learn from experience E w.r.t some task T and some performance measure P, if its performance on T, as measured by P improves with experience E — Tom Michell.

Ex To predict the traffic pattern

Traffic pattern - Task T

Past Traffic pattern - Experience E

If the machine has successfully learned then it will do better at predicting future traffic pattern (Performance P)

Why ML? ⇒ Recent Progress in algorithms and theorems

→ Growing flood of online data.

→ Computational power is available.

→ Budding industries

Applications Of ML:

-> Social Media (Behaviour Study)
-> Medical Diagnostics (Based of on Historical medical regards detects wit which treatment is better for the patients)
-> Super Market (Business Intelligents)
-> Aircraft Design (Auto learning)
-> Robot Design
-> Finance (Predict if a stock will rise or fall)
-> Self Driving Cars (Auto Parking Cars)
-> Natural Language Processing - NLP (Movie Recomadations)
-> Speech ~~Reg~~ Recognition.
-> Computer Machine (Face detection / ~~Object~~ Recognition)
-> Fraud Detection
-> Identification of Spam Mail
-> Rainfall Prediction
-> Google Translator.

Examples Of Machine Learning Applications:

-> The mining task are divided into two
1) Descriptive - characterise the properties of data.
2) Predictive - Inference on current data to make predictions.

Functionalities:

1) Class or concept description < Characterization (Summerization) / Discrimination (Comparison)

↳ Compare target class with contrasting class

2) Association analysis
3) Classification and prediction
4) Clustering (analysis)
5) Outlier Analysis.
6) Evolution Analysis.

Association Analysis:

-> Find the frequent items/elements
-> Find the association between the frequent elements
* Single dimensional association rule.

~~Here~~
Ex. buys ('x', computer) => buys ('x', software)

* Multidimensional Association rule

Eg. age ('X', 40-60) ∧ income ('X', 1 lakh - 10 lakh) ⟹ buys ('X', computer)

Here, 3 dimensions are used namely age, income and buys.

ALGORITHMS USED:

1) Apriori Algorithm
2) FP-Tree Algorithm

APRIORI ALGORITHM:

Ex:

T_{ID}	Items
T_1	I_1, I_2, I_5
T_2	I_2, I_4
T_3	I_2, I_3
T_4	I_1, I_2, I_4
T_5	I_1, I_3
T_6	I_2, I_3
T_7	I_1, I_3
T_8	I_1, I_2, I_3, I_5
T_9	I_1, I_2, I_3

Given: The Transaction database 'D'

Minimum support count = 2

Minimum confidence Threshold = 70%

Scan D for count of each candidate →

IS	SC
I_1	6
I_2	7
I_3	6
I_4	2
I_5	2

compare all csc with msc →

L_1

IS	SC
I_1	6
I_2	7
I_3	6
I_4	2
I_5	2

→

o lakh) ⟹
('x', computer)

Generate C_2 candidates from L_1 →

C_k from L_{k-1}
C_2 from L_1

C_2

IS	SC
$I_1 I_2$	4
$I_1 I_3$	4
$I_1 I_4$	1
$I_1 I_5$	2
$I_2 I_3$	4
$I_2 I_4$	2
$I_2 I_5$	2
$I_3 I_4$	0
$I_3 I_5$	1
$I_4 I_5$	0

Compare all csc with msc →

L_2

IS	SC
$I_1 I_2$	4
$I_1 I_3$	4
$I_1 I_5$	2
$I_2 I_3$	4
$I_2 I_4$	2
$I_2 I_5$	2

→

Generate C_3 candidates from L_2 →

C_3

IS	SC
$I_1 I_2 I_3$	2
$I_1 I_2 I_5$	2
$I_1 I_2 I_4$	1
$I_1 I_3 I_5$	1
~~$I_1 I_4$~~	
$I_1 I_2 I_3 I_4$	0
$I_1 I_2 I_3 I_5$	1
$I_1 I_2 I_4 I_5$	0
$I_2 I_3 I_4$	0
$I_2 I_3 I_5$	1
$I_2 I_4 I_5$	0

Compare al csc with msc →

L_3

IS	SC
$I_1 I_2 I_3$	2
$I_1 I_2 I_5$	2

Generate C_4 candidates from L_3 →

C_4

IS	SC
$I_1 I_2 I_3 I_5$	1

∴ The resultant frequent item set are sets

IS	Sc
$I_1 I_2 I_3$	2
$I_1 I_2 I_5$	2

→

Finding Association Rules:

