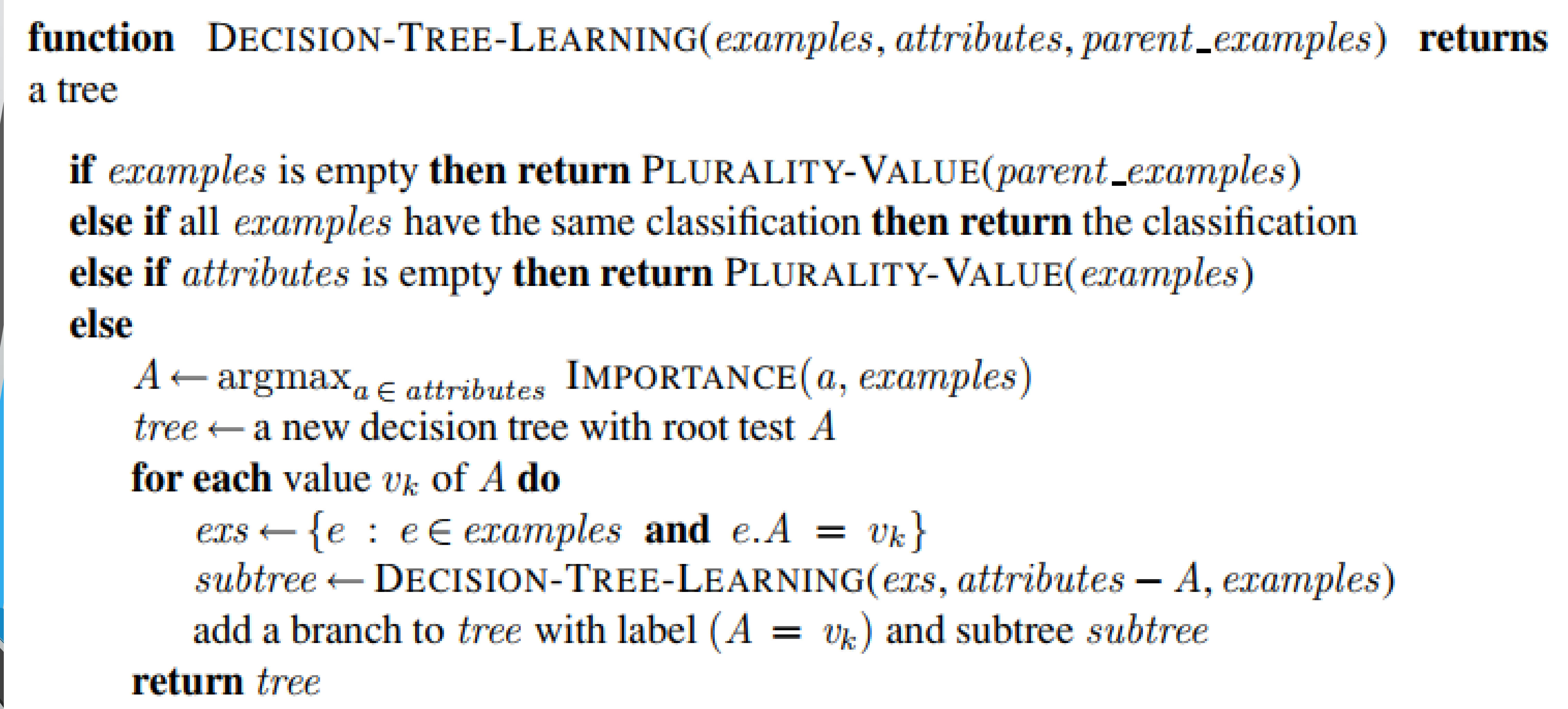
INSTITUT TEKNOLOGI BANDUNG
1920

Decision Tree Learning (Russel & Norvig, 2021)



```
function DECISION-TREE-LEARNING($examples, attributes, parent\_examples$) returns a tree

  if $examples$ is empty then return PLURALITY-VALUE($parent\_examples$)
  else if all $examples$ have the same classification then return the classification
  else if $attributes$ is empty then return PLURALITY-VALUE($examples$)
  else
      $A \leftarrow \text{argmax}_{a \in attributes}$ IMPORTANCE($a, examples$)
      $tree \leftarrow$ a new decision tree with root test $A$
      for each value $v_k$ of $A$ do
          $exs \leftarrow \{e : e \in examples$ and $e.A = v_k\}$
          $subtree \leftarrow$ DECISION-TREE-LEARNING($exs, attributes - A, examples$)
          add a branch to $tree$ with label $(A = v_k)$ and subtree $subtree$
      return $tree$
```

The function PLURALITY-VALUE selects the most common output value among a set of examples, breaking ties randomly.

Issues in DTL

Overfitting training data

Continuous -valued attribute

Handling attributes with differing costs

Handling missing attribute value

Alternative measures for selecting attributes

Modul : Issues in Decision Tree Learning (DTL)

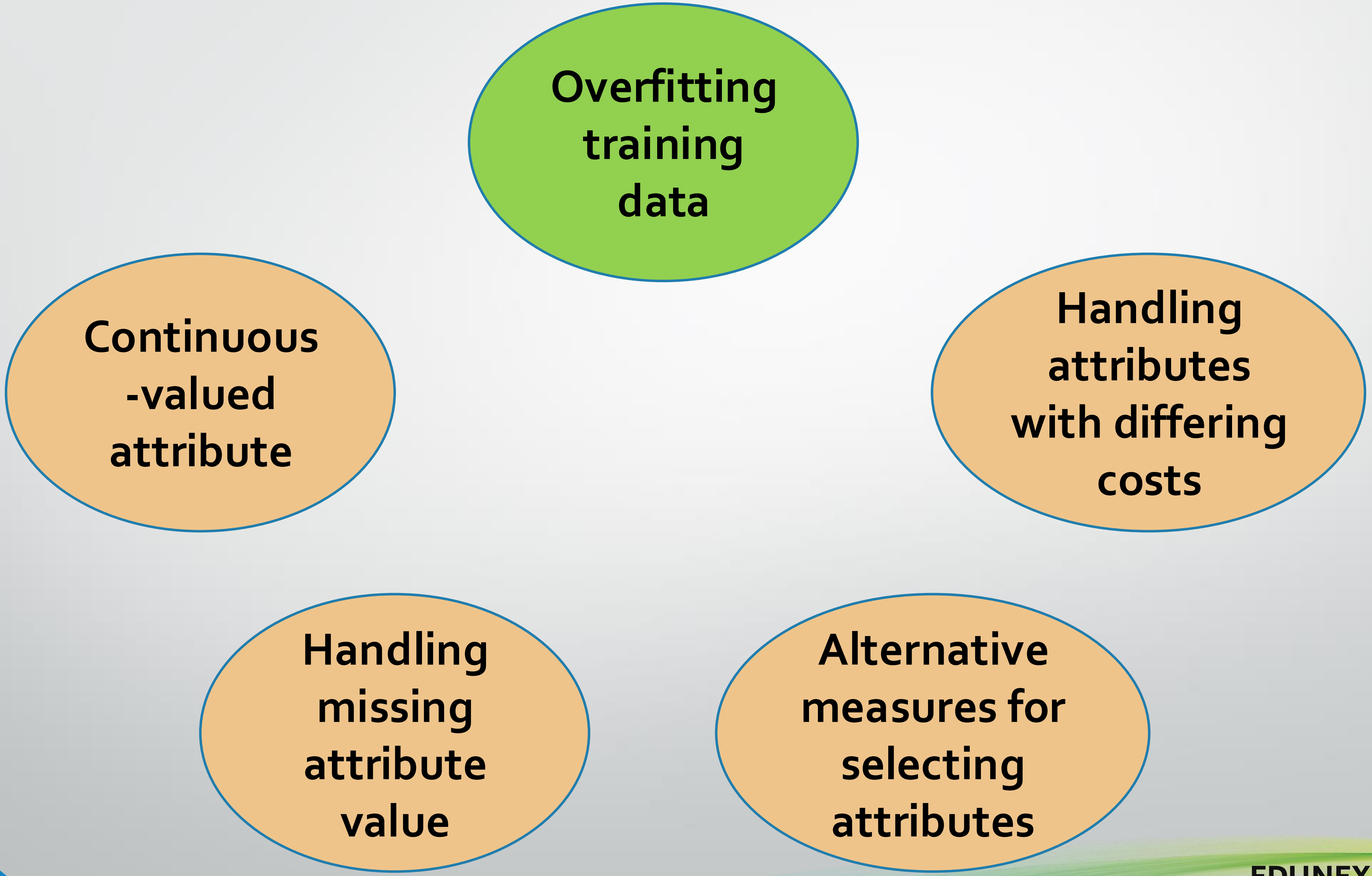
Overfitting

Nur ULFA Maulidevi

KK IF - Teknik Informatika- STEI ITB

Pembelajaran Mesin
(Machine Learning)

