

Sr. No.	Module	Detailed Contents	Hrs
1	Business Intelligence-	Introduction and overview of BI-Effective and timely decisions, Data Information and knowledge, BI Architecture, Ethics and BI. BI Applications- Balanced score card, Fraud detection, Telecommunication Industry, Banking and finance, Market segmentation.	06
2	Prediction methods and models for BI	Data preparation, Prediction methods-Mathematical method, Distance methods, Logic method, heuristic method-local optimization technique, stochastic hill climber, evaluation of models	06
3	BI using Data Warehousing	Introduction to DW, DW architecture, ETL Process, Top-down and bottom-up approaches, characteristics and benefits of data mart, Difference between OLAP and OLTP. Dimensional analysis- Define cubes. Drill- down and roll- up – slice and dice or rotation, OLAP models- ROLAP and MOLAP. Define Schemas- Star, snowflake and fact constellations.	08
4	Data Mining and Preprocessing	Data mining- definition and functionalities, KDD Process, Data Cleaning: - Missing values, Noisy data, data integration and transformations. Data Reduction: - Data cube aggregation, dimensionality reduction- data compression, Numerosity reduction- discretization and concept hierarchy.	06
5	Associations and Correlation	Association rule mining:-support and confidence and frequent item sets, market basket analysis, Apriori algorithm, Incremental ARM, Associative classification- Rule Mining.	06
6	Classification and Prediction	Introduction, Classification methods:-Decision Tree- ID3, CART, Bayesian classification- Baye'stheorem(Naïve Bayesian classification),Linear and nonlinear regression.	08
7	Clustering	Introduction, categorization of Major, Clustering Methods:- partitioning methods- K-Means. Hierarchical- Agglomerative and divisive methods, Model- based- Expectation and Maximization.	08
8	Web mining and Text	Text data analysis and Information retrieval, text retrieval methods, dimensionality reduction for text	04

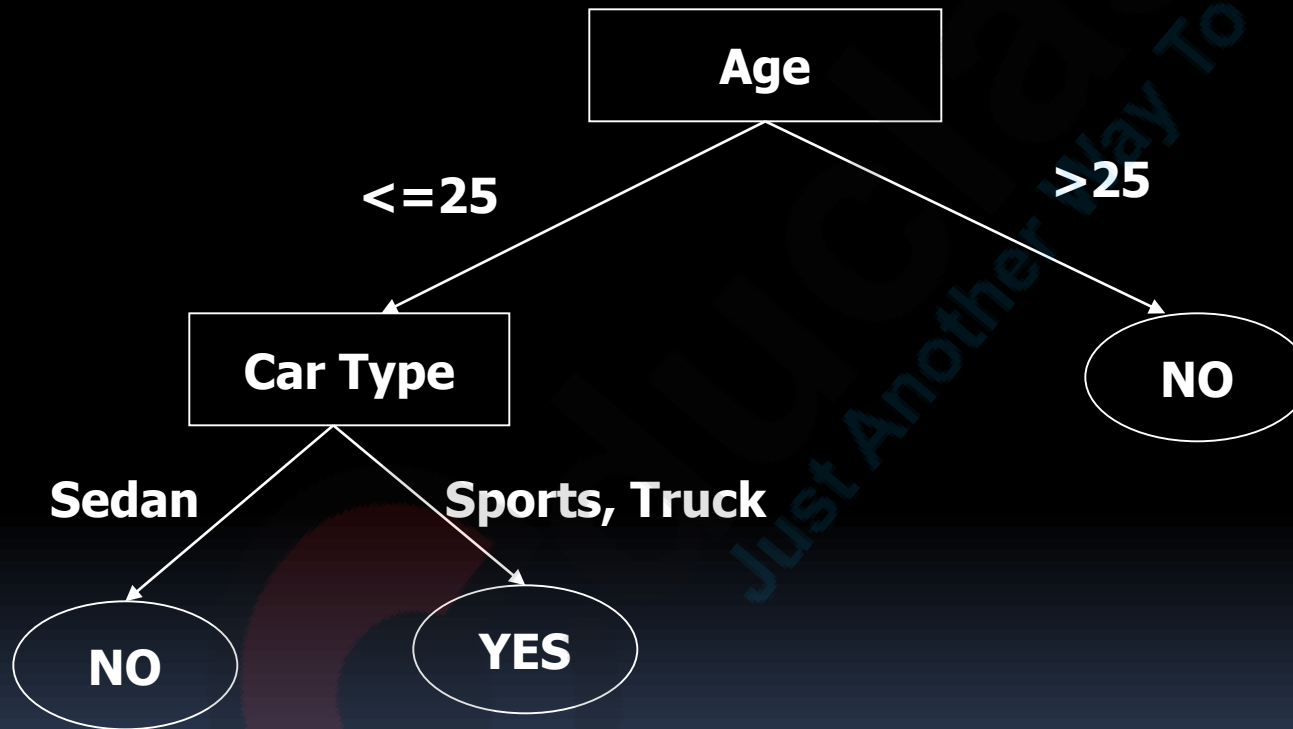
Outline

- Distance based algo
 - ✓ K - Nearest Neighbor Algo
- Decision Tree - based Algorithm
 - ✓ ID₃
 - ✓ C_{4.5}
 - ✓ CART
- Statistical algo
 - ✓ Bayesian Classification
- Neural Networks based algo
 - ✓ Propagation
- ✓ Linear and non linear regression

Tree-Structured rules

- Supervised learning
- The type of rule discussed can be represented by a tree.
- Trees that represent classification rules are called **classification trees** or **decision trees** & trees that represents regression rules are called **regression trees**.
- Tree-structured rules are very popular since they are easy to interpret and are very accurate.

Example



- * Above fig shows Insurance risk example Decision Tree
- * Each path from root node to a leaf node represents one classification rule.

Decision Trees

- Also called **classification tree**
- Graphical representation of set of classification rules
- Each internal node represents **predictor / splitting attribute**
- Each arc/ edge is labeled with **predicate or splitting criteria**
- Each leaf node is labeled with a class C_j

Decision Trees

Basic step

- Build the tree
- Apply the tree to the database

Just Another Way To Learn

Decision Trees

- A decision tree is usually constructed in two phases.
 - The growth phase
 - The pruning phase
- In **growth phase**, an overly large tree is constructed. This tree represents the record in the input database very accurately.
- In **pruning phase**, the final size of the tree is determined.
- The rules represented by the tree constructed in phase one are usually overspecialized.
- **By reducing the size of the tree, we generate a smaller number of more general rules that are better than a very large number of very specialized rules.**

Decision Trees

- The splitting criterion at a node is found through application of a split selection method.
- A split selection method is an algorithm that takes as input a relation and outputs the locally 'best' splitting criterion.
- Following is the decision tree induction schema:

Input: node n , partition D , split selection method S

Output: decision tree for D rooted at node n

BuildTree(Node n , Partition D , split selection method S)

Apply S to D to find the splitting criterion

If (a good splitting criterion is found)

 Create two children nodes n_1 & n_2 of n

 Partition D into D_1 & D_2 .

 BuildTree(n_1, D_1, S)

 BuildTree(n_2, D_2, S)

endif

ID3 Background

- “Iterative Dichotomizer 3”.
- Invented by Ross Quinlan in 1979.
- Generates Decision Trees using **Entropy**.
- **Information Gain** is used to select the most useful attribute for classification.
- Builds the tree in top down fashion.
- Succeeded by Quinlan’s C4.5 and C5.0 algorithms.

Entropy

$$\text{Gain}(A) = I(p,n) - E(A)$$

TEMP	p	n	I(p,n)
Hot	0	2	0
Mild	1	1	1
cold	1	0	0

- Introduced by Claude Shannon in 1948
- Quantifies "randomness"
- Lower value implies less uncertainty
- Higher value implies more uncertainty
- A completely homogeneous sample has entropy of 0
- An equally divided sample has entropy of 1
- Formula:

Entropy of Attribute

$$E(A) = \sum \frac{p_i + n_i}{p+n} [I_i(p,n)]$$

IG of the table
Or

Entropy of the

starting set or parent table

$$I(p,n) = \frac{-p}{p+n} \log_2 \left[\frac{p}{p+n} \right] - \frac{n}{p+n} \log_2 \left[\frac{n}{p+n} \right]$$

Information Gain (IG)

- The information gain is based on the decrease in entropy after a dataset is split on an attribute.
- Can decide upon which attribute creates the most homogeneous branches?
- Formula: **Gain**(A) = $I(p,n) - E(A)$

Outlook	p	n	$I(p,n)$
Sunny	2	3	0.970
Overcast	4	0	0
Rain	3	2	0.970

$$\begin{aligned} E(\text{outlook}) &= [(2+3)/(9+5)](0.970) + 0 + [(3+2)/14](0.970) \\ &= 0.692 \end{aligned}$$

$$\begin{aligned} \text{Gain}(\text{outlook}) &= IG - E(\text{outlook}) \\ &= 0.940 - 0.692 \\ &= 0.248 \end{aligned}$$

Outlook	p	n	I(p,n)
Sunny	2	3	0.970
Overcast	4	0	0
Rain	3	2	0.970

$$\mathbf{E}(\text{outlook}) = \left[\frac{(2+3)}{(9+5)} \right] (0.970) + 0 + \left[\frac{(3+2)}{14} \right] (0.970) \\ = \mathbf{0.692}$$

$$\text{Gain (outlook)} = \text{IG} - \mathbf{E}(\text{outlook}) \\ = 0.940 - 0.692 \\ = 0.248$$

IG = Entropy of original dataset - Entropies of split dataset

Entropies of split dataset = Weighted sum of entropies
after each of subdivided dataset

Weight of each dataset = fraction of dataset being placed
in that division

ID3

- A branch set with entropy of 0 is a leaf node.
- Otherwise, the branch needs further splitting to classify its dataset.
- The ID3 algorithm is run recursively on the non-leaf branches, until all data is classified.