$I_1 \ I_2 \ I_3$

A	⇒	B
I_1	⇒	$I_2 \wedge I_3$
I_2	⇒	$I_1 \wedge I_3$
I_3	⇒	$I_1 \wedge I_2$
$I_1 \wedge I_2$	⇒	I_3
$I_1 \wedge I_3$	⇒	I_2
$I_1 \wedge I_3$	⇒	I_1

Confidence

$$A \rightarrow B \Rightarrow \frac{P(A \cup B)}{P(A)}$$

$$\Rightarrow \frac{S \ c(A \cup B)}{SC(A)}$$

$$\frac{SC(I_1 I_2 I_3)}{SC(I_1)} = \frac{2}{6} = \frac{1}{3} = 30\%.$$

$$\frac{SC(I_1 I_2 I_3)}{SC(I_2)} = \frac{2}{7} = 0.28 \Rightarrow 28\%.$$

$$\Rightarrow \frac{2}{6} = \frac{1}{3} = 30\%.$$

$$\Rightarrow 2/4 = 1/2 = 0.5 = 50\%.$$

$$\Rightarrow 2/4 = 1/2 = 50\%.$$

$$\Rightarrow 2/4 = 1/2 = 50\%.$$

$I_1 \ I_2 \ I_5$

2

I_1	$I_2 \wedge I_5$	$2/6 = 0.3 = 30\%$.
I_2	$I_1 \wedge I_5$	$2/7 = 0.28$
I_5	$I_1 \wedge I_2$	$2/2 = 100\%$.
$I_1 \wedge I_2$	I_5	$2/4 = 1/2 = 0.5$
$I_1 \wedge I_5$	I_2	$2/2 = 100\%$.
$I_2 \wedge I_5$	I_1	$2/2 = 100\%$.

The

Note

CLAS

→

AL

1) D

2)

3)

P

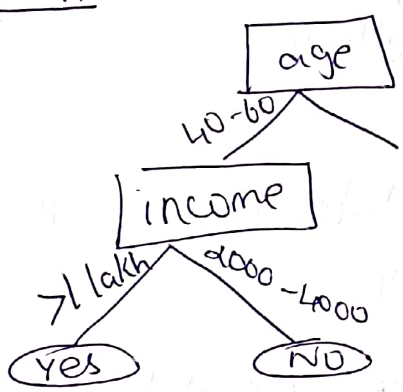
=

∴ The resultant association rules are

I_1 $\Rightarrow$	$I_1 \wedge I_2$
$I_1 \wedge I_5$ $\Rightarrow$	I_2
$I_2 \wedge I_5$	I_1

<u>Note</u>: 2 Algorithms are used for association rules mining
1) Aprion algorithm.
2) FP-tree (Frequent pattern growth) tree algorithm

CLASSIFICATION:



→ If age equals to 40-60 and income = greater than 10 Lakh then buys computer = Yes

ALGORITHMS:

1) Desicision Tree induction:
2) Naive Bayesian Classificaton
3) Neural Networks.

PREDICTION:

⇒ Predicting the future trends based on past application

Algorithms:

1) Linear Regression
2) Multiple Regression

CLUSTER ANALYSIS:

-> Similar elements are grouped together to form cluster SU
Eg super computer

Algorithms:

1) K-Means
2) Agglomerative Nesting (AGNES)
3) Divisive Analysis (DIANA)

OUTLIER ANALYSIS:

-> The element / data that doesn't fit into any of the cluster is called an outlier.

EVOLUTION ANALYSIS:

-> Based on all the historical data we can ~~make~~ take efficient decision. making.

LEARNING:

1) Supervised. Learning
2) Unsupervised "
3) Semi supervised "
4) Reinforcement "

SUPERVISED: All data is labeled and the algorithms learned to predict the O/P from the i/P data.

UNSUPERVISED: All data is unlabelled and the algorithms learn to inherit structure from the i/P data

SEMI SUPERVISED: Some data is labelled but most of it is unlabelled and a mixture of supervised and unsupervised techniques can be used.

REINFORCEMENT: It allows machines and software agents to automatically determine the ideal behaviour within a specific context, in order to maximize its ~~perform~~ performance. Simple reward feedback is required for the agent to learn its behaviour.