Issues in DTL



Overfitting training data

Continuous -valued attribute

Handling attributes with differing costs

Handling missing attribute value

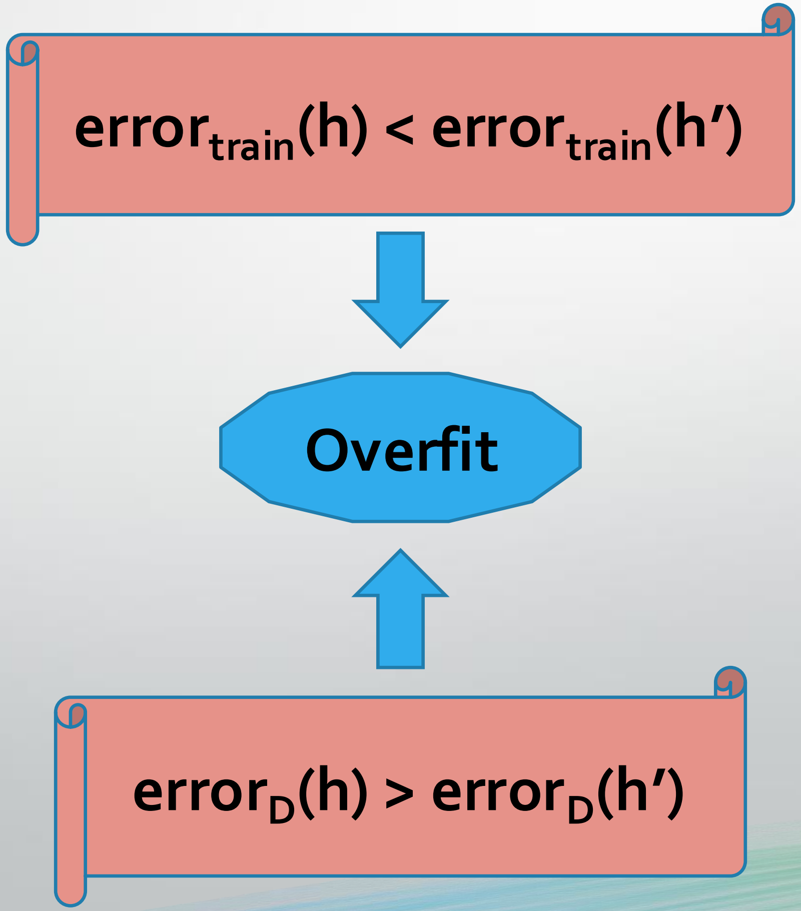
Alternative measures for selecting attributes

What is Overfit

H: Hypothesis space

A hypothesis: $h \in H$; Alternative hypothesis: $h' \in H$

train: training examples; D: entire distribution of data



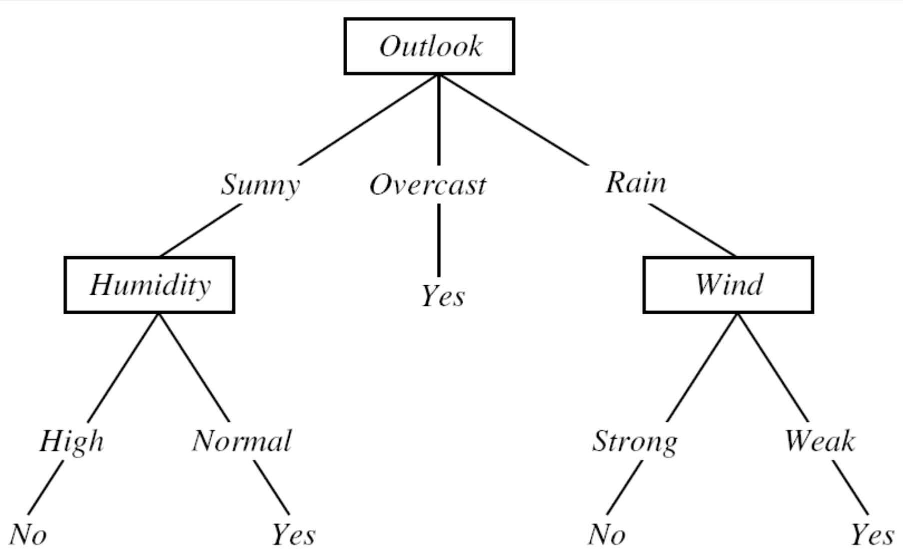
$\text{error}_{train}(h) < \text{error}_{train}(h')$

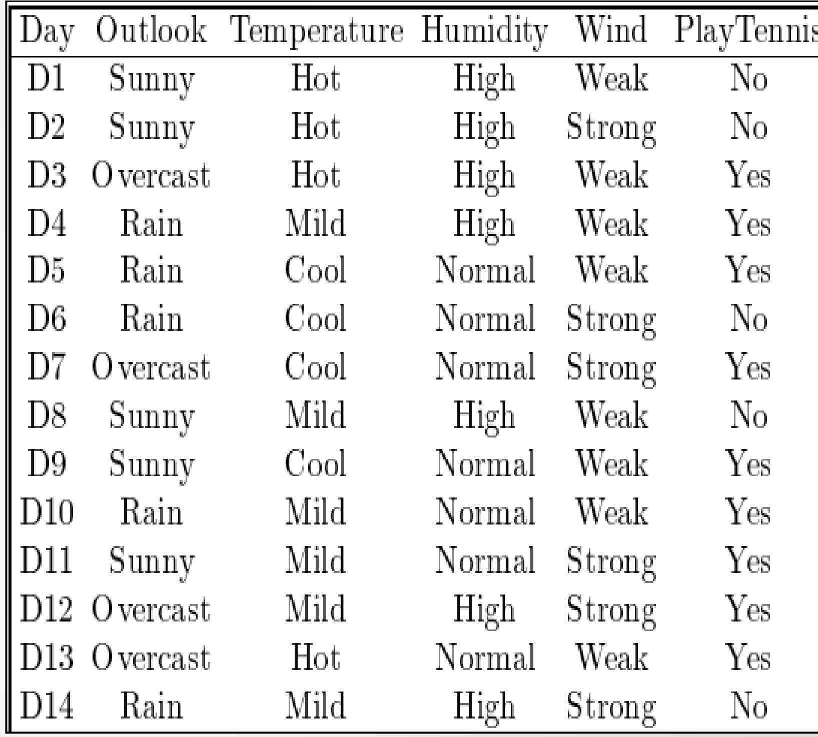
Overfit

$\text{error}_{D}(h) > \text{error}_{D}(h')$

Illustration

D15 (noisy training examples):
Outlook = Sunny;
Temperature = Hot;
Humidity = Normal;
Wind = Strong;
PlayTennis = No





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

?

Overfitting can happen even training examples is noise-free (when small numbers of examples are associated with leaf Nodes) → decrese accuracy 10 – 25% on most problems

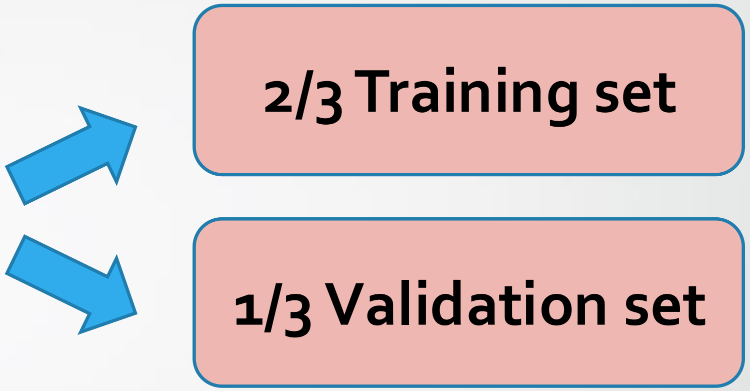
Solution Approaches

	Pros	Cons
1. Stop growing the tree earlier, before it reaches the point where it perfectly classifies the training data	More Direct	Difficulty of estimating precisely when to stop growing the tree
2. allow the tree to overfit the data, and then post-prune the tree	More Successfull in practice	Requires more steps (grow until fit, then prune)

what criterion is to be used to determine the correct final tree size

Approaches in Determine the Correct Final Tree Size

1. Use separate examples distinct from training to evaluate the pruning tree



2/3 Training set

1/3 Validation set

2. Use all available data for training, then conduct a statistical test to check whether expanding (or pruning) a node will produce improvement , example: chi square test

3. Use explicit measure of the complexity for encoding the training examples and decision tree → Minimum Description Length Principle

Source:
Machine Learning by Tom Mitchell chapter 6.6

$$h_{MDL} = \operatorname*{argmin}_{h \in H} L_{C_1}(h) + L_{C_2}(D|h)$$

$L_{C_1}(h): Length\ (number\ of\ bits)\ of\ hypothesis\ encoding$

$L_{C_2}(D|h): Length\ of\ data\ D\ given\ hypothesis\ h\ encoding$

Reduced Error Pruning

Consider decision (attribute) node as candidates for pruning → assign the most common classification affiliated with that node

Grow until fit then prune

Split data into training and validation set

Do until further pruning is harmful:

1. Evaluate impact on validation set of pruning each possible node (plus those below it)
2. Greedily remove the one (node) that most improves validation set accuracy

an effective approach provided a large amount of data is available

Rule Post-Pruning

Improvement of ID3 Algorithm: C4.5

Suitable for limited data

1. Growing the tree from training set, until the training data is fit as well as possible and allowing overfitting to occur.
2. Convert the learned tree into an equivalent set of rules by creating one rule for each path from the root node to a leaf node.
3. Prune (generalize) each rule by removing any preconditions that result in improving its estimated accuracy.
4. Sort the pruned rules by their estimated accuracy, and consider them in this sequence when classifying subsequent instances.

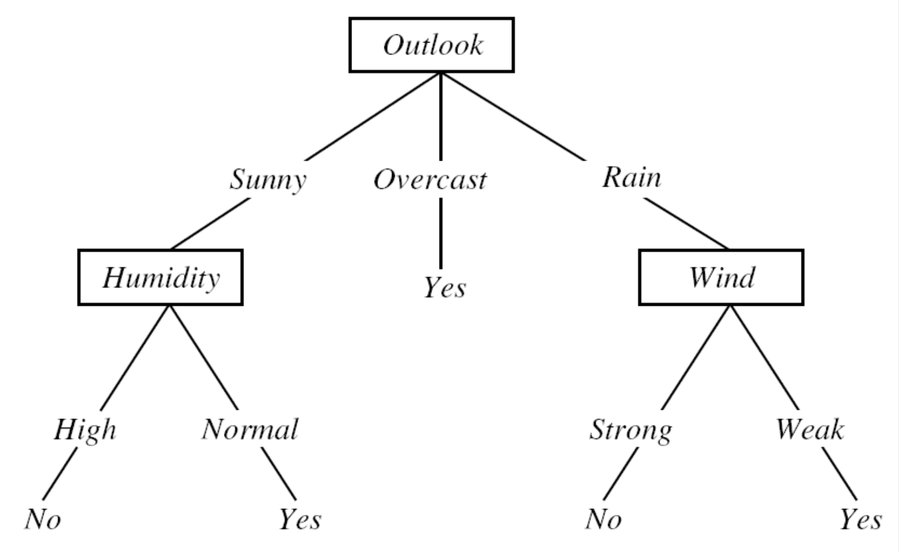
Example

Decision Rules:
If Outlook = Sunny and Humidity=High then No

Pruning:
If Outlook = Sunny then No
OR
If Humidity=High then No

Increase/ Reduce Accuracy?