Overfitting

- During the construction of a DT , the tree repeatedly splits the data into node to get successively pure subsets of data
- If nodes are fitting to noise in training data, model will not generalize well
- This occurs when model is too complex
- Complexity is determined by “no. of nodes” in the tree
- To avoid overfitting
 - **Post pruning**
 - Grow tree to max size , then prune based on validation set
 - Computationally expensive method
 - Replace sub tree with leaf node if generalization error improves or dose not change
 - **Pre pruning**
 - Stop growing the tree before fully grown to fit the training data
 - Stop splitting when not statistically significant

Advantages of using ID3

- Understandable prediction rules are created from the training data.
- Builds the fastest tree.
- Only need to test enough attributes until all data is classified.
- Finding leaf nodes enables test data to be pruned, reducing number of tests.
- Whole dataset is searched to create tree.

Disadvantages of using ID3

- Data may be over-fitted or over-classified, if a small sample is tested.
- Smaller decision trees should be preferred over larger ones. This algorithm usually produces small trees, but it does not always produce the smallest possible tree
- Only one attribute at a time is tested for making a decision.
- Classifying continuous data may be computationally expensive.

DT Advantages/Disadvantages

- Advantages:
 - DTs are easy to use and efficient.
 - Rules generated are easy to interpret and understand
 - They scale well for large databases as tree size is independent of database size.
 - Trees can be constructed for data with many attributes.
- Disadvantages:
 - Does not easily handle continuous data.
 - May suffer from over fitting.
 - Can be quite large – pruning is necessary.
 - Correlations among attributes in the database are ignored in DT process

C4.5

- A successor of ID₃
- Builds DT using divide and conquer, top down, recursive approach
- Builds DT based on information theory concept
- It chooses splitting attribute with **highest Gain Ratio**

$$\text{GainRatio}(D, S) = \frac{\text{Gain}(D, S)}{\text{Split entropy}}$$

- It is ratio of IG for a splitting attribute and entropy of an attribute split (ignoring classes)

Gain Ratio

- Formula: **Gain ratio**(A) = GAIN(A) / **split entropy**(A)

temp	p	n	I(p,n)
Hot	2	2	1
Mild	4	2	0.09
cool	3	1	0.81

$$I(2,2)=1$$

$$I(4,2)= -4/6 \log_2(4/6) - 2/6 \log_2(2/6)$$

$$E(\text{temp})= [(2+2)/14](1) + [(4+2)/14](0.09) + [(3+1)/14](0.81) \\ = 0.9110$$

$$\text{Gain}(\text{temp})= IG - E(\text{outlook}) \\ = 0.940 - 0.9110 \\ = 0.0292$$

$$\text{Split info}(\text{temp})= -4/14 \log_2(4/14) - 6/14 \log_2(6/14) - 4/14 \log_2(4/14)$$

$$\text{Split info}(\text{temp})= 0.926$$

$$\text{Gain Ratio}(\text{temp})= 0.0292 / 0.926$$

C4.5

1. Handles both continuous and discrete attributes
 - The basic idea is to divide the data into ranges based on the attribute values for that item that are found in the training sample
2. Handling training data with missing attribute values
 - Missing attribute values are simply not used in gain ratio and entropy calculations
 - To classify a record with a missing attribute value, the value for that item can be predicted based on what is known about the attribute values for the other records

C4.5

3. Pruning trees after creation

- C4.5 goes back through the tree once it's been created and attempts to remove branches that do not help by replacing them with leaf nodes

Just Another Way to Learn

C4.5

EXAMPLE

To calculate the GainRatio for the gender split, we first find the entropy associated with the gender split (ignoring classes)

$$H(9/15, 6/15) = 9/15 \log(15/9) + 6/15 \log(15/6) = 0.292$$

This gives the GainRatio value for the gender attribute as

$$\frac{0.09688}{0.292} = 0.332$$

The entropy for the split on height (ignoring classes) is :

$$H(4/15, 7/15, 2/15, 2/15)$$

CART

- *Classification and regression tree*
- If the target variable is nominal (categorical) then the tree is called **Classification tree**.
- If the target variable is numerical (continuous) then the tree is called **Regression tree**.
- CART handles missing data by ignoring them in calculating the goodness of split on the attribute.
- CART contains pruning strategy.

CART

- *Classification and regression trees (CART)* is a technique that generates a **binary decision tree**
- Unlike ID₃, however, where a child is created for each subcategory, only two children are created
- The splitting is performed around what is determined to be the best split point.
- At each step, an exhaustive search is used to determine the best split, where "best" is defined by a measure $\phi(s/t)$

CART

- Create Binary Tree
- Formula to choose split point, s , for node t :

$$\Phi(s/t) = 2P_L P_R \sum_{j=1}^m | P(C_j | t_L) - P(C_j | t_R) |$$

- This formula is evaluated at the current node, t , and for each possible splitting attribute and criterion, s

CART

- L = left subtree of the current node
- R = Right subtree of the current node .
- P_L = probability that a tuple in the training set will be on the Left side of the tree
- P_R = probability that a tuple in the training set will be on the Right side of the tree
This is defined as $\frac{[\text{tuples in subtree}]}{[\text{tuples in training set}]}$
- $P(C_j|t_L)$ is the probability that a tuple is in class, C_j , and in the left subtree
- $P(C_j|t_R)$ is the probability that a tuple is in class, C_j , and in the right subtree
- This is defined as the $\frac{[\text{tuples of class } j \text{ in subtree}]}{[\text{tuples at the target node}]}$
- At each step, only one criterion is chosen as the best over all possible criteria

TABLE 4.1: Data for Height Classification

Name	Gender	Height	Output1
Kristina	F	1.6 m	Short
Jim	M	2 m	Tall
Maggie	F	1.9 m	Medium
Martha	F	1.88 m	Medium
Stephanie	F	1.7 m	Short
Bob	M	1.85 m	Medium
Kathy	F	1.6 m	Short
Dave	M	1.7 m	Short
Worth	M	2.2 m	Tall
Steven	M	2.1 m	Tall
Debbie	F	1.8 m	Medium
Todd	M	1.95 m	Medium
Kim	F	1.9 m	Medium
Amy	F	1.8 m	Medium
Wynette	F	1.75 m	Medium

Gender	short	medium	Tall	Total
F	3	6	0	9
M	1	2	3	6

Output1

Kristina

F

1.6 m

Short

Jim

M

2 m

Tall

$$\Phi(s/t) = 2P_L P_R \sum_{j=1}^m |P(C_j | t_L) - P(C_j | t_R)|$$

Medium

Medium

Short

Bob

M

1.85 m

Medium

Kathy

F

1.6 m

Short

Dave

M

1.7 m

Short

Worth

M

2.2 m

Tall

Steven

M

2.1 m

Tall

Debbie

F

1.8 m

Medium

Todd

M

1.95 m

Medium

Kim

F

1.9 m

Medium

Amy

F

1.8 m

Medium

Wynette

F

1.75 m

Medium

Gender	short	medium	Tall	Total
F	3	6	0	9
M	1	2	3	6

	3-1=2	6-2=4	3-0=3
--	-------	-------	-------

gender

F

M

$$\Phi(\text{gender}) = 2 * 9/15 * 6/15 * (2/15 + 4/15 + 3/15) = 0.224$$

S=3

M=6

T=0

S=1

M=2

T=3

$$\Phi(s/t) = 2P_L P_R \sum_{j=1}^m |P(C_j | t_L) - P(C_j | t_R)|$$

\downarrow \downarrow
 9/15 6/15

height	Less than	Greater than	Total
1.6	0	15	15
1.7	2	13	11
1.8	5	10	5
1.9	9	6	3
2	12	3	9