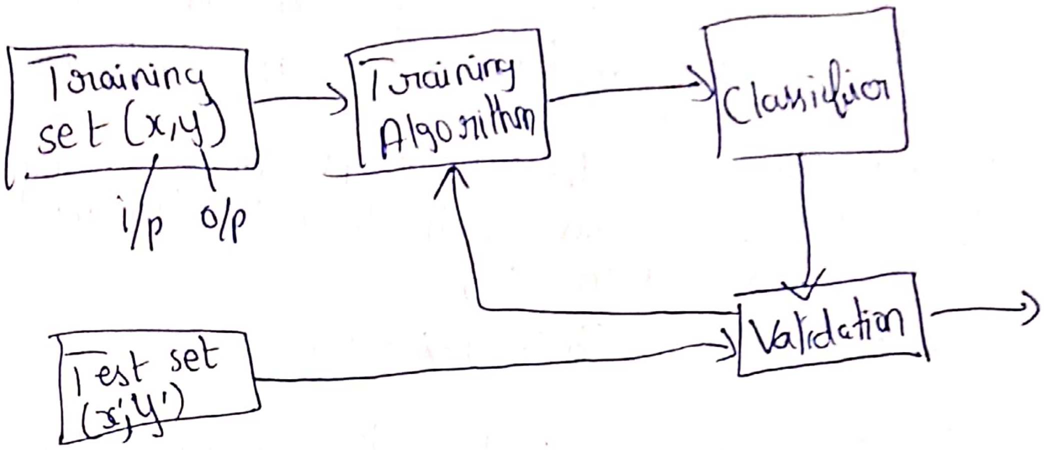
SOFTWARE AGENT: It is a computer program that reacts to its environment and runs without continuous & direct supervision to perform some function for an end user or another program.

Scanned with CamScanner

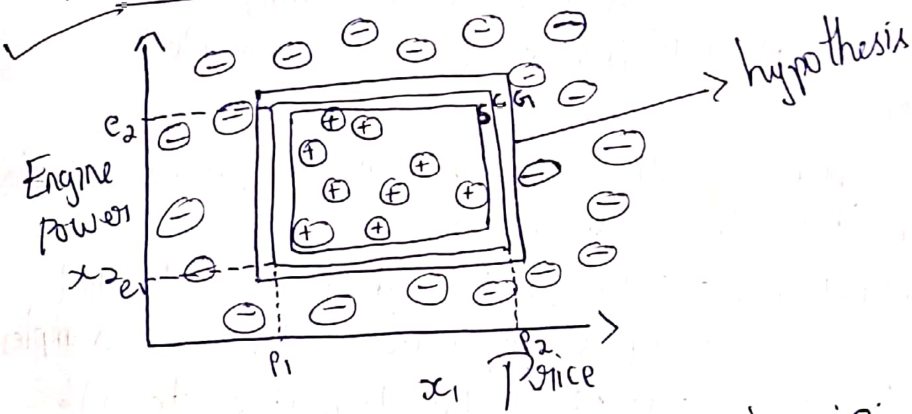
SUPERVISED LEARNING

$x_1 = (20000, 28)$ $\quad y_1 \Rightarrow$ Cannot buy computer

$x_2 = (90000, 35)$ $\quad y_2 \Rightarrow$ buys computer



LEARNING FROM EXAMPLES:



-> Let us say we want to learn the class 'C' - family car.

-> Family ~~car~~ car are assumed to be +ve examples and others are -ve examples.

-> Among all the features a car may have, the features that seperate the family car from other cars are price, engine power.

-> Let price be represented by x_1 and engine power by x_2

$$\therefore X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

-> Its label denote its type

$$\text{label} \leftarrow r = \begin{cases} 1 & \text{if } X \text{ is a +ve example} \\ 0 & \text{if } X \text{ is a -ve example} \end{cases}$$

-> Therefore each car is represented by such an ordered pair (X, r) and the training set contains N such samples.

$$X = \{x^t, r^t\}_{t=1}^{N}$$

→ Therefore for the concept or the hypothesis C,

$P_1 \leq$ price $\leq P_2$ AND $E_1 \leq$ Engine power $\leq E_2$

→ Like ~~this~~ this many different hypothesis can be drawn

→ All these hypothesis are put into the hypothesis class 'H'.

→ The particular hypothesis 'h' belonging to 'H' should be most approximately close to 'C'.

→ The empirical error should be less (the portion of training instances where predictions of 'h' do not match the required values given in 'H')

$$E(h/x) = \sum_{t=1}^{N} (h(x^t) \neq r^t)$$

* The value will be equal to 1 if there is an error and equal to 0 if there is no error

→ $h(x) = \begin{cases} 1, & \text{if } h \text{ classifies } x \text{ as a +ve example.} \\ 0, & \text{if } h \text{ classifies } x \text{ as a -ve example.} \end{cases}$

GENERALIZATION:

→ How well our hypothesis will correctly classify future examples that are not part of the training set.

→ Most specific hypothesis(S): It is the ~~tighest~~ tightest that includes all the +ve examples and none of the -ve examples

→ ~~Most~~ Most general hypothesis(G): It is the largest rectangle we can draw that includes all the +ve examples and none of the -ve examples.