Over validation set/ training set (C4.5)



Why Decision Tree → Decision Rule ?

1. Distinct path ~ distinct rule: independent pruning
2. No distinction between attribute tests
3. Improves readability

Modul : Decision Tree Learning (DTL)

Variable (Attribute) Types

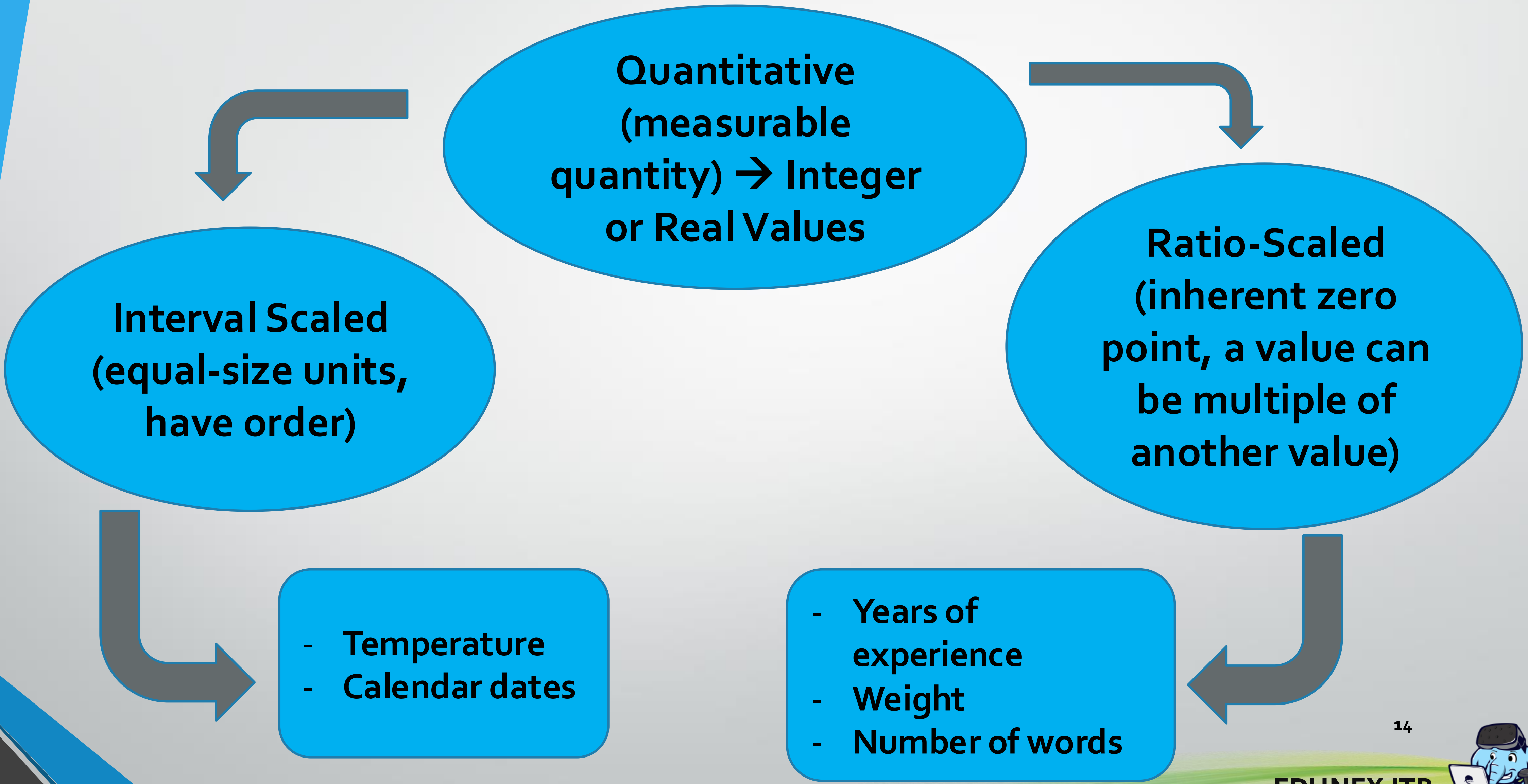
Source: DataMining Concepts and Techniques
by Jiawei Han, Micheline Kamber, Jian Pei

Nur ULFA Maulidevi

KK IF - Teknik Informatika- STEI ITB

Pembelajaran Mesin
(Machine Learning)

Numeric



Quantitative (measurable quantity) → Integer or Real Values

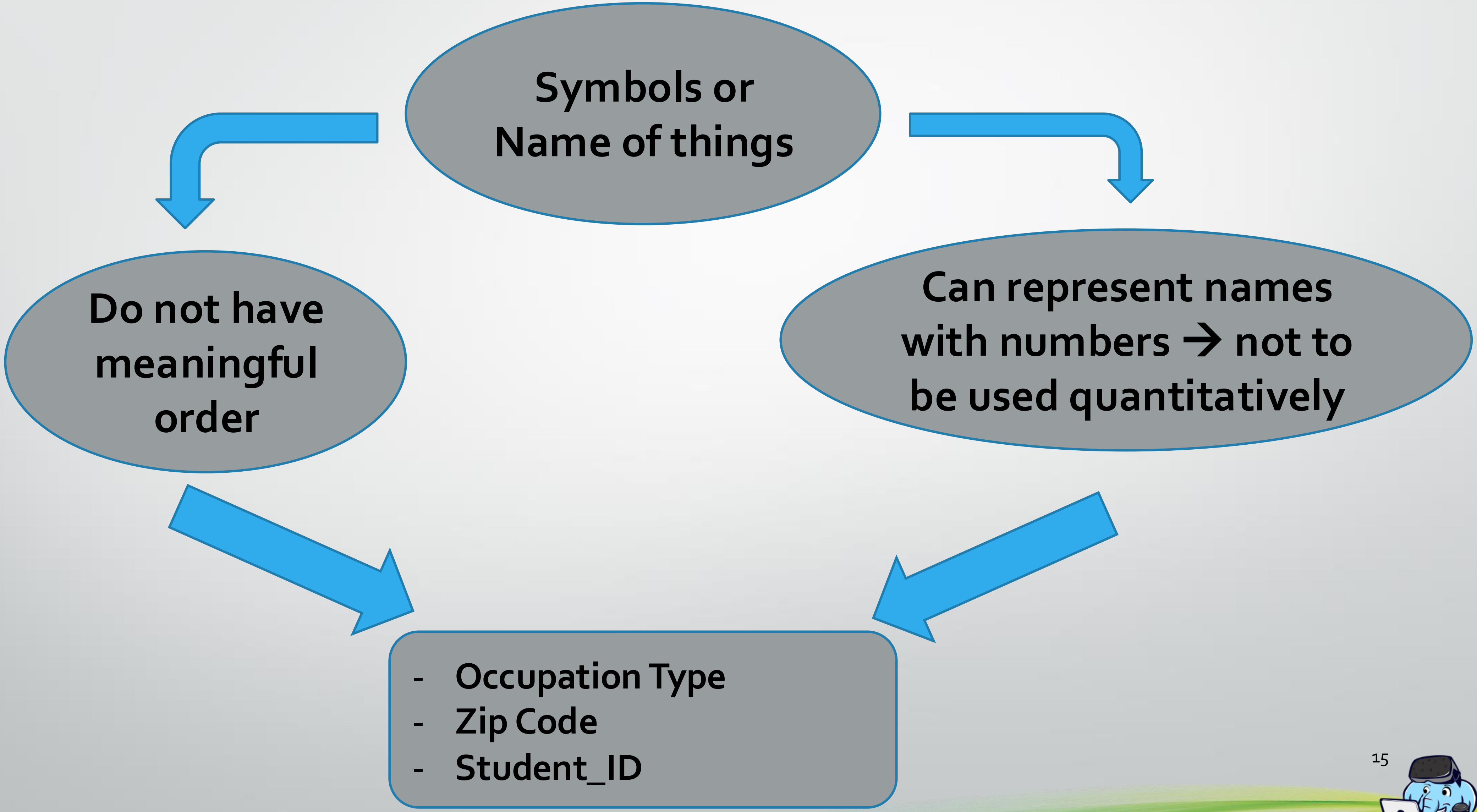
Interval Scaled (equal-size units, have order)

- Temperature
- Calendar dates

Ratio-Scaled (inherent zero point, a value can be multiple of another value)