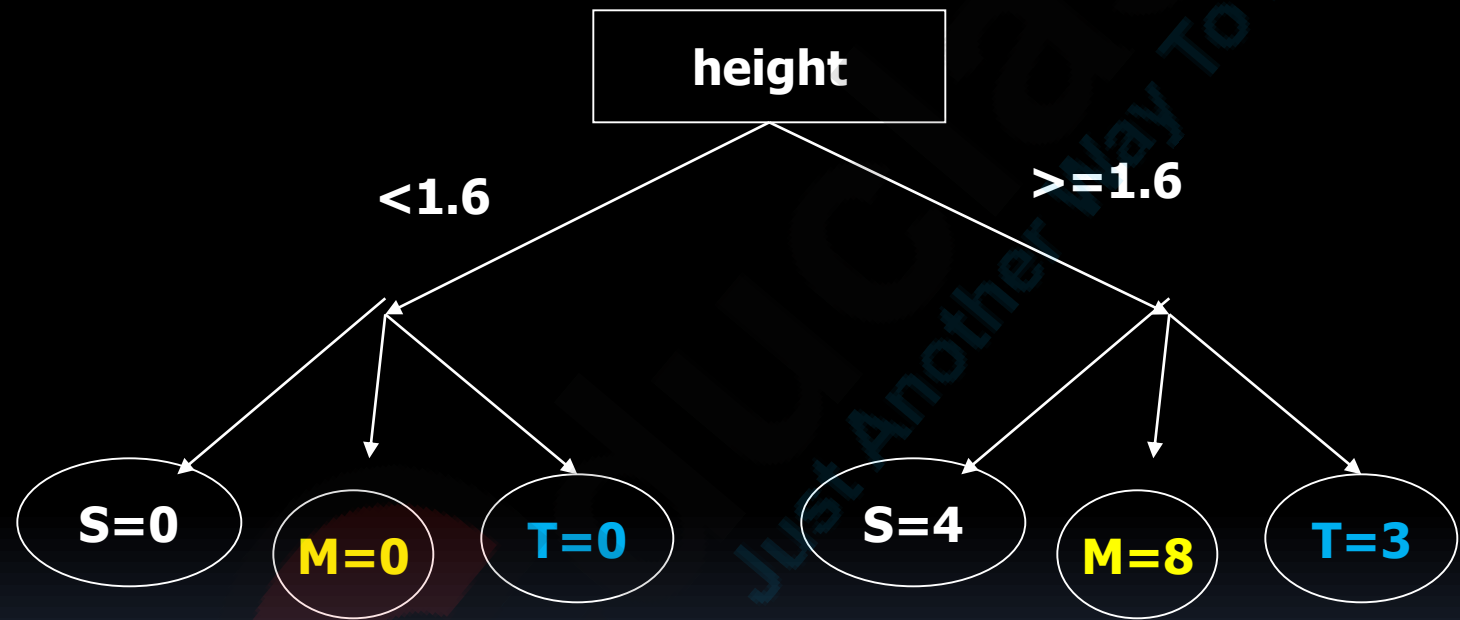
Just Another Way To Learn

height	S	M	T	Total
<1.6	0	0	0	0
>=1.6	4	8	3	15

$$\Phi(s/t) = 2P_L P_R \sum_{j=1}^m |P(C_j | t_L) - P(C_j | t_R)|$$

\downarrow \downarrow
 $2 * 0/15 * 15/15 *$

4	8	3
---	---	---



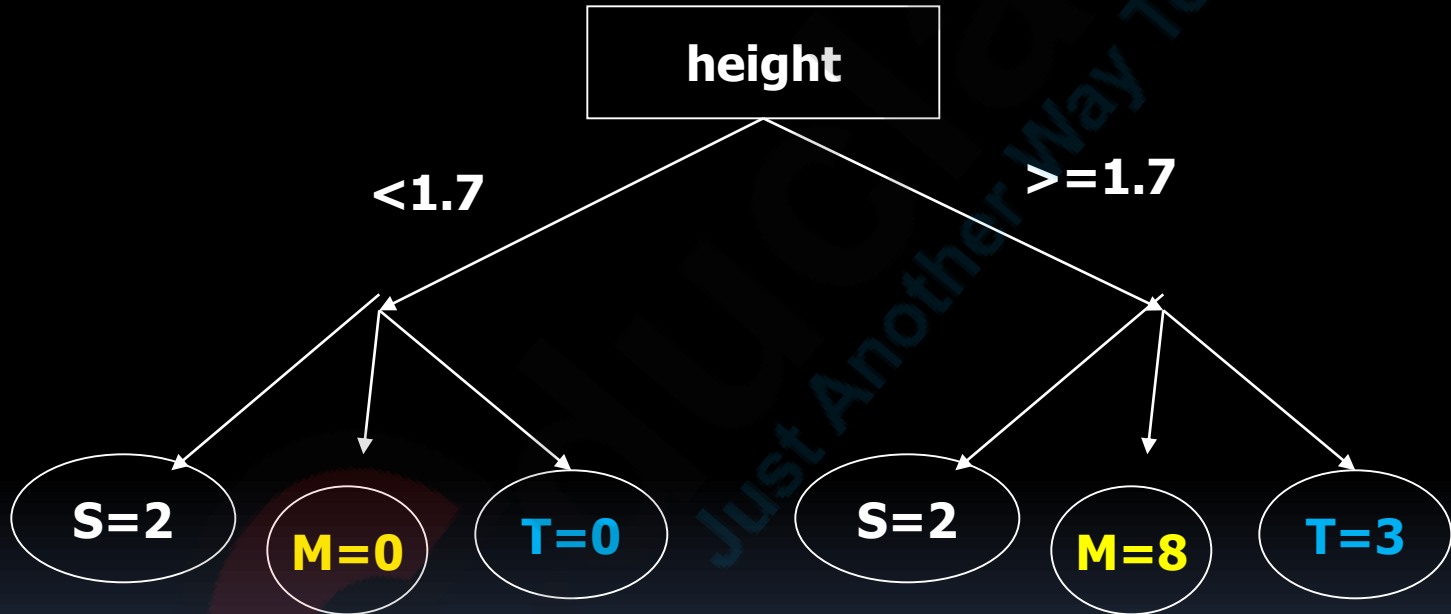
$$\Phi(1.6) = 2 * 0/15 * 15/15 (4/15 + 8/15 + 3/15) = 0$$

height	S	M	T	Total
<1.7	2	0	0	2
>=1.7	2	8	3	13

	0	8	3
--	---	---	---

$$\Phi(s/t) = 2P_L P_R \sum_{j=1}^m |P(C_j | t_L) - P(C_j | t_R)|$$

$2 * \frac{2}{15} * \frac{13}{15} * (0 + \frac{8}{15} + \frac{3}{15}) = 0.169$
 $2 * \frac{0}{15} * \frac{15}{15} *$

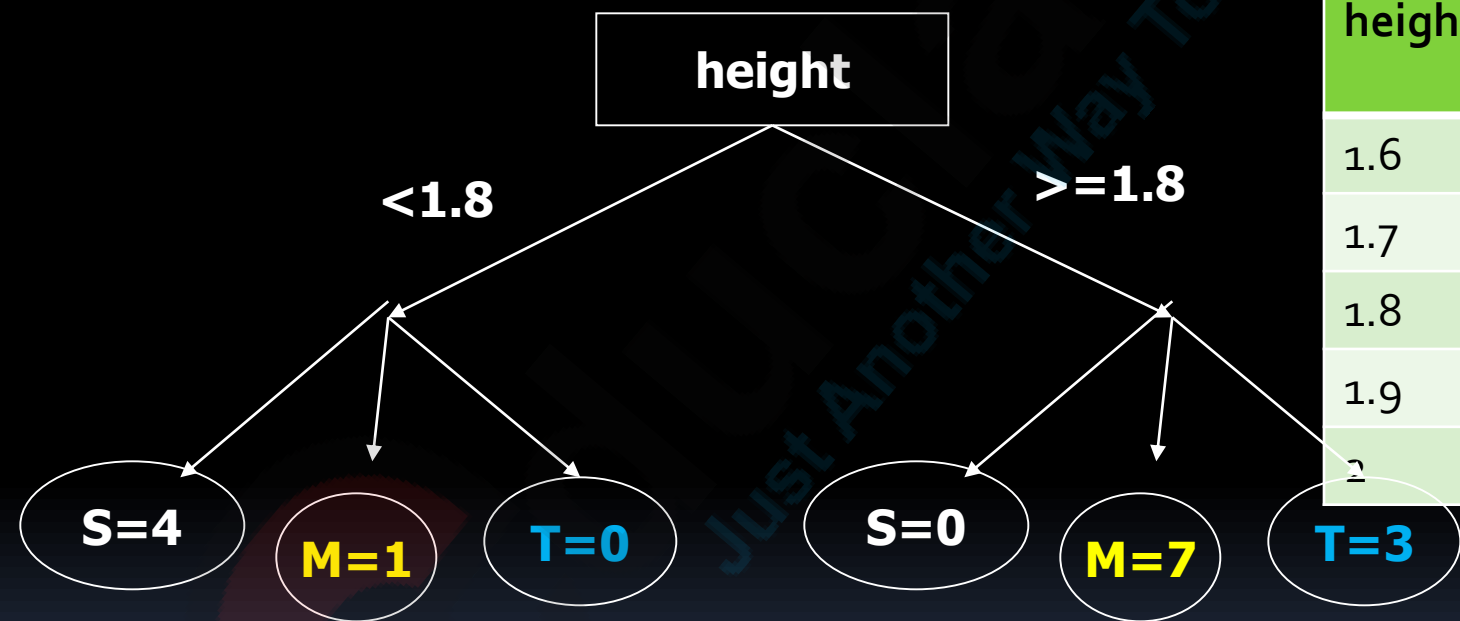


$$\Phi(1.7) = 2 * \frac{2}{15} * \frac{13}{15} (0/15 + 8/15 + 3/15) = 0.169$$

height	S	M	T	Total
<1.8	4	1	0	5
>=1.8	0	7	3	10

	4	6	3
--	---	---	---

$$\Phi(s/t) = 2P_L P_R \sum_{j=1}^m |P(C_j | t_L) - P(C_j | t_R)|$$



height	Less than	
1.6	0	
1.7	2	
1.8	5	
1.9	9	
2	12	

$$\Phi(1.8) = 2 * 5/15 * 10/15 (4/15 + 6/15 + 3/15) = 0.385$$

CART Example

- At the start, there are six choices for split point (right branch on equality):
 - $P(\text{Gender}) = 2(6/15)(9/15)(2/15 + 4/15 + 3/15) = \mathbf{0.224}$
 - $P(1.6) = \mathbf{0}$
 - $P(1.7) = 2(2/15)(13/15)(0 + 8/15 + 3/15) = \mathbf{0.169}$
 - $P(1.8) = 2(5/15)(10/15)(4/15 + 6/15 + 3/15) = \mathbf{0.385}$
 - $P(1.9) = 2(9/15)(6/15)(4/15 + 2/15 + 3/15) = \mathbf{0.256}$
 - $P(2.0) = 2(12/15)(3/15)(4/15 + 8/15 + 3/15) = \mathbf{0.32}$
- Split at 1.8