→ Any $h \in H$ between S and G is a valid hypothesis with no error are said to be consistent with the training set and such 'h' make up the version space.

→ Margin is the distance between the boundary and the instances closest to it.

PAC (Probably Approximately Correct) learning

→ consider a concept class 'C' defined over

Error
- True Error: $error_D(L) = P_{x \sim D}(C(x) \neq L(x))$
- Training Error → 'n' no. of classes

$\boxed{m = \text{samplesize}}$ $error_S(L) \equiv \frac{1}{m} \sum_{i=1} ((L(n_i) \neq C(x))$

Scanned with CamScanner

Consistent hypothesis:

$$m \geq \frac{1}{\epsilon}\left(\ln|H| + \ln\left(\frac{1}{\delta}\right)\right)$$

NOISE:

Simple vs Complex Model

Learning multiple classes

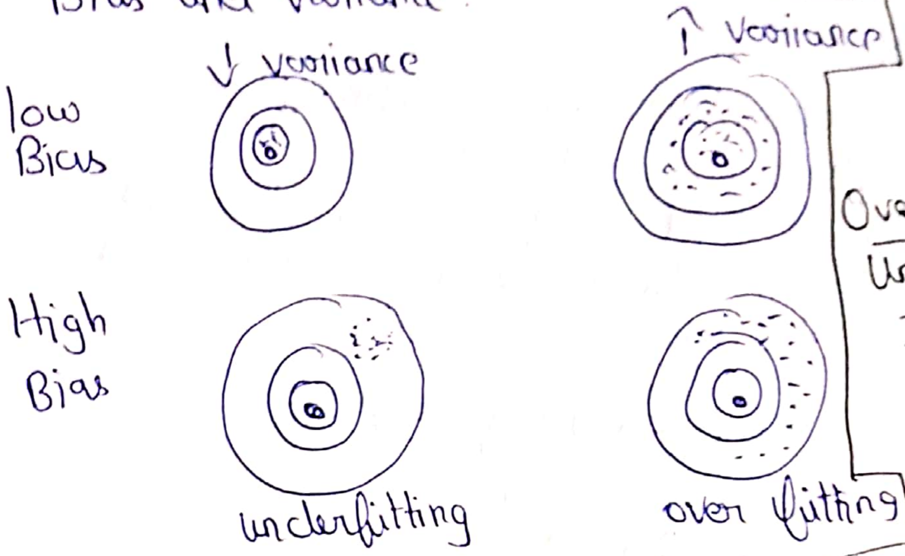
Bias and variance.

BIAS – Difference between the expected prediction of our model and the correct values which we are trying to predict.

VARIANCE: Stability of the model in response to new training samples.

Overfitting and Underfitting:

If a model has high variance then it is called as over fitting. If a model has high bias then it is called as underfitting.



Model Selection and generalisation:

1) Ill posed problem
2) Inductive Bias
3) Model Selection
4) Generalisation
5) Overfitting
6) Under fitting
7) Triple pair of
8) Cross validation
9) Validation set
10) Test set (publication set)

* How to detect High Bias?
  - → High training error
  - → Validation error is similar in magnitude to training error
* How to detect high variance?
  - → low training error.
  - → very high validation error

→ Complexity of hypothesis
→ Amount of training data
→ Generalisation error

REGRESSION: – linear
– Multiple

→ It is used for prediction. This method is one of the prediction data.

LINEAR REGRESSION

→ Linear regression is used for prediction.

→ $Y = \alpha + \beta X$

where, α; β are regression co-efficients.

→ It uses the method of least squares.

→ It estimates the best fit using straight line. It minimizes the error between the actual data and the estimate of that line.