- Years of experience
- Weight
- Number of words

Nominal/ Categorical



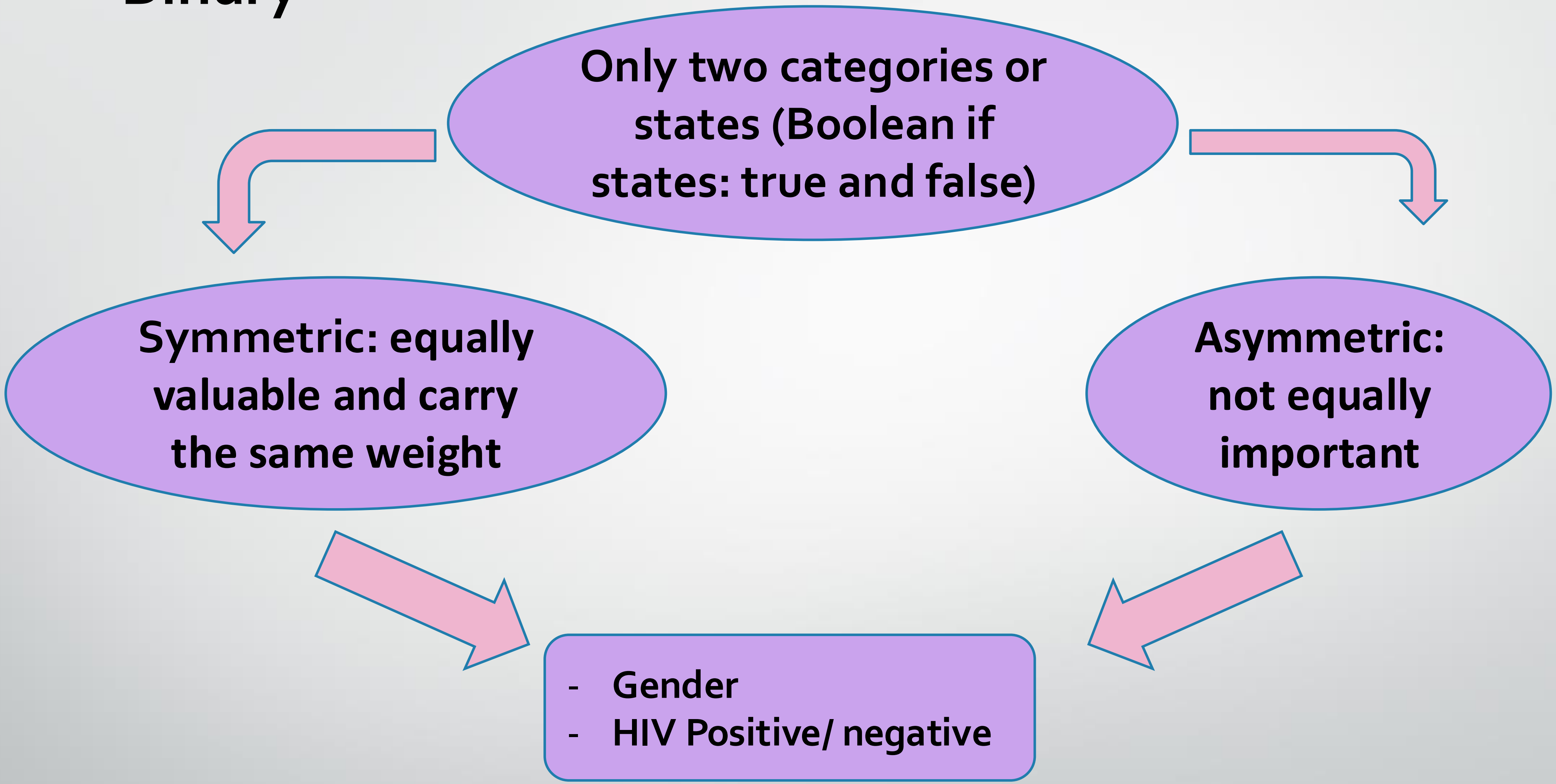
Symbols or Name of things

Do not have meaningful order

Can represent names with numbers → not to be used quantitatively

- Occupation Type
- Zip Code
- Student_ID

Binary

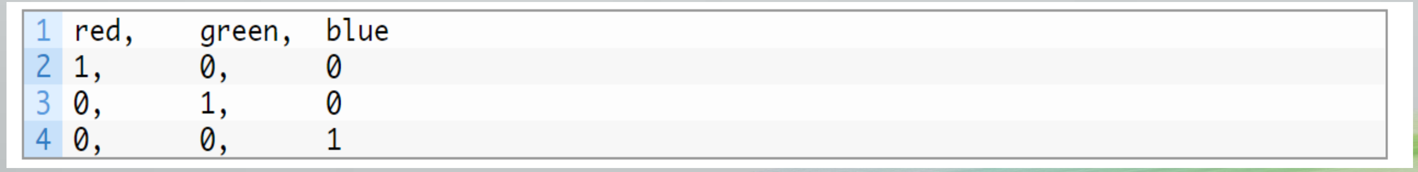


Only two categories or states (Boolean if states: true and false)

Symmetric: equally valuable and carry the same weight

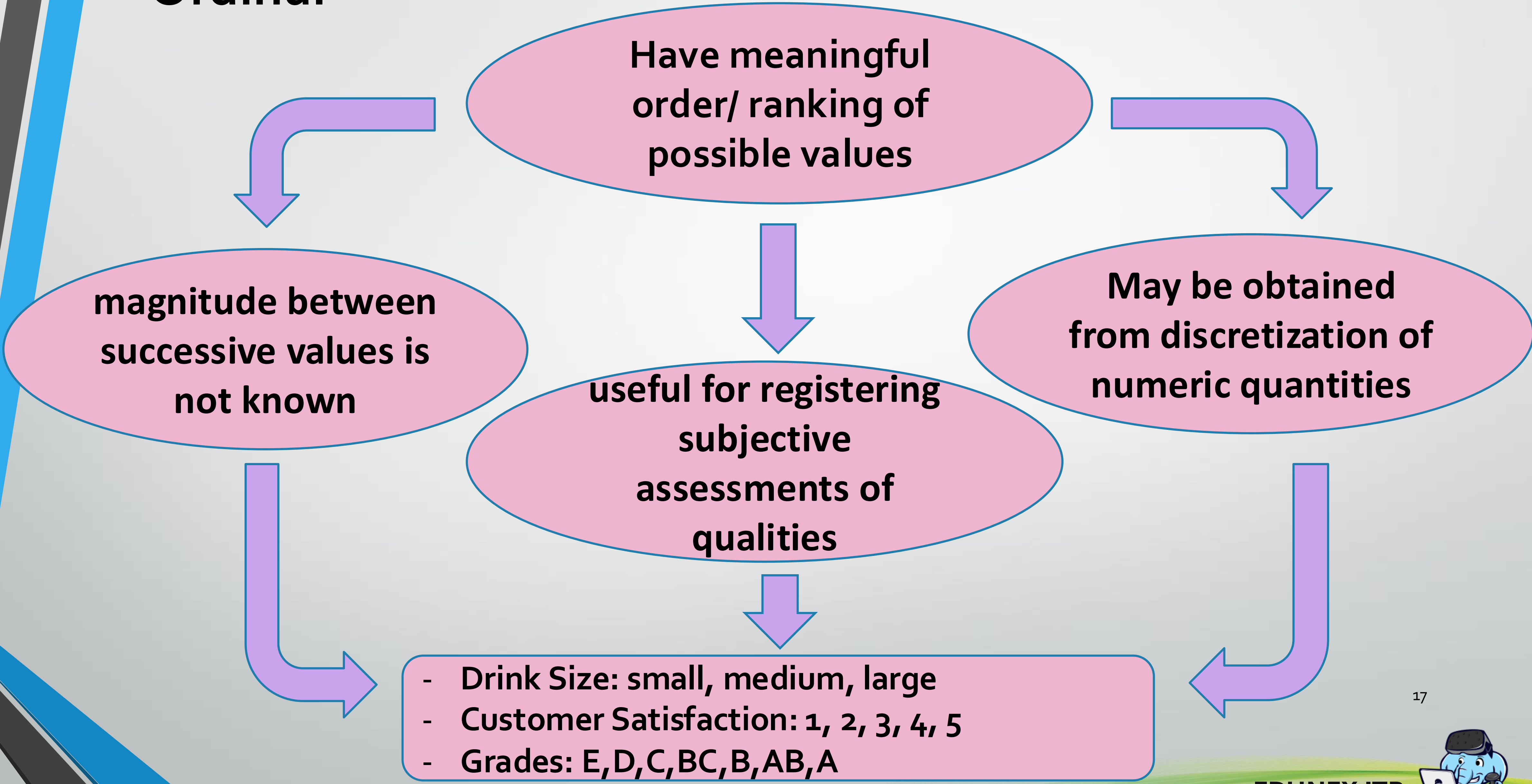
Asymmetric: not equally important

- Gender
- HIV Positive/ negative



```
red,    green,  blue
1,      0,      0
0,      1,      0
0,      0,      1
```

Ordinal



Have meaningful order/ ranking of possible values

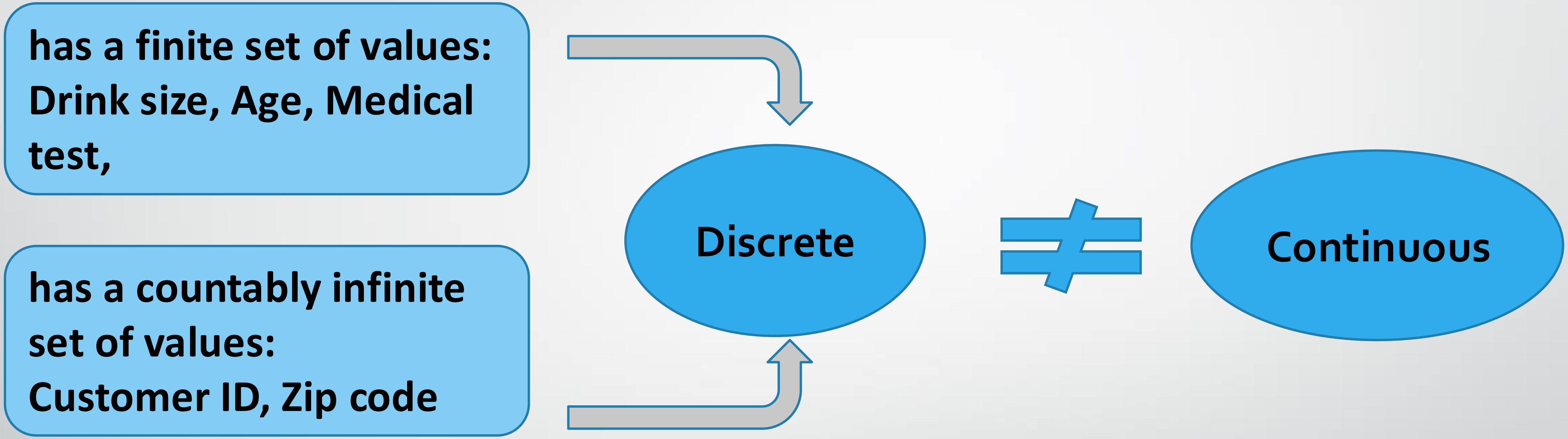
magnitude between successive values is not known

useful for registering subjective assessments of qualities

May be obtained from discretization of numeric quantities

- Drink Size: small, medium, large
- Customer Satisfaction: 1, 2, 3, 4, 5
- Grades: E,D,C,BC,B,AB,A