CART Example

- $P(1.6) = p(\text{height} < 1.6) = 0$
- $P(1.7) = p(\text{height} < 1.7) = 2 * (2/15) * (13/15) [|2/15 - 2/15| + |0 - 8/15| + |0 - 5/15|]$
- $P(1.8) = p(\text{height} < 1.8) = 2 * (5/15) * (10/15) [|4/15 - 0| + |1/15 - 7/15| + |0 - 3/15|]$
- $P(1.9) = p(\text{height} < 1.9) = 2 * (9/15) * (6/15) [|4/15 - 0| + |5/15 - 3/15| + |0 - 3/15|]$
- $P(2.0) = p(\text{height} < 2.0) = 2 * (12/15) * (3/15) [|4/15 - 0| + |8/15 - 0| + |0 - 3/15|]$


Bayesian Classification

- Bayes Rule or Bayes Theorem is
- Suppose there are m different hypotheses then

$$P(x_i) = \sum P(x_i | h_j)P(h_j)$$


$$P(h_1 | x_i) = \frac{P(x_i | h_1)P(h_1)}{P(x_i)}$$

- Here $P(h_1 | x_i)$ is called the posterior probability, while $P(h_1)$ is the prior probability associated with hypothesis h_1
- $P(x_i)$ is the probability of the occurrence of data value x_i and $P(x_i | h_1)$ is the conditional probability that, given a hypothesis, the tuple satisfies it.



Bayesian Classification

Just Another Way To Learn



Car No.	color	Type	Origin	stolen
1	Red	sports	domestic	Y
2	Red	sports	Domestic	N
3	Red	Sports	Domestic	Y
4	Yellow	Sports	Domestic	N
5	Yellow	Sports	Importer	Y
6	Yellow	SUV	Importer	N
7	Yellow	SUV	Importer	Y
8	Yellow	SUV	Domestic	N
9	Red	SUV	Importer	N
10	Red	Sports	Importer	Y

$$P(y) = 5/10$$

$$P(n) = 5/10$$

Color	
$P(\text{red} y) = 3/5$	$P(\text{red} N) = 2/5$
$P(\text{yellow} y) = 2/5$	$P(\text{yellow} N) = 3/5$

Type	
$P(\text{SUV} y) = 1/5$	$P(\text{suv} N) = 3/5$
$P(\text{sports} y) = 4/5$	$P(\text{sports} n) = 2/5$

Origin	
$P(\text{dom} y) = 2/5$	$P(\text{dom} N) = 3/5$
$P(\text{imp} y) = 3/5$	$P(\text{imp} n) = 2/5$

Sample X=(red&SUV&DOM) decision=?
Unlabeled sample

$$P(X|Y) = P(\text{red}|Y)P(\text{suv}|Y)P(\text{dom}|Y)$$

$$P(X|Y) = 3/5 * 1/5 * 2/5 = 0.048$$

$$P(X|N) = P(\text{red}|N)P(\text{suv}|N)P(\text{dom}|N)$$

$$P(X|N) = 2/5 * 3/5 * 3/5 = 0.144$$

$P(X|N) > P(X|Y)$ therefore sample X is class "N"

Car No.	A1	A2	A3	Class
1	A	C	A	C1
2	C	A	C	C1
3	A	A	C	C2
4	B	C	A	C2
5	c	c	b	C2

$$P(c1) = 2/5$$

$$P(c2) = 3/5$$

Sample X = A1=c, A2=c and A3=a class ?

Sample Y = A1=a, A2=c and A3=b class ?

A1	
$P(a C1) =$	$P(a C2) =$
$P(b c1) =$	$P(b c2) =$
$P(c c1) =$	$P(c c2) =$

A2	
$P(a C1) =$	$P(a C2) =$
$P(b c1) =$	$P(b c2) =$
$P(c c1) =$	$P(c c2) =$

A3	
$P(a C1) =$	$P(a C2) =$
$P(b c1) =$	$P(b c2) =$
$P(c c1) =$	$P(c c2) =$

$$P(X|c1) P(c1) = P(A1|c1)P(A2|c1)P(A3|c1) P(c1)$$

$$P(X|c2) P(c2) = P(A1|c2)P(A2|c2)P(A3|c2) P(c2)$$


$P(X | c1) > P(X | c2)$ therefore sample X is class "c1"



UNIVERSITY OF
Just Another Way To Learn




UOL
Just Another Way To Learn



Bayesian Classification

Just Another Way To Learn




Bayesian Classification

- Assuming that the contribution by all attributes are independent and that each contributes equally to the classification problem, a simple classification scheme called *naive Bayes* classification has been proposed that is based on Bayes rule of conditional probability
- By analyzing the contribution of each "independent" attribute, a conditional probability is determined
- A classification is made by combining the impact that the different attributes have on the prediction to be made
- The approach is called "naive" because it assumes the independence between the various attribute values

Statistical-based algorithm (Bayesian Classification)

- When classifying a target tuple, the conditional and prior probabilities generated from the training set are used to make the prediction
- This is done by combining the effects of the different attribute values from the tuple
- Suppose that tuple t_i has p independent attribute values $\{x_{i1}, x_{i2}, \dots, x_{ip}\}$. From the descriptive phase, we know $P(x_{ik}|C_j)$, for each class C_j and attribute x_{ik}
- We then estimate $P(t_i|C_j)$ by $P(t_i|C_j) = \prod P(x_{ik}|C_j)$
- We then have the needed prior probabilities $P(C_j)$ for each class and the conditional probability $P(t_i|C_j)$
- To calculate $P(t_i)$, we can estimate the likelihood that t_i is in each class. This can be done by finding the likelihood that this tuple is in each class and then adding all these values



Statistical-based algorithm (Bayesian Classification)



- The probability that t_i is in a class is the product of the conditional probabilities for each attribute value
 - The posterior probability $P(C_j|t_i)$ is then found for each class
 - The class with the highest probability is the one chosen for the tuple.
- 
- 

TABLE 4.1: Data for Height Classification

Name	Gender	Height	Output1	Output2
Kristina	F	1.6 m	Short	Medium
Jim	M	2 m	Tall	Medium
Maggie	F	1.9 m	Medium	Tall
Martha	F	1.88 m	Medium	Tall
Stephanie	F	1.7 m	Short	Medium
Bob	M	1.85 m	Medium	Medium
Kathy	F	1.6 m	Short	Medium
Dave	M	1.7 m	Short	Medium
Worth	M	2.2 m	Tall	Tall
Steven	M	2.1 m	Tall	Tall
Debbie	F	1.8 m	Medium	Medium
Todd	M	1.95 m	Medium	Medium
Kim	F	1.9 m	Medium	Tall
Amy	F	1.8 m	Medium	Medium
Wynette	F	1.75 m	Medium	Medium

Bayesian Classification

- There are 4 tuples classified as short, 8 as medium, and 3 as tall.
- The Output classification uses the simple divisions shown below:
 $2 \text{ m} \leq \text{Height}$ Tall
 $1.7 \text{ m} < \text{Height} < 2 \text{ m}$ Medium
 $\text{Height} \leq 1.7 \text{ m}$ Short
- The Output₂ results require a much more complicated set of divisions using, both height and gender attributes.
- To facilitate classification, we divide the height attribute into six ranges:
 $(0, 1.6]$, $(1.6, 1.7]$, $(1.7, 1.8]$, $(1.8, 1.9]$, $(1.9, 2.0]$, $(2.0, \infty)$

Just Another Way To Learn

Statistical-based algorithm (Bayesian Classification)

- With these training data, we estimate the prior probabilities:

$$P(\text{short}) = 4/15 = 0.267, P(\text{medium}) = 8/15 = 0.533, \\ \text{and } P(\text{tall}) = 3/15 = 0.2$$