$$\beta = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

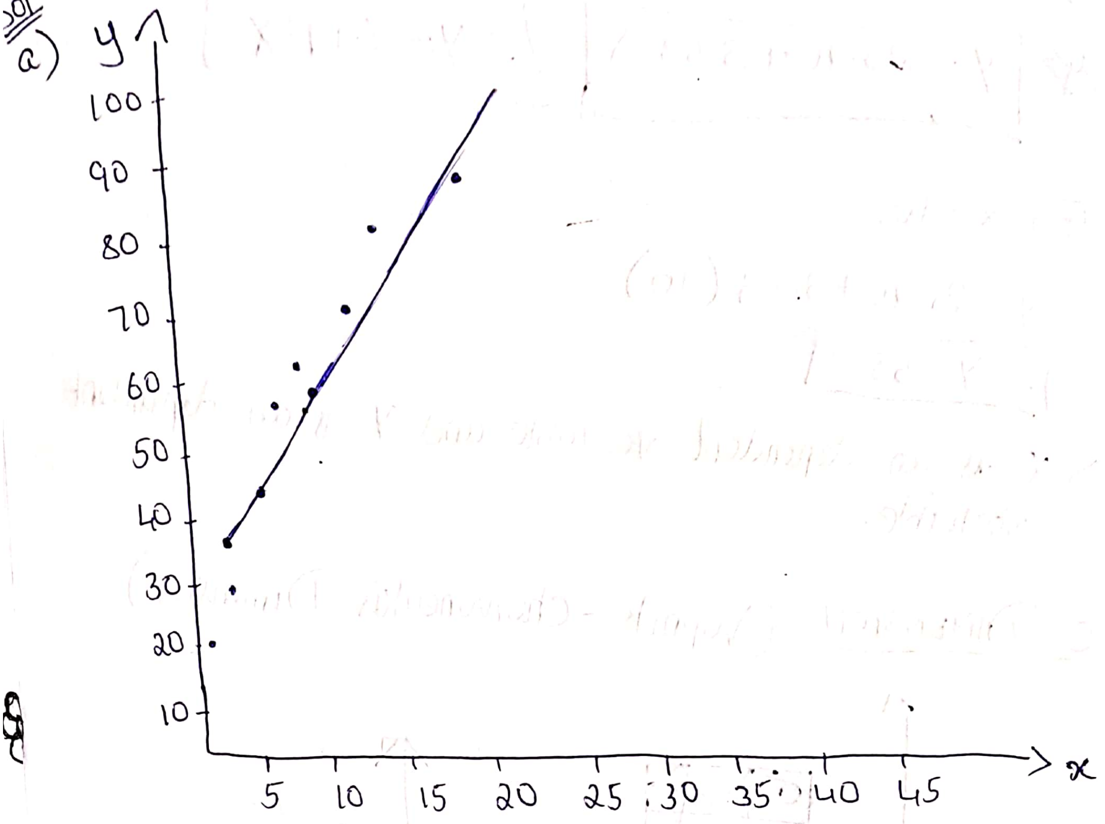
$$\alpha = \bar{y} - \beta \bar{x}$$

EXAMPLE:

x (years of experience)	y (salary in thousands)
3	30
8	57
9	64
13	72
3	36
6	43
11	59
21	90
1	20
16	83

) Plot the data. b) Apply the method of least squares.
c) Predict the value of y given $x=10$

Sol
a)



∴ From the Graph we infer that x and y have some relationship.

b) Method of least squares.

$$\beta = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$$

$$\cancel{= \frac{(3-9.1)(30-55.4)}{(3-9.1)^2}}$$

90-85
~~268.85~~
~~247.96~~+
~~575.36~~+
~~477.18~~
~~1269.4~~
~~358.9~~

$$\beta = \frac{\begin{array}{l}(3-9.1)(30-55.4)+(8-9.1)(57-55.4) \\ +(9-9.1)(64-55.4)+(13-9.1)(72-55.4) \\ +(3-9.1)(36-55.4)+(6-9.1)(43-55.4) \\ +(11-9.1)(59-55.4)+(21-9.1)(90-55.4) \\ +(1-9.1)(20-55.4)+(16-9.1)(83-55.4)\end{array}}{\begin{array}{l}(3-9.1)^2+(8-9.1)^2+(9-9.1)^2+(13-9.1)^2+(3-9.1)^2 \\ +(6-9.1)^2+(11-9.1)^2+(21-9.1)^2+(1-9.1)^2+(16-9.1)^2\end{array}}$$

$\beta = 3.5374$

$\alpha = \bar{y} - \beta \bar{x}$

$\boxed{\alpha = 23.20}$

$\therefore \boxed{Y = 23.20 + 3.53\,X}$ $\left[\because Y = \alpha + \beta X\right]$

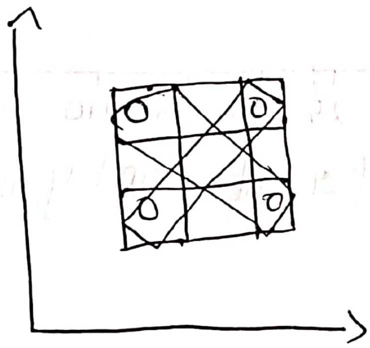
c) For $x = 10$,

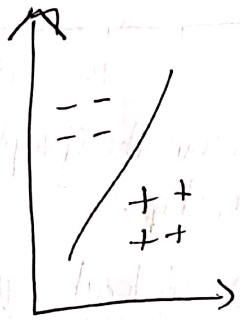
$Y = 23.20 + 3.53\,(10)$

$\boxed{\therefore Y = 58.5}$

→ X is an idependent variable and Y is an dependent variable.