Discrete vs Continuous



has a finite set of values: Drink size, Age, Medical test,

has a countably infinite set of values: Customer ID, Zip code

Discrete ≠ Continuous

Modul : Issues in Decision Tree Learning (DTL)

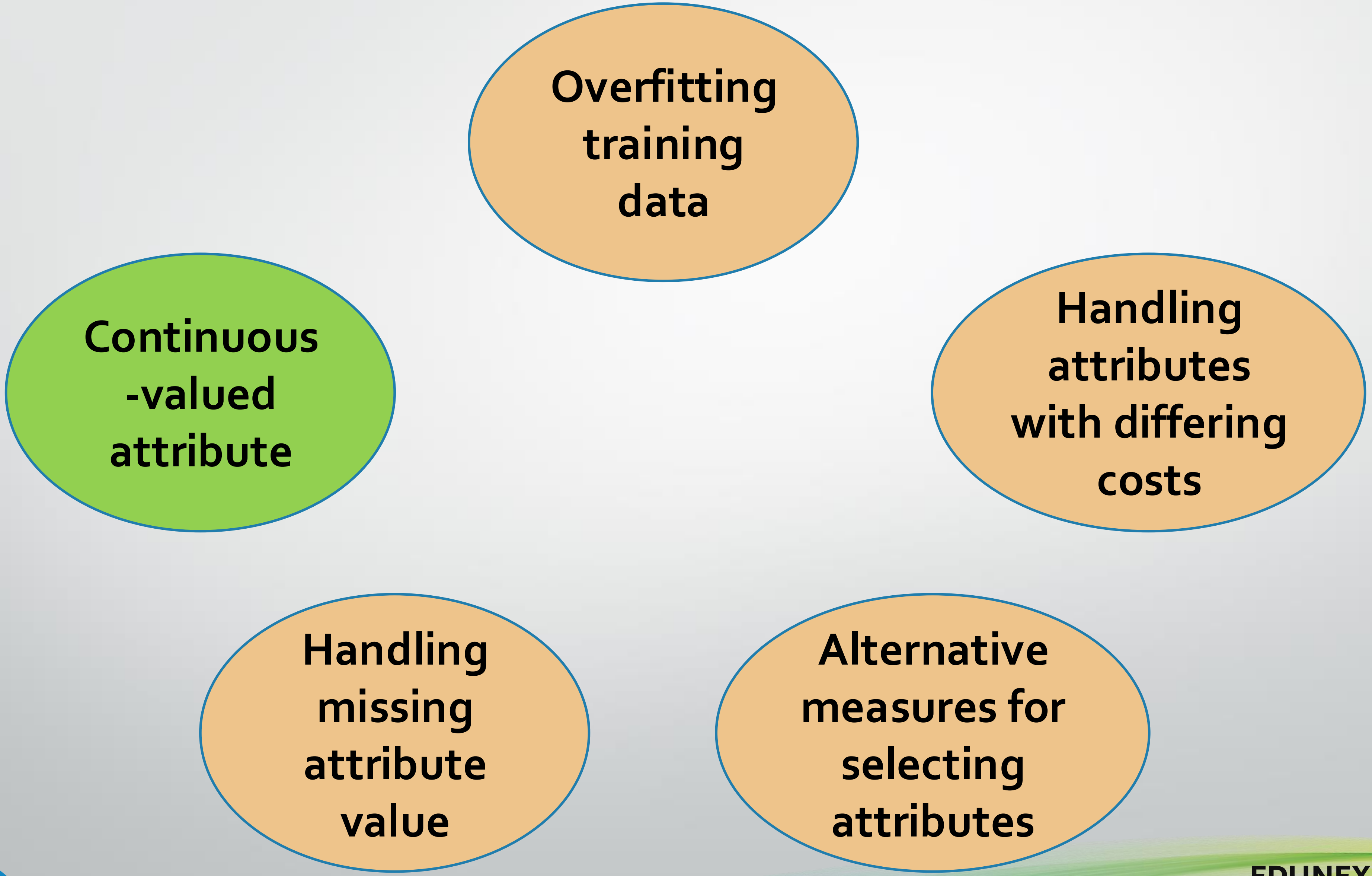
Continuous-valued Attribute

Nur ULFA Maulidevi

KK IF - Teknik Informatika- STEI ITB

Pembelajaran Mesin
(Machine Learning)

Issues in DTL



Overfitting training data

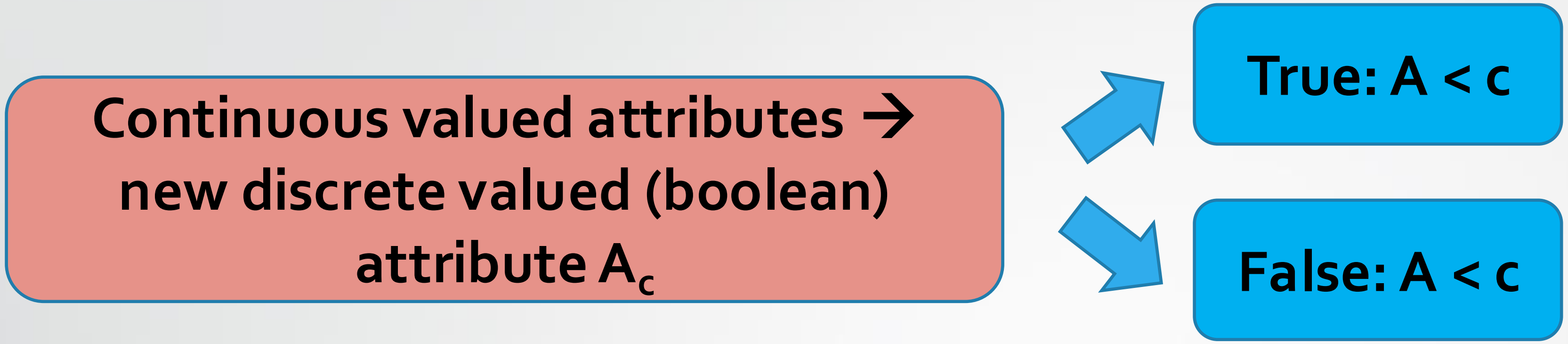
Continuous-valued attribute

Handling attributes with differing costs

Handling missing attribute value

Alternative measures for selecting attributes

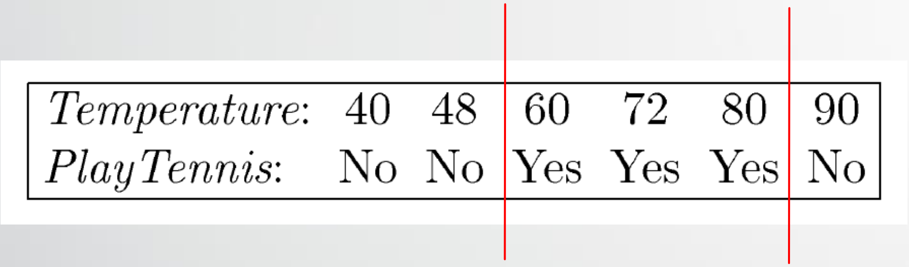
Discretization



Continuous valued attributes → new discrete valued (boolean) attribute A_c

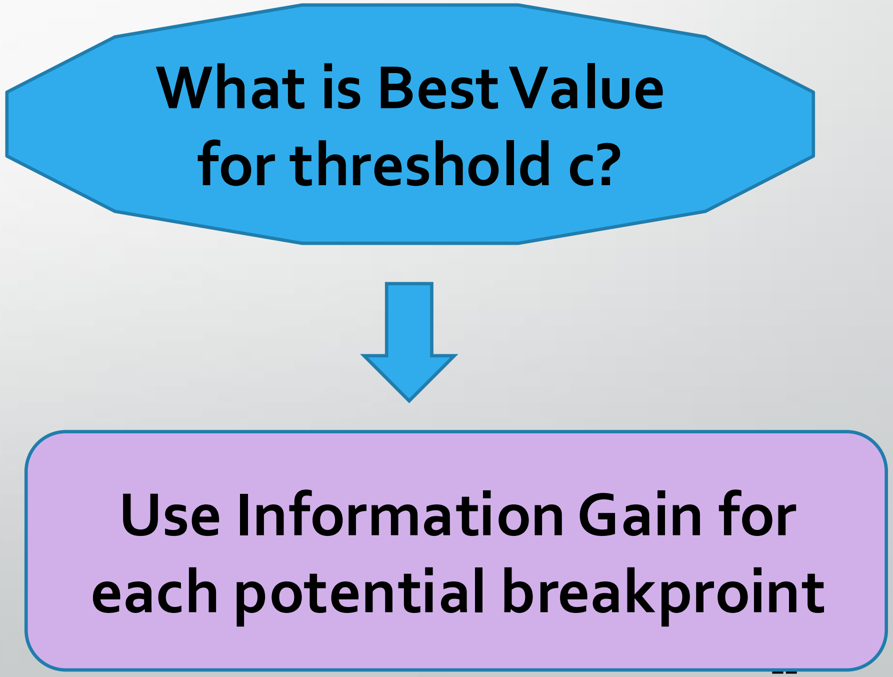
True: A < c

False: A < c



Temperature:	40	48	60	72	80	90
PlayTennis:	No	No	Yes	Yes	Yes	No

Potential optimal breakpoints



What is Best Value for threshold c?