- We use these values to classify a new tuple. For example, suppose we wish to classify $t = (\text{Adam}, M, 1.95 \text{ m})$
- By using these values and the associated probabilities of gender and height, we obtain the following estimates:

$$P(t|\text{short}) = 1/4 \times a = a$$

$$P(t|\text{medium}) = 2/8 \times 1/8 = 0.031$$

$$P(t|\text{tall}) = 3/3 \times 1/3 = 0.333$$

Prediction

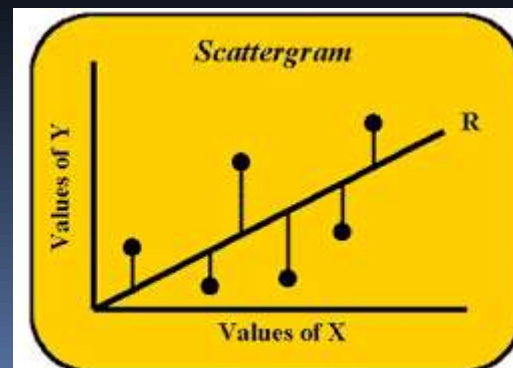
- Dependent variable , y
 - The variable whose values we want to explain or forecast
- Independent variable , x
 - Variable that explains the other
- **Linear regression**
 - assumes a linear relationship between input variable and output variable
- **Logistic regression**
 - Used when the dependent variable is binary
 - 0/1, T/F, Y/N

Linear Regression

- Objective
 - To establish if there is a relationship between two variables.
 - Income & spending
 - Students' weight & exam score
 - Forecast new observation
 - Sales in next quarter

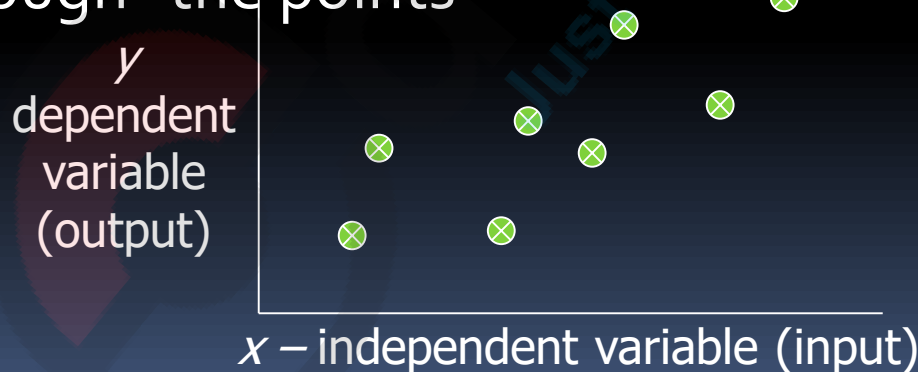
Just Another Way To Learn

- If the scatter diagram indicates some relationship between two variable x and y then the dots of the scatter diagram will be concentrated round a curve
- The curve is called the curve of regression and the relationship is said to the be expressed by means of **curvilinear regression**
- In the particular case, when the curve is a **straight** line, it is called a line of regression and the regression is said to be linear.



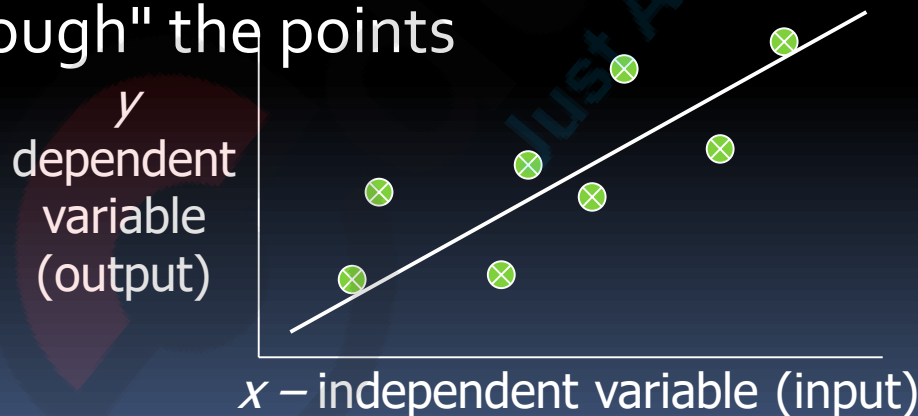
Regression

- For classification the output(s) is nominal
- In regression the output is continuous
 - Function Approximation
- Many models could be used – Simplest is linear regression
 - Fit data with the best hyper-plane which "goes through" the points



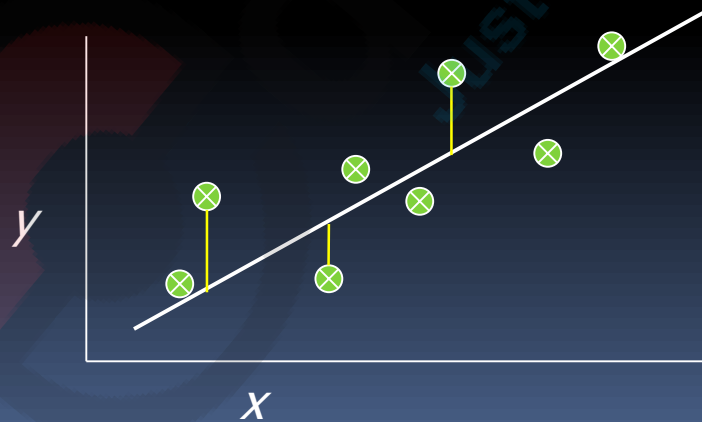
Regression

- For classification the output(s) is nominal
- In regression the output is continuous
 - Function Approximation
- Many models could be used – Simplest is linear regression
 - Fit data with the best hyper-plane which "goes through" the points



Regression

- For classification the output(s) is nominal
- In regression the output is continuous
 - Function Approximation
- Many models could be used – Simplest is linear regression
 - Fit data with the best hyper-plane which "goes through" the points
 - For each point the differences between the predicted point and the actual observation is the *residue*



Simple Linear Regression

- For now, assume just one (input) independent variable x , and one (output) dependent variable y
 - Multiple linear regression assumes an input vector \mathbf{x}
 - Multivariate linear regression assumes an output vector \mathbf{y}
- We will "fit" the points with a line (i.e. hyper-plane)
- Which line should we use?
 - Choose an objective function
 - For simple linear regression we choose sum squared error (SSE)
 - $\sum (\text{predicted}_i - \text{actual}_i)^2 = \sum (\text{residue}_i)^2$
 - Thus, find the line which minimizes the sum of the squared residues (e.g. least squares)

$$Y = \beta_0 + \beta_1 x$$

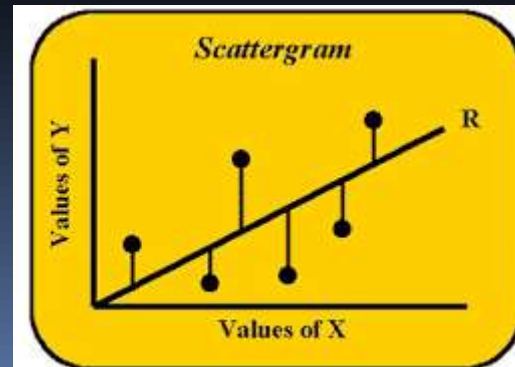
- The equation of the line of regression of is $y = a + bx$, where y is dependent variable and x is independent variable.

- The line of regression always passes through point (\bar{x}, \bar{y}) , $a = \bar{y} - b\bar{x}$

b is the slope of the line $r \frac{\sigma_y}{\sigma_x}$

line gives the best estimate of y given a value of x .

$Y = 4 + 2x$, for every increase in x , y increase 2 times



Regression line

$y = a + bx$ substitute the values of a and b

$$a = \bar{y} + b\bar{x} \quad b = \frac{r\sigma_y}{\sigma_x}$$

Consumption = 49.13 + (0.85) Income + error
Every increase in income the consumption will increase 0.85 times

where $r = \text{cov}(x, y) / \sigma_x \sigma_y$

$$\text{Cov}(X, Y) = [1/n \sum xy] - \bar{x} \bar{y}$$

' a ' is the intercept and ' b ' is the slope of the line

Examples

- Calculate the regression line of y on x for the following data. Also obtain prediction of y which should corresponding on the average to $x = 6.2$

x	1	2	3	4	5	6	7	8	9
y	9	8	10	12	11	13	14	16	15

$$r_{xy} = \text{cov}(x, y) / \sigma_x \sigma_y$$


$$\text{Cov}(X, Y) = [1/n \sum xy] - \bar{x} \bar{y}$$

$$\sigma_x^2 = (1/n \sum X^2) - \bar{X}^2$$

$$a = \bar{y} - b\bar{x} \quad b = \frac{r\sigma_x}{\sigma_y} \quad y = a + bx \quad \text{substitute the values of } a \text{ and } b$$



□ **Linear** regression

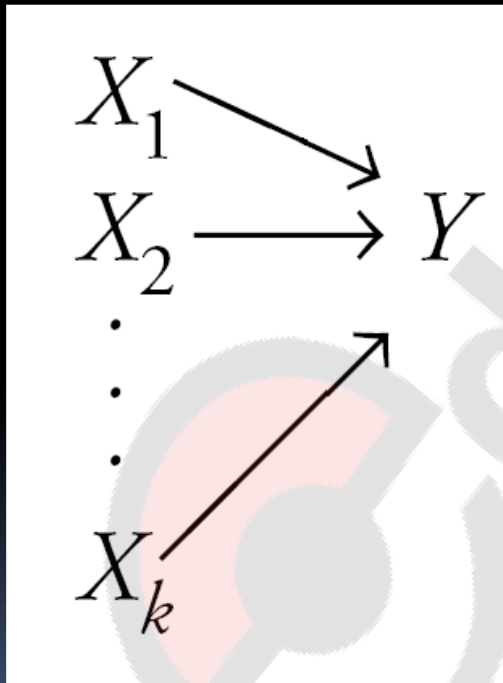
- not applicable for most complex problems
 - donot work with non numeric data
 - Assume a linear relationship
 - The straight line values can be greater than 1 and less than 0
 - Cannot be used as the probability of occurrence of target class
- 
- Just Another Way To Learn

Regression

Simple regression considers the relation between a single explanatory variable and response variable

$$X \rightarrow Y$$

Multiple regression simultaneously considers the influence of multiple explanatory variables on a response variable Y



The intent is to look at the independent effect of each variable.