VC DIMENSION: (Vapnik - Chervonenkis Dimension)





→ We have the dataset containing N points, these N points can be labelled in 2^N ways.

→ Find the hypothesis $h \in H$ that seperates the +ve examples from the -ve. Then we say H shatters N points, (i.e) any learning problem definable by N examples can be learned with no error by the hypothesis drawn from H.

→ The maximum number of points that can be shattered by H is called VC dimension of H and is denoted by VC(H) and measures the capacity of H.

OCCAM'S RAZOR: It states that simpler explanations are more plausible and any unnecessary complexity should be shaved off.

UNIT-II

Bayesian Decision Theory:

Naive Bayesian Classification:

$$P(C_i/x) = \frac{P(x/C_i)\,P(C_i)}{P(x)} \quad \text{(Bayes Theorem)}$$

→ For naive Bayesian classification we use,

$$\boxed{P(x/C_i)\,P(C_i)}$$

Problem: X

Age	Income	Student	Credict Rating	Class: Buys Computer
Youth	High	No	Fair	No
Youth	High	No	Excellent	No
middle aged	High	No	Fair	Yes
Senior	Medium	No	Fair	Yes
Senior	Low	Yes	Fair	Yes
Senior	Low	Yes	Excellent	No
middle aged	Low	Yes	Excellent	Yes
Youth	Medium	No	Fair	No
Youth	Low	Yes	Fair	Yes
Senior	Medium	Yes	Fair	Yes
Youth	Medium	Yes	Excellent	Yes
middle aged	Medium	No	Excellent	Yes
middle aged	High	Yes	Fair	Yes
Senior	Medium	No	Excellent	No

→ The tuple we wish to classify is,

X = (age = Youth, income = medium, student = Yes, Credict Rating = Fair)

Sol:

P(Ci):

i = 2 ⟹ Because two classes are present yes and No in class: Buys Computer.

$P(\text{buys\_Computer} = \text{yes}) = \frac{9}{14} = 0.64$

$P(\text{buys\_Computer} = \text{No}) = \frac{5}{14} = 0.357$

· P(x/Ci):

$P(\text{age} = \text{youth} / \text{buys\_Computer} = \text{Yes}) = 2/9 = 0.222$

$P(\text{age} = \text{youth} / \text{buys\_computer} = \text{No}) = 3/5 = 0.6$

$P(\text{income} = \text{medium} / \text{buys\_Computer} = \text{Yes}) = 4/9 = 0.444$

$P(\text{income} = \text{medium} / \text{buys\_computer} = \text{No}) = 2/5 = 0.4$

$P(\text{Student} = \text{yes} / \text{buys\_computer} = \cancel{\text{No}}\ \text{Yes}) = 6/9 = 0.667$

$P(\text{Student} = \text{yes} / \text{buys\_computer} = \text{No}) = 1/5 = 0.2$

$P(\text{Credictrating} = \text{fair} / \text{buys\_computer} = \text{Yes}) = 6/9 = 0.667$

$P(\text{Credictrating} = \text{fair} / \text{buys\_computer} = \text{No}) = 2/5 = 0.4$

$$P(X/C_i) = P(x_1/C_i) * P(x_2/C_i) * P(x_3/C_i) \ldots P(x_n/C_i)$$

∴ For yes class:

$$\begin{aligned} P(x / \text{buys\_computer} = \text{yes}) = {} & P(\text{age} = \text{yes} / \text{buys comp} = \text{Yes}) \\ & * P(\text{inc} = m / \text{buys comp} = \text{Yes}) \\ & * P(\text{st} = \text{Yes} / \text{buys comp} = \text{Yes}) \\ & * P(\text{cr} = \text{fair} / \text{buys comp} = \text{Yes}) \\ = {} & 0.04 \end{aligned}$$

∴ For No class:

$P(X / buys\_computer = No) = P(age = y / bc = No) * P(inc = m / bc = No) * P(cr = fair / bc = No) * P(st = y / bc = No)$

$= 0.019$

$P(X/C_i)P(C_i)$:

$P(X / buys\ computer = Yes) * P(buys\text{-}computer = yes)$

$= 0.04 \times 0.64$

$= 0.0256$

$P(X / buys\ computer = No) * P(buys\_computer = No)$

$= 0.019 \times 0.357$

~~= 0.0256~~

$= 0.0068$

∴ Here $P(X / buys\ computer = Yes) * P(buys.computer = yes)$ is greater than $P(X / buys\ computer = No) * P(buys\ computer = No)$