Use Information Gain for each potential breakproint

$C = (48+60)/2 = 54$

Or

$C = (80+90)/2 = 85$

Illustration

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
D1		72			Yes
D2		40			No
D3		90			No
D4		60			Yes
D5		48			No
D6		80			Yes

1. Sort The Continuous-valued attribute

Temperature	40	48	60	72	80	90
Play Tennis	No	No	Yes	Yes	Yes	No

2. Identify Adjacent examples that differ in their target class

3. Candidates: midway between corresponding values → C : 54 or C : 85

For C: 54
Temperature < 54: 2 examples → yes/0, no/2
Temperature ≥ 54: 4 examples → yes/3, no/1
Gain(S,T_{54}) = Entropi(S) – [(2/6*Entropi(0,2))+(4/6*Entropi(3,1))]

4. Find the greatest Gain from the candidates, and other discrete-valued attributes

For C: 85
Temperature < 85: 5 examples → yes/3, no/2
Temperature ≥ 85: 1 examples → yes/0, no/1
Gain(S,T_{85}) = Entropi(S) – [(5/6*Entropi(3,2))+(1/6*Entropi(0,1))]

Modul : Issues in Decision Tree Learning (DTL)

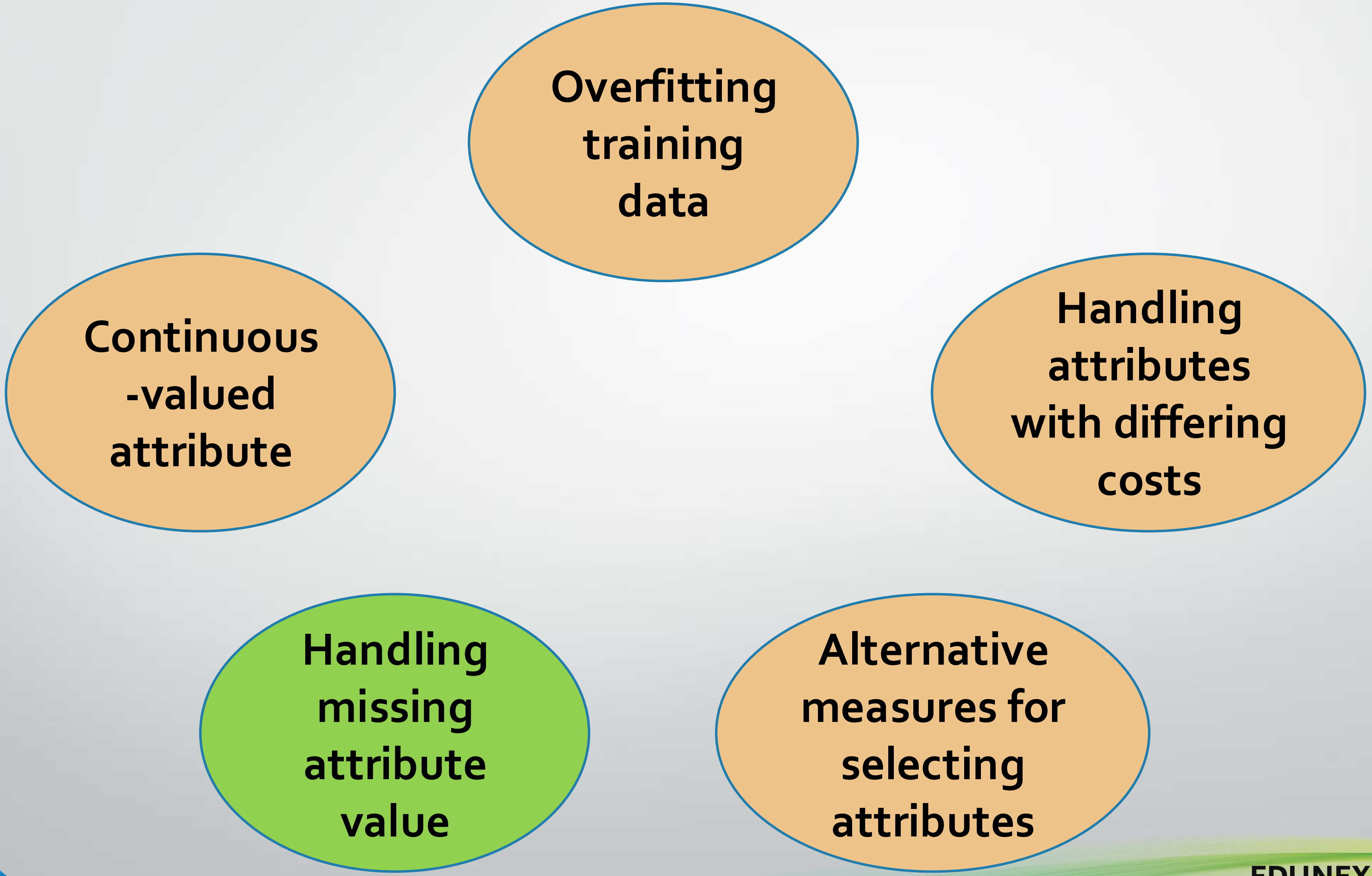
Missing Attribute Values

Nur ULFA Maulidevi

KK IF - Teknik Informatika- STEI ITB

Pembelajaran Mesin
(Machine Learning)

Issues in DTL



Overfitting training data

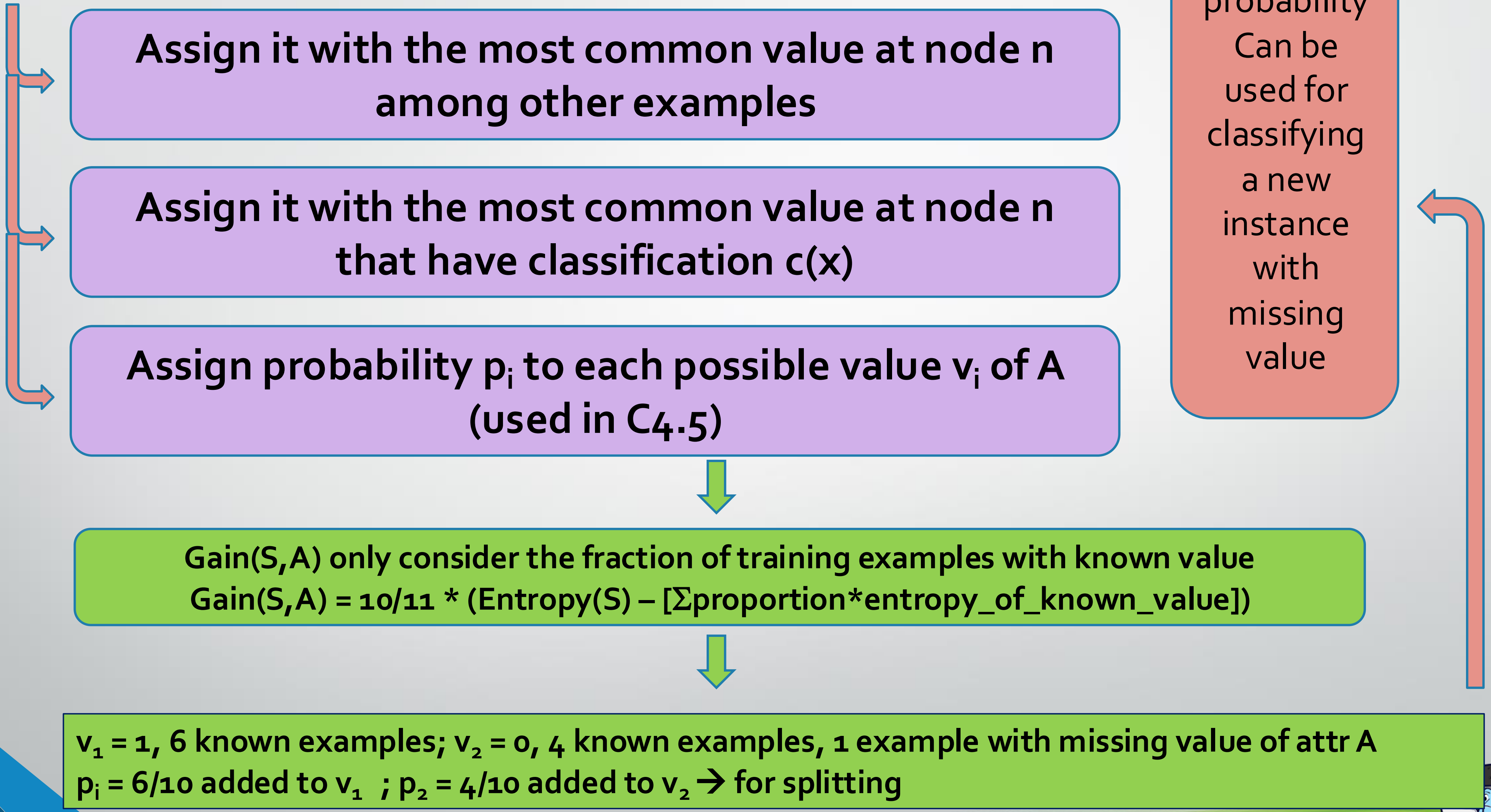
Continuous -valued attribute

Handling attributes with differing costs

Handling missing attribute value

Alternative measures for selecting attributes

Alternative Strategies



Assign it with the most common value at node n among other examples

Assign it with the most common value at node n that have classification c(x)

Assign probability p_i to each possible value v_i of A (used in C4.5)