Regression Modeling

- A simple regression model (one independent variable) fits a regression *line* in 2-dimensional space
- A multiple regression model with two explanatory variables fits a regression *plane* in 3-dimensional space

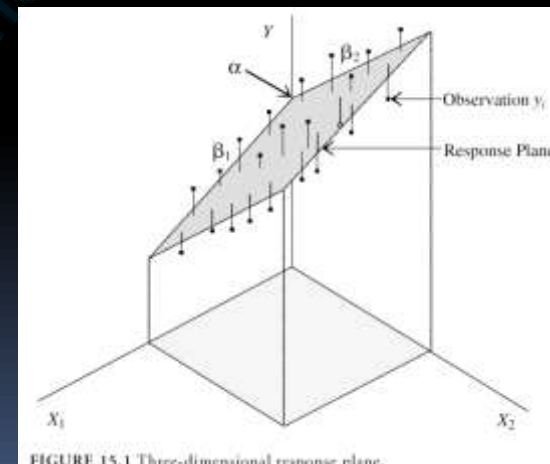
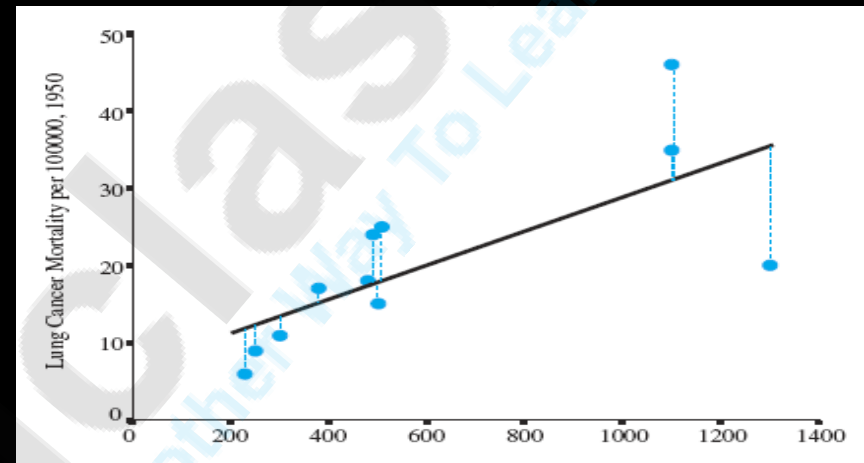


FIGURE 15.1 Three-dimensional response plane.

Simple Regression Model

Regression coefficients are estimated by minimizing $\sum \text{residuals}^2$ (i.e., sum of the squared residuals) to derive this model:

$$\hat{y} = a + bx$$

The standard error of the regression ($s_{Y|x}$) is based on the squared residuals:

$$s_{Y|x} = \sqrt{\sum \text{residuals}^2 / df_{\text{res}}}$$

Multiple Regression Model

Again, estimates for the *multiple* slope coefficients are derived by minimizing $\sum \text{residuals}^2$ to derive this multiple regression model:

$$\hat{y} = a + b_1x_1 + b_2x_2$$

Again, the standard error of the regression is based on the $\sum \text{residuals}^2$:

$$S_{Y|x} = \sqrt{\sum \text{residuals}^2 / df_{\text{res}}}$$

Multiple Regression Model

Intercept α predicts where the regression *plane* crosses the Y axis

Slope for variable X_1 (β_1) predicts the change in Y per unit X_1 holding X_2 constant

The slope for variable X_2 (β_2) predicts the change in Y per unit X_2 holding X_1 constant

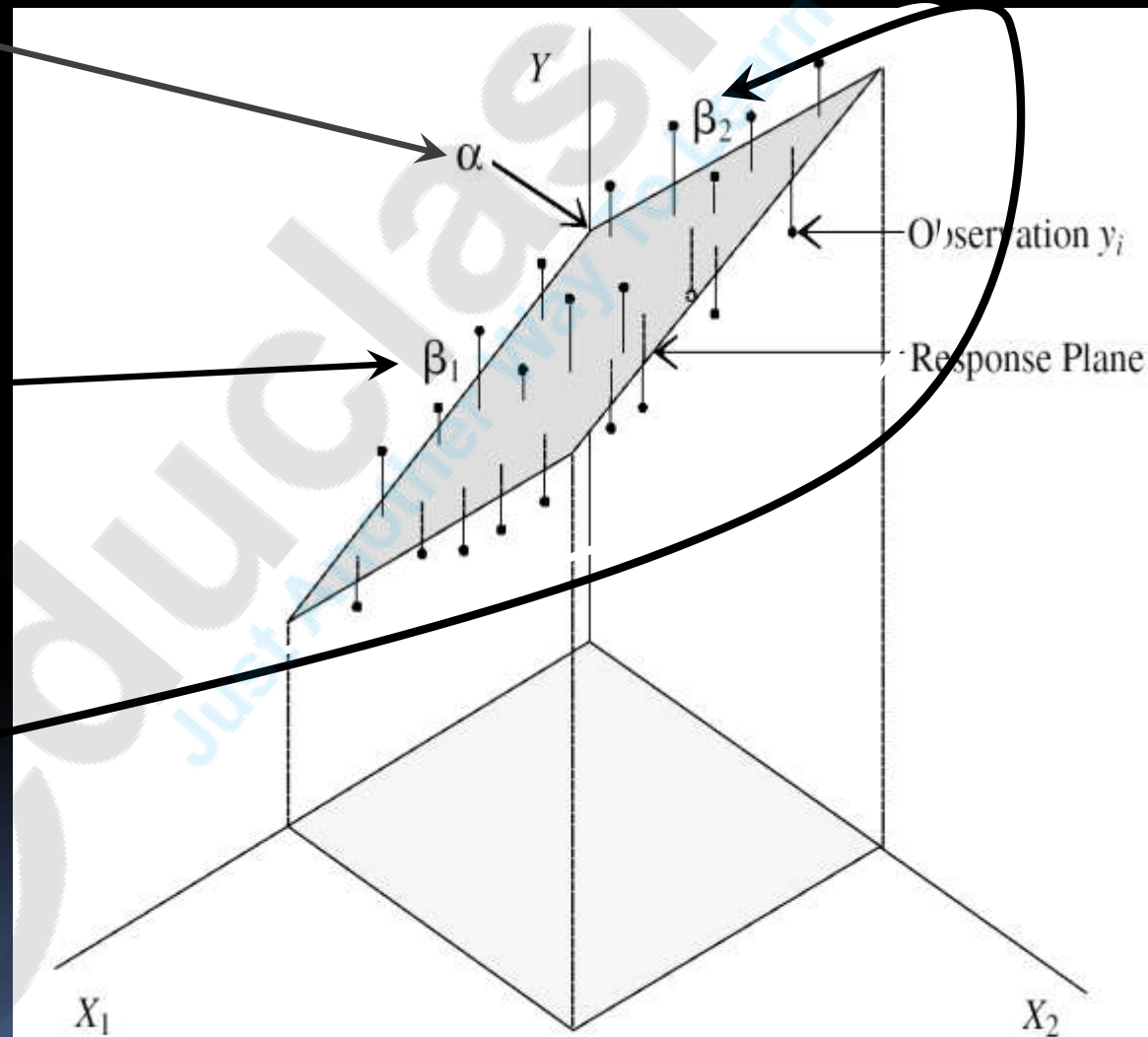


FIGURE 15.1 Three-dimensional response plane.

Multiple Regression Model

A multiple regression model with k independent variables fits a regression "surface" in $k + 1$ dimensional space (cannot be visualized)



Multiple regression

- Method for analysing a linear relationship involving more than two variables
 - $x_1, x_2, x_3, \dots, x_n$
 - $Y = a + b_1 x_1 + b_2 x_2 + b_3 x_3 + \dots + b_n x_n$

Height of mother	Height of father	Height of daughter
63	64	58.6
67	65	64.7
64	67	66.3
...