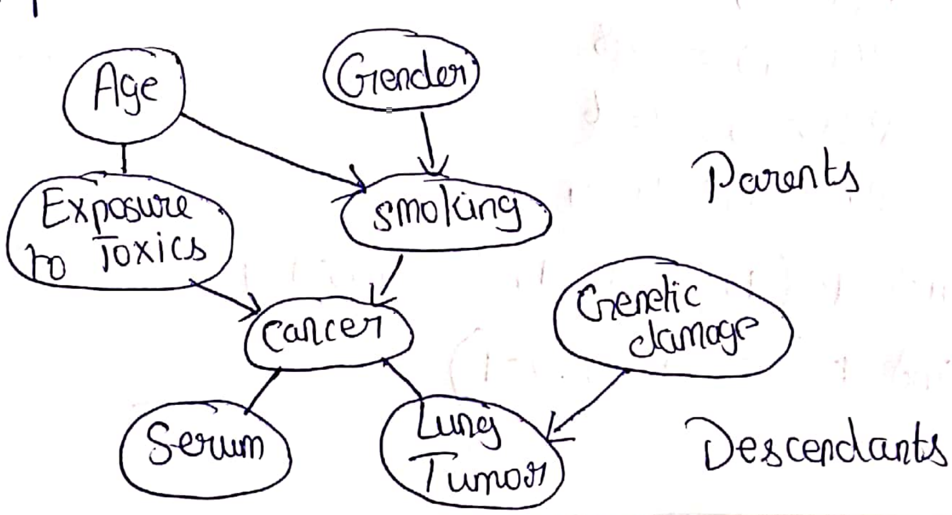
So, ∴ We conclude that the given tuple X = (age = youth, income = medium, student = yes, credit rating = fair) will buy a computer. That implies, X belongs to buys computer = Yes, class / category.

Bayesion Network:

→ Probabilistic graphical model - Models conditional probability distributions among the random variables.

→ Conditional independence - A variable (node) is conditionally independent of its non-descendant given its parents.

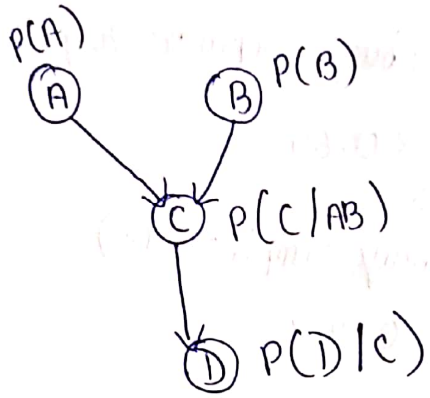
Eg



→ Cancer is independant of genetic damage given exposure to toxics and smoking.

Eg 1 Problems:

Suppose 4 random variables A, B, C, D



Joint probability distribution of random variable,

$P(A, B, C, D) = P(A)\, P(B)\, P(C/AB)\, P(D/C)$

P(C/AB)

AB	C=0	C=1
00	0.2	0.8
01	0.6	0.4
10	0.3	0.7
11	0.5	0.5

P(D/C)

	D=0	D=1
C=0	0.1	0.9
C=1	0.9	0.1

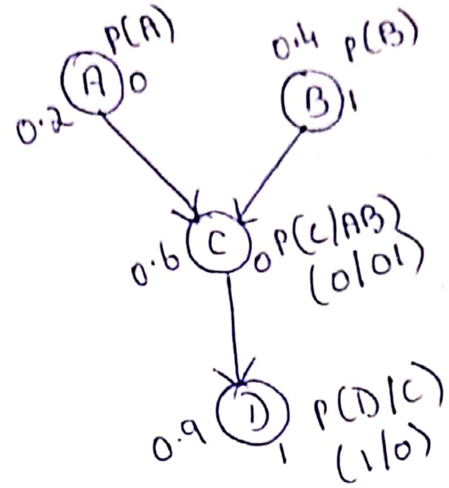
$P(A=0) = 0.2$
$P(A=1) = 0.8$
$P(B=0) = 0.6$
$P(B=1) = 0.4$

① Find $P(A=0, B=1, C=0, D=1)$

② Find $P(A=0, B=1, D=1)$

Solution:

(i)