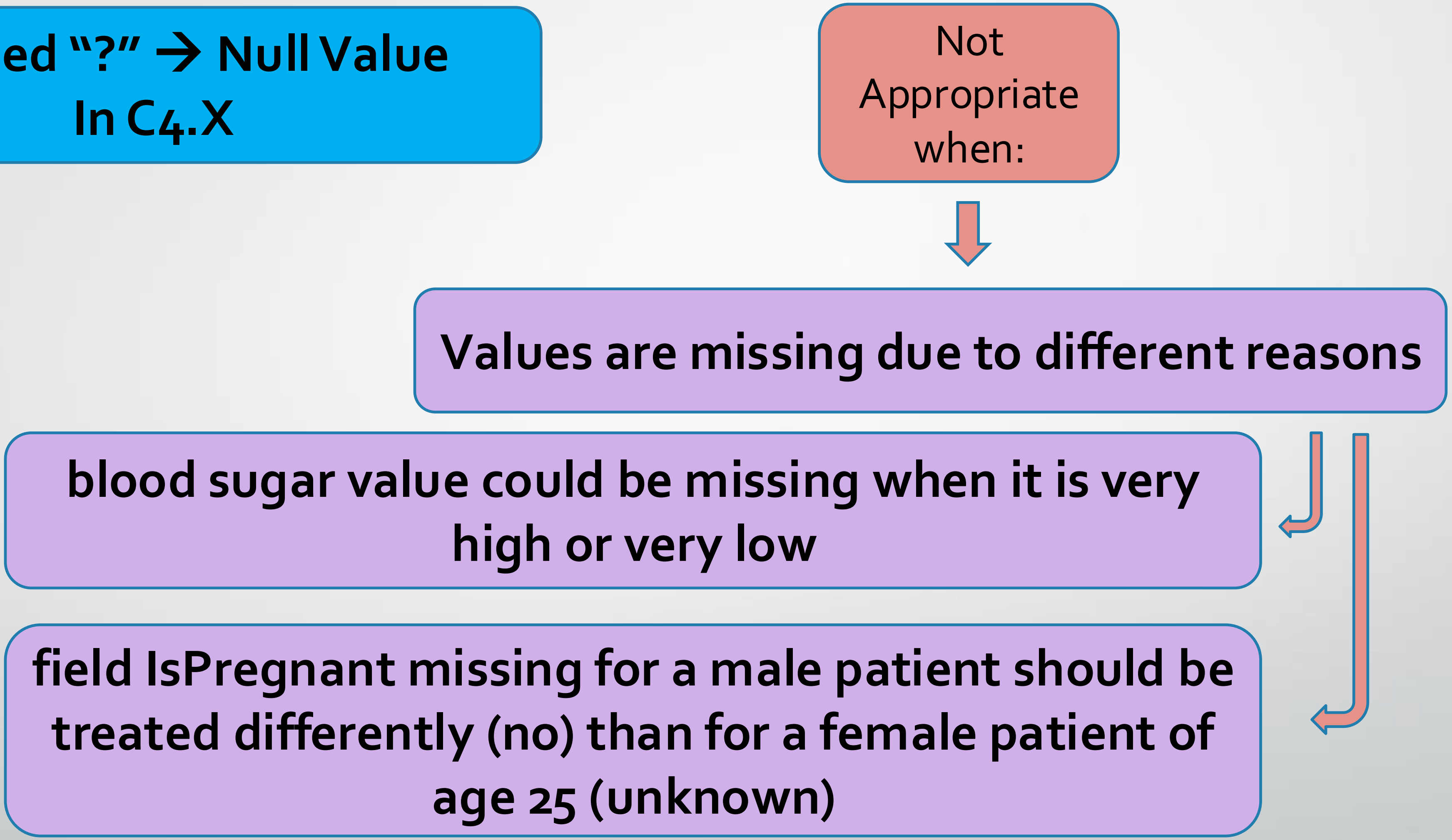
The probability Can be used for classifying a new instance with missing value

Gain(S,A) only consider the fraction of training examples with known value
Gain(S,A) = 10/11 * (Entropy(S) – [Σproportion*entropy_of_known_value])

v_1 = 1, 6 known examples; v_2 = 0, 4 known examples, 1 example with missing value of attr A
p_i = 6/10 added to v_1 ; p_2 = 4/10 added to v_2 → for splitting

Missing Value as Separate Value

Denoted "?" → Null Value
In C4.X



Not Appropriate when:

Values are missing due to different reasons

blood sugar value could be missing when it is very high or very low

field IsPregnant missing for a male patient should be treated differently (no) than for a female patient of age 25 (unknown)

Modul : Issues in Decision Tree Learning (DTL)

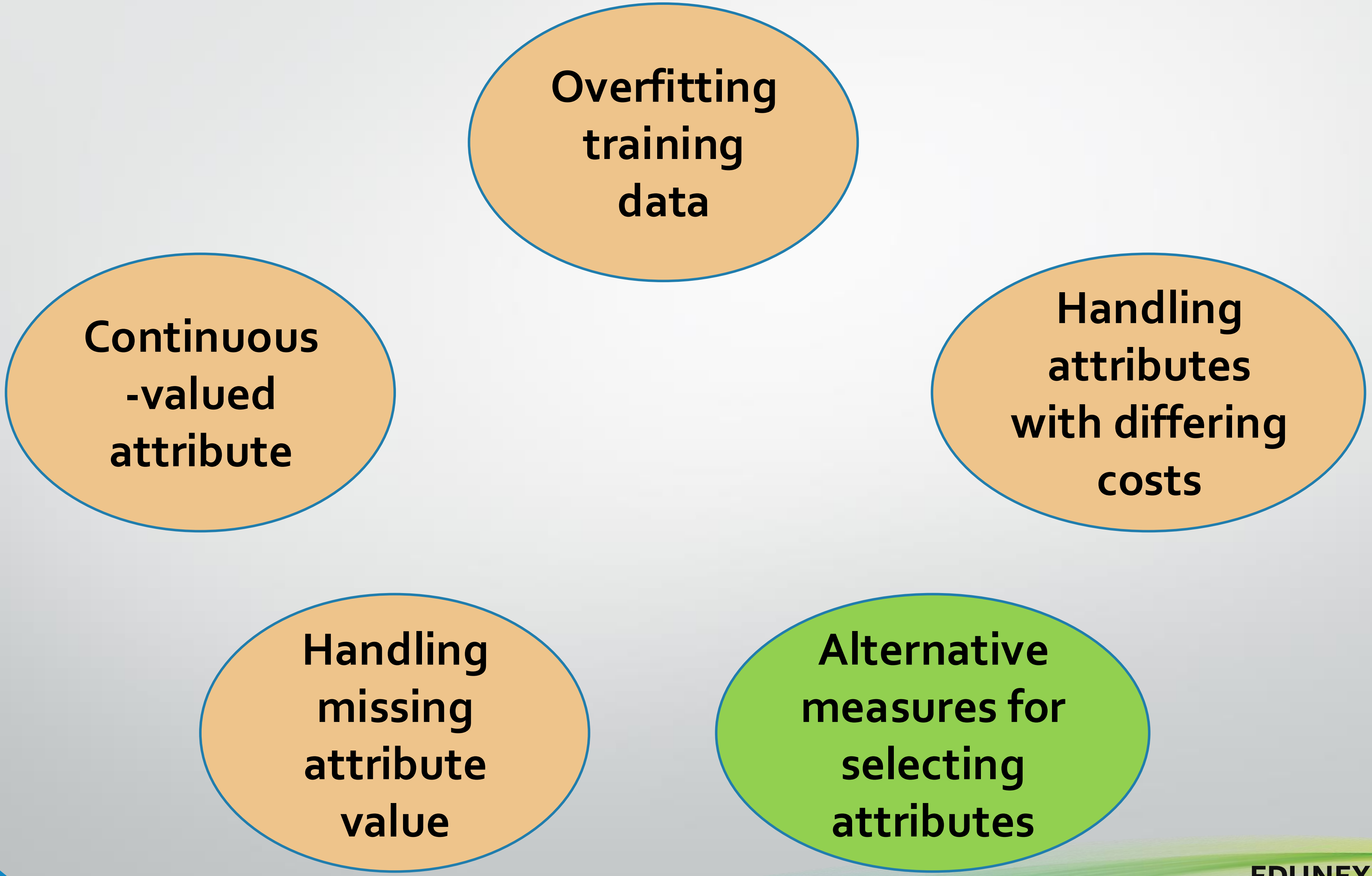
Alternative Measures for Selecting Attribute

Nur ULFA Maulidevi

KK IF - Teknik Informatika- STEI ITB

Pembelajaran Mesin
(Machine Learning)

Issues in DTL



Overfitting training data

Continuous -valued attribute

Handling attributes with differing costs

Handling missing attribute value

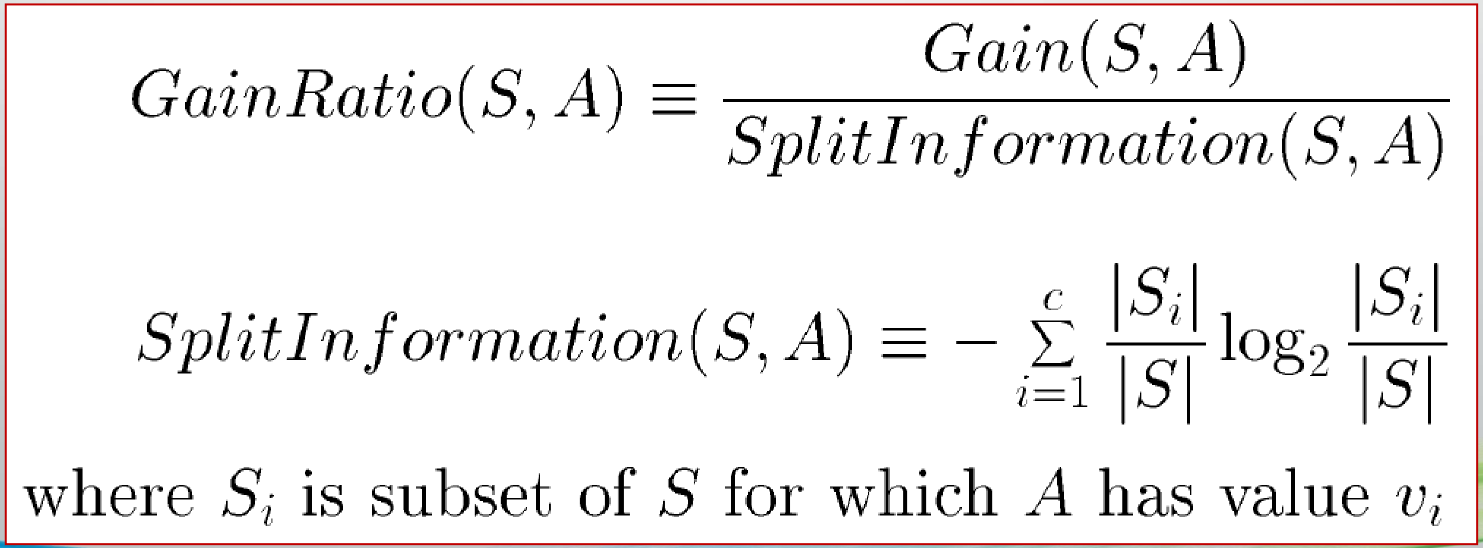
Alternative measures for selecting attributes