Daughter Ht = $7.5 + 0.707 \text{mother} + 0.614 \text{father}$

Non linear regression

- The difference between linear and nonlinear regression models isn't as straightforward as it sounds.
- You'd think that linear equations produce straight lines and nonlinear equations model curvature.
- Unfortunately, that's *not* correct
- Both types of models can fit curves to your data—so that's not the defining characteristic

Non linear regression

- Linear model $\rightarrow y=a+bx$
 - Multiple linear regression $\rightarrow y=a+bx_1+cx_2$
 - $\rightarrow y=a+bx+cx^2$
 - if you take derivative with respect to any **parameter** the resultant is **1(constant)**.
 - $Y= a+bx \rightarrow dy/da = 1 \quad dy/db=1x$
 - $y=a+bx+cx^2 \rightarrow dy/da= 1 \quad dy/db= 1x$
 - A regression model is called **non linear** if the derivative of the model depends on one or more parameters.
 - $Y= a+b^2x \rightarrow dy/db=2bx$ i.e.the derivative is dependent on 'b'
- Non linear by parameter and not ,non linear by independent variable**

Logistic regression

- Used when the dependent variable is binary
 - 0/1, T/F, Y/N
- $Y = a + f_1(x_1) + \dots + f_n(x_n)$
- f_1 is the function being used to transform the predictor

Just Another Way To Learn

Logistic regression

- Uses a logistic curve

$$p = \frac{e^{(a+bx)}}{1 + e^{(a+bx)}}$$

dp/db will depend on b, thus logistic regression is nonlinear regression

- Logistic curve gives a value between 0 and 1 so it can be interpreted as the probability of class membership
- $\text{Log}_e (p/(1-p)) = a+bx$
- Dependent variable, $Y = a+bx$
- p is the probability of being in the class
- $(1-p)$ is the probability that it is not



Thanks



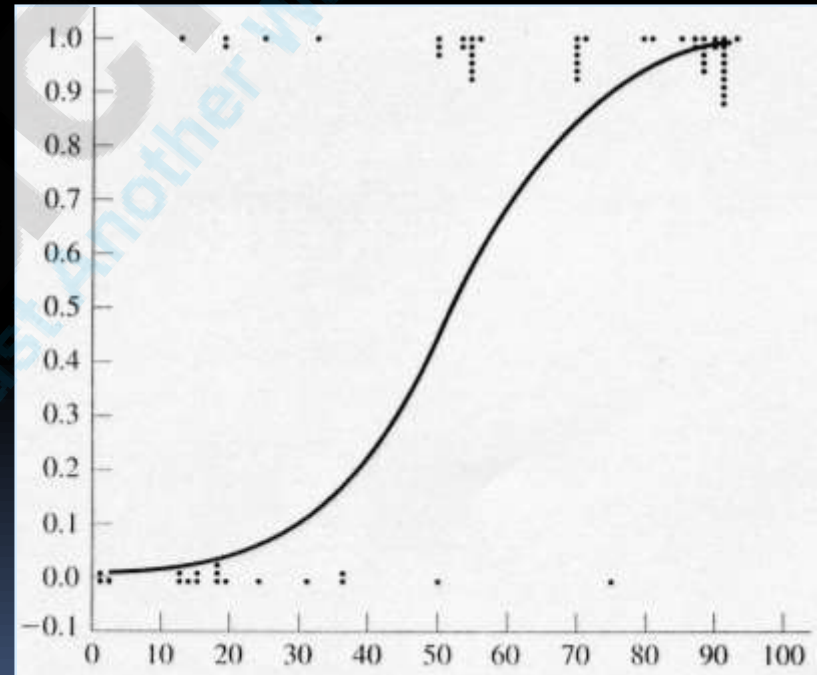
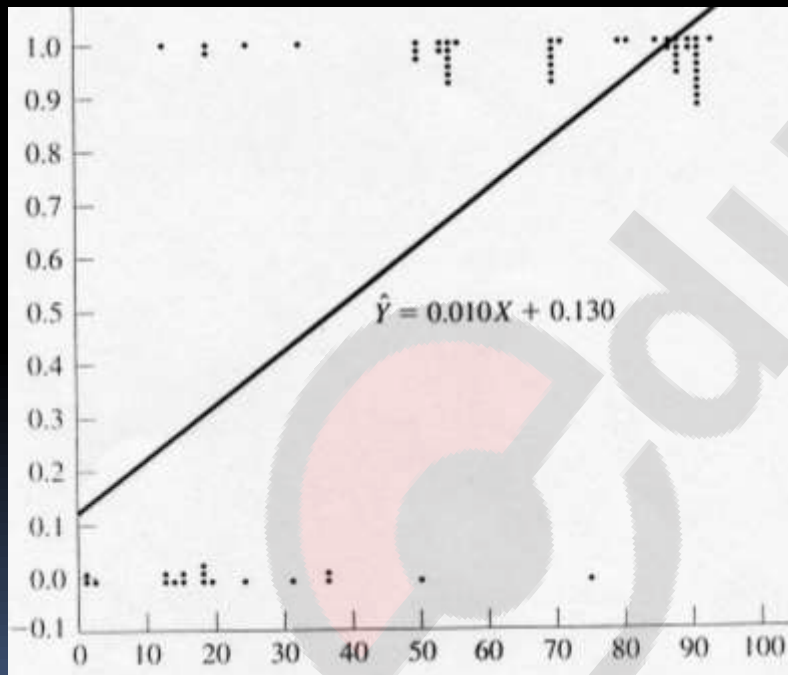
Udacity
Just Another Way To Learn

Logistic Regression

- One commonly used algorithm is Logistic Regression
- Assumes that the dependent (output) variable is binary which is often the case in medical and other studies. (Does person have disease or not, survive or not, accepted or not, etc.)
- Like Quadric, Logistic Regression does a particular non-linear transform on the data after which it just does linear regression on the transformed data
- Logistic regression fits the data with a sigmoidal/logistic curve rather than a line and outputs an approximation of the probability of the output given the input

Logistic Regression Example

- Age (X axis, input variable) – Data is fictional
- Heart Failure (Y axis, 1 or 0, output variable)
- Could use value of regression line as a probability approximation
 - Extrapolates outside 0-1 and not as good empirically
- Sigmoidal curve to the right gives empirically good probability approximation and is bounded between 0 and 1



Logistic Regression Approach

Learning

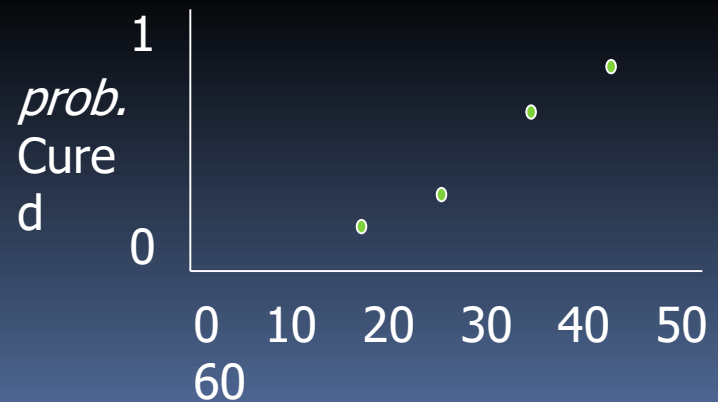
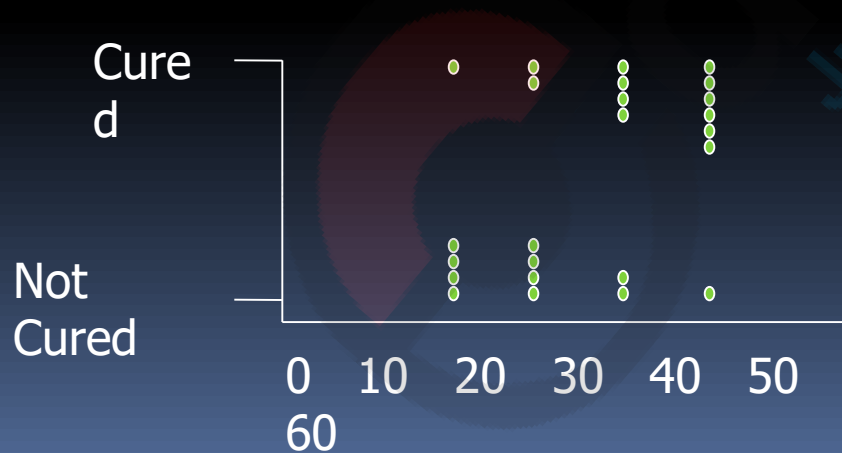
1. Transform initial input probabilities into log odds (logit)
2. Do a standard linear regression on the logit values
 - This effectively fits a logistic curve to the data, while still just doing a linear regression with the transformed input (ala quadric machine, etc.)