$= P(A)\, P(B)\, P(C/AB)\, P(D/C)$

$= 0.2 \times 0.4 \times 0.6 \times 0.9$

$= 0.0432$

(2) $P(A=0, B=1, D=1)$

$= \sum_{C \in \{0,1\}} P(A=0, B=1, C=C, D=1)$

$= P(A=0, B=1, C=0, D=1) + P(A=0, B=1, C=1, D=1)$

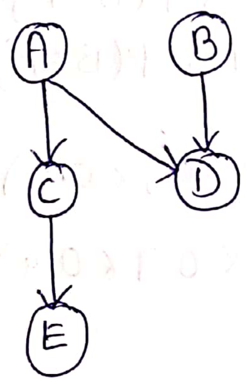
$= P(A)\, P(B)\, P(C/AB)\, P(D/C) + P(A)\, P(B)\, P(C/AB)\, P(D/C)$

$= (0.2 * 0.4 * 0.6 * 0.9) + (0.2 * 0.4 * 0.4 * 0.1)$

$= 0.0432 + 0.0288$

$= 0.072$

(*) BAYESIAN NETWORK: (Bayesian Belief Network)



Given: * A and B are independent.
* C is independent of B given A
* So D is independent of C given A and B
* E is independent of A, B and D given C.

$P(A=T) = 0.3$

$P(B=T) = 0.6$

$\cancel{P(C=T)}$ $P(C=T/A=T) = 0.8$

$P(C=T/A=F) = 0.4$

$P(D=T/A=T, B=T) = 0.7$

$P(D=T/A=T, B=F) = 0.8$

$P(D=T/A=F, B=T) = 0.1$

$P(D=T/A=F, B=F) = 0.2$

① Solve probablity of $P(D=T)$

② $P(D=F, C=T)$

Sol: ① $P(D=T)$:

$$
\begin{aligned}
P(D=T) &= P(D=T, A=T, B=T) \\
&\quad + P(D=T, A=T, B=F) \\
&\quad + P(D=T, A=F, B=T) \\
&\quad + P(D=T, A=F, B=F) \\
&= P(D=T/A=T, B=T)\, P(A=T, B=T) \\
&\quad + P(D=T/A=T, B=F)\, P(A=T, B=F) \\
&\quad + P(D=T|A=F, B=T)\, P(A=F, B=T) \\
&\quad + P(D=T|A=F, B=F)\, P(A=F, B=F) \\
&= P(D=T|A=T, B=T)\, P(A=T)\, P(B=T) \\
&\quad + P(D=T/A=T, B=F)\, P(A=T)\, P(B=F) \\
&\quad + P(D=T|A=F, B=T)\, P(A=F)\, P(B=T) \\
&\quad + P(D=T|A=F, B=F)\, P(A=F)\, P(B=F) \\
&= (0.7 \times 0.3 \times 0.6) + (0.8 \times 0.3 \times 0.4) \\
&\quad + (0.1 \times 0.7 \times 0.6) + (0.2 \times 0.7 \times 0.4) \\
&= 0.32 \;\;//
\end{aligned}
$$

② $P(D=F, C=T) = P(D=F, C=T, A=T, B=T)$
$+ P(D=F, C=T, A=T, B=F)$
$+ P(D=F, C=T, A=F, B=T)$
$+ P(D=F, C=T, A=F, B=F)$

$= P(D=F, C=T / A=T, B=T)\, P(A=T, B=T)$
$+ P(D=F, C=T / A=T, B=F)\, P(A=T, B=F)$
$+ P(D=F, C=T / A=F, B=T)\, P(A=F, B=T)$
$+ P(D=F, C=T / A=F, B=F)\, P(A=F, B=F)$

$= P(D=F / A=T, B=T)\, P(C=T / A=T, B=T)\, P(A=T, B=T)$
$+ P(D=F / A=T, B=F)\, P(C=T / A=T, B=F)\, P(A=T, B=F)$
$+ P(D=F / A=F, B=T)\, P(C=T / A=F, B=T)\, P(A=F, B=T)$
$+ P(D=F / A=F, B=F)\, P(C=T / A=F, B=F)\, P(A=F, B=F)$

$= P(D=F / A=T, B=T)\, P(C=T / A=T)\, P(A=T)\, P(B=T)$
$+ P(D=F / A=T, B=F)\, P(C=T / A=T)\, P(A=T)\, P(B=F)$
$+ P(D=F / A=F, B=T)\, P(C=T / A=F)\, P(A=F)\, P(B=T)$
$+ P(D=F / A=F, B=F)\, P(C=T / A=F)\, P(A=F)\, P(B=F)$

$= (0.3 \times 0.8 \times 0.3 \times 0.6) + (0.2 \times 0.8 \times 0.3 \times 0.4)$
$+ (0.9 \times 0.4 \times 0.7 \times 0.6) + (0.8 \times 0.4 \times 0.7 \times 0.4)$

$= 0.0432 + 0.0192 + 0.1512 + 0.0896$

$= 0.3032$ //