Generalization

1. Find the value for the new input on the logit line
2. Transform that logit value back into a probability

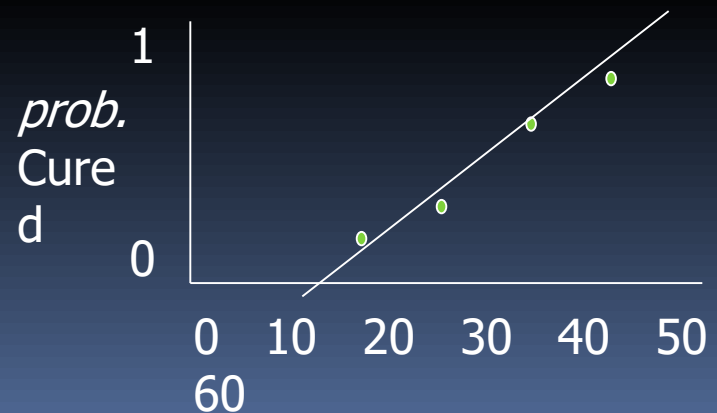
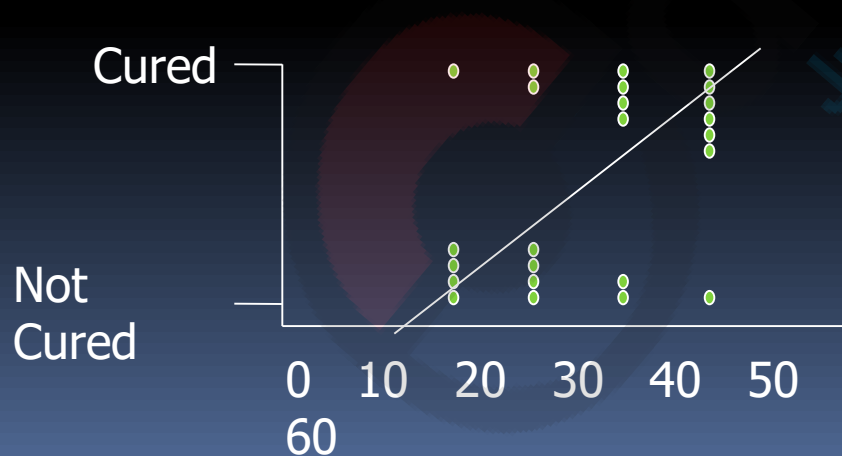
Non-Linear Pre-Process to Logit (Log Odds)

Medication Dosage	# Cured	Total Patients	Probability: # Cured/Total Patients
20	1	5	.20
30	2	6	.33
40	4	6	.67
50	6	7	.86



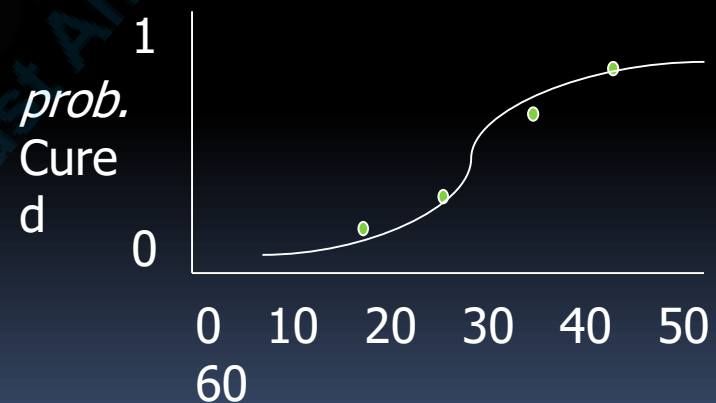
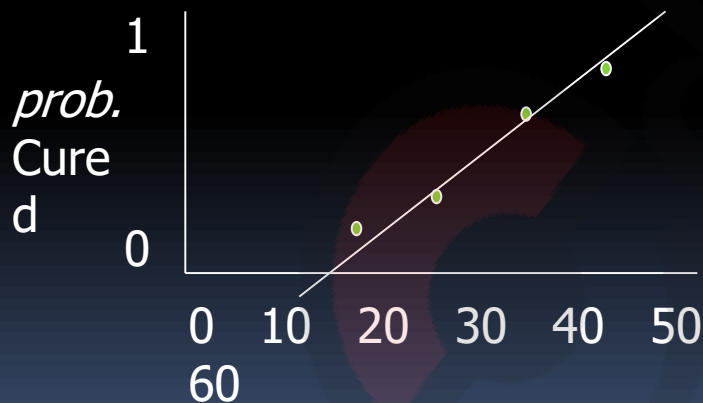
Non-Linear Pre-Process to Logit (Log Odds)

Medication Dosage	# Cured	Total Patients	Probability: # Cured/Total Patients
20	1	5	.20
30	2	6	.33
40	4	6	.67
50	6	7	.86



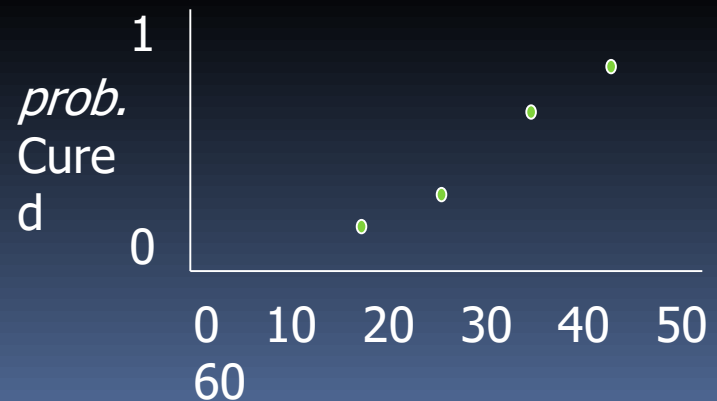
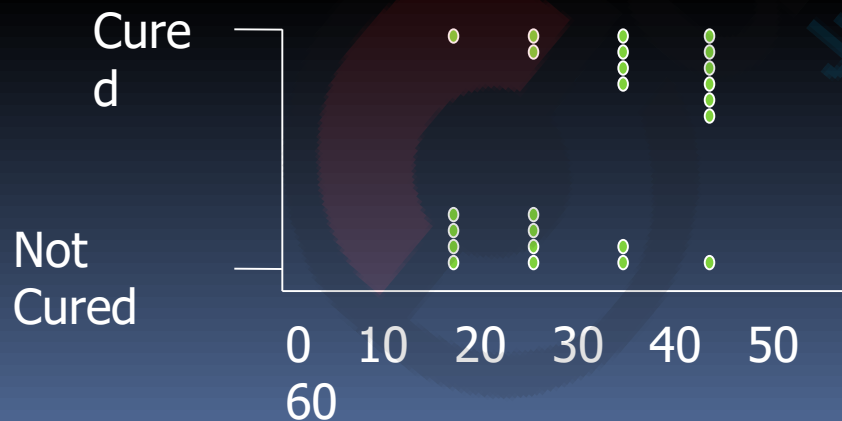
Logistic Regression Approach

- Could use linear regression with the probability points, but that would not extrapolate well
- Logistic version is better but how do we get it?
- Similar to Quadric we do a non-linear pre-process of the input and then do linear regression on the transformed values – do a linear regression on the log odds - Logit



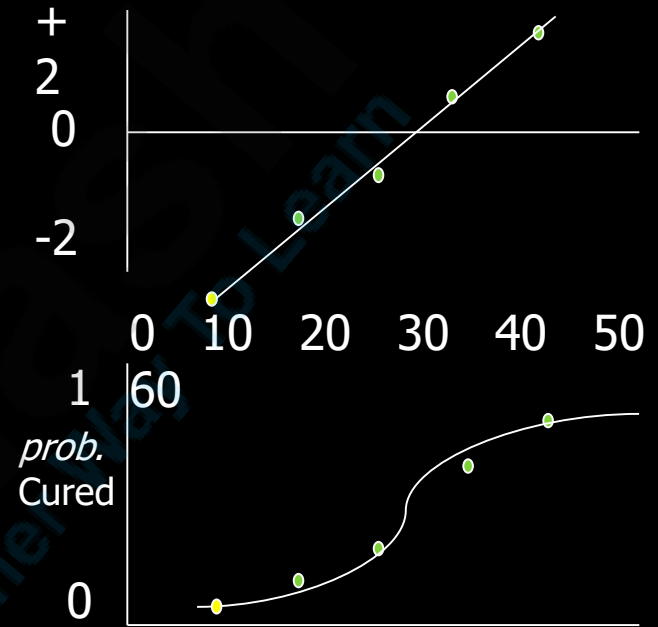
Non-Linear Pre-Process to Logit (Log Odds)

Medication Dosage	# Cured	Total Patients	Probability: # Cured/Total Patients	Odds: $p/(1-p) = \# \text{ cured} / \# \text{ not cured}$	Logit Log Odds: $\ln(\text{Odds})$
20	1	5	.20	.25	-1.39
30	2	6	.33	.50	-0.69
40	4	6	.67	2.0	0.69
50	6	7	.86	6.0	1.79



Regression of Log Odds

Medication Dosage	# Cured	Total Patients	Probability: # Cured/Total Patients	Odds: $p/(1-p) = \# \text{ cured} / \# \text{ not cured}$	Log Odds: $\ln(\text{Odds})$
20	1	5	.20	.25	-1.39
30	2	6	.33	.50	-0.69
40	4	6	.67	2.0	0.69
50	6	7	.86	6.0	1.79



- $y = .11x - 3.8$ - Logit regression equation
- Now we have a regression line for log odds (logit)
- To generalize, we interpolate the log odds value for the new data point
- Then we transform that log odds point to a probability: $p = e^{\text{logit}(x)} / (1 + e^{\text{logit}(x)})$
- For example assume we want p for dosage = 10
 - $\text{Logit}(10) = .11(10) - 3.8 = -2.7$
 - $p(10) = e^{-2.7} / (1 + e^{-2.7}) = .06$ [note that we just work backwards from logit to p]
- These p values make up the sigmoidal regression curve (which we never have to actually plot)

Heart rate	50	50	50	50	70	70	90	90	90	90	90
Heart Attack	y	n	n	n	n	y	y	y	n	y	y