Attribute with many values (C4.5)

Gain will always select it → example *Date=2021_Jan_31*

Date will perfectly classify training examples, but very poor predictor for unseen data

GAIN RATIO



$$GainRatio(S, A) \equiv \frac{Gain(S, A)}{SplitInformation(S, A)}$$

$$SplitInformation(S, A) \equiv - \sum_{i=1}^{c} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

where S_i is subset of S for which A has value v_i

Illustration

Date	Atr2	Atr3	Class
2021_Jan_01	v1		No
2021_Jan_02	v1		No
2021_Jan_03	v2		Yes
2021_Jan_04	v2		Yes
2021_Jan_05	v1		Yes
2021_Jan_06	v1		No

$$\text{SplitInformation(S,Date)} = -\sum_{i=1}^{6} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

$$= -\left(\frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6}\right)$$

$$\text{SplitInformation(S,Atr2)} = -\sum_{i=1}^{2} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

$$= -\left(\frac{4}{6}\log_2\frac{4}{6} + \frac{2}{6}\log_2\frac{2}{6}\right)$$

What if SplitInformation is very small or zero (|Si|≈ |S|) → GainRatio undefined or very large

→

Heuristic: Apply GainRatio test only for Attribute with above average Gain

Modul : Issues in Decision Tree Learning (DTL)

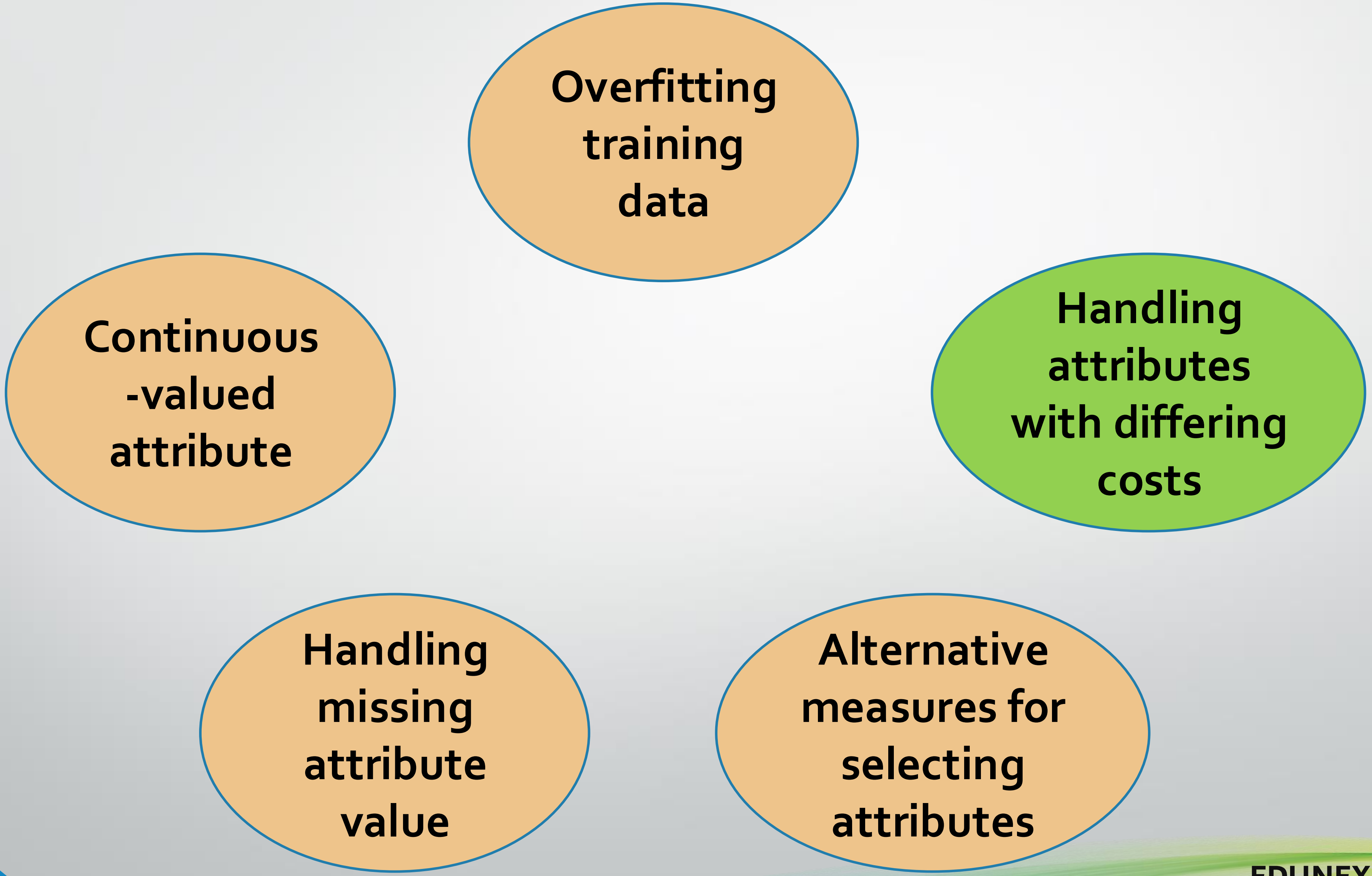
Attributes with Differing Costs

Nur ULFA Maulidevi

KK IF - Teknik Informatika- STEI ITB

Pembelajaran Mesin
(Machine Learning)

Issues in DTL



Overfitting training data

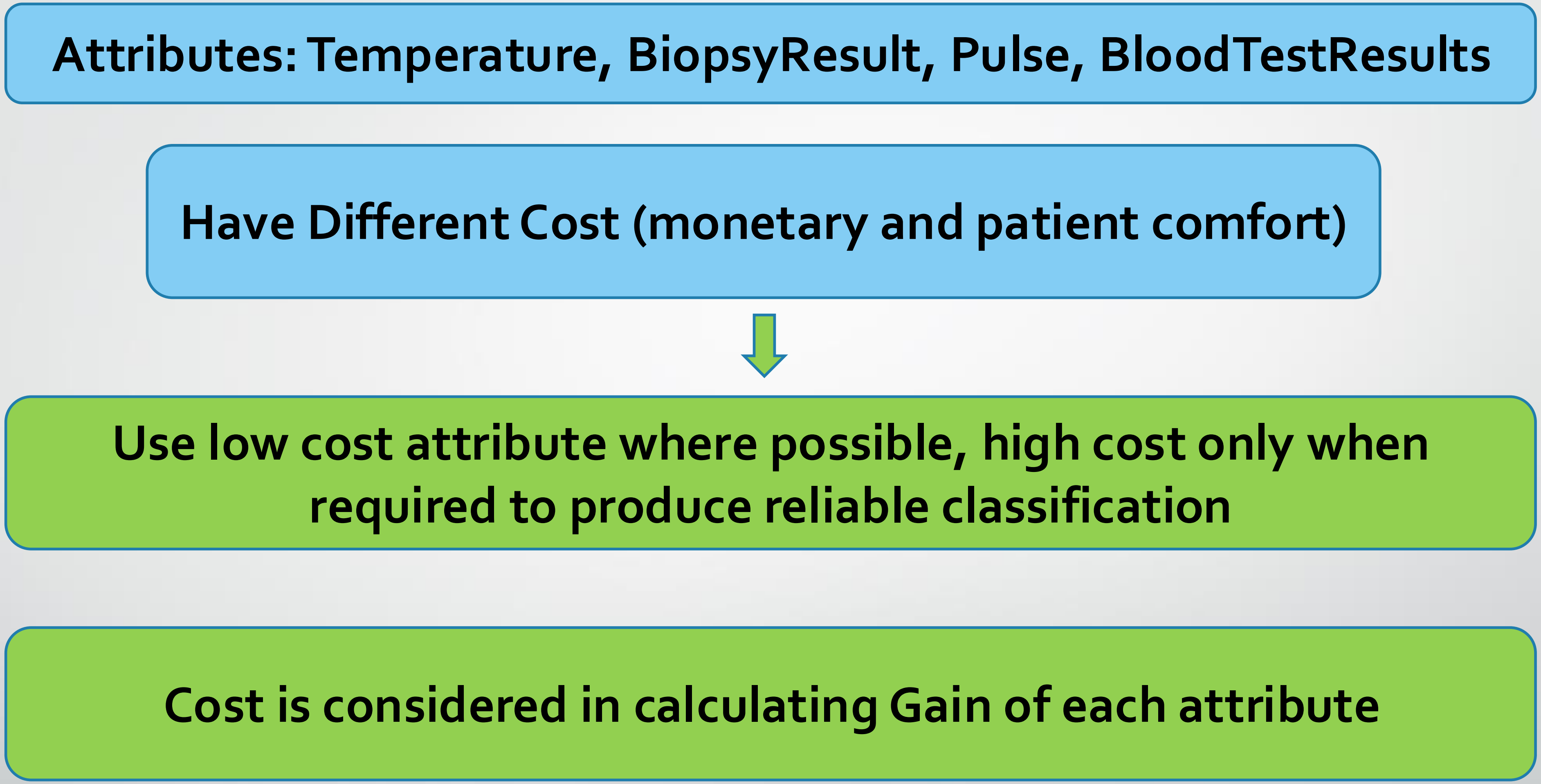
Handling attributes with differing costs

Continuous -valued attribute

Handling missing attribute value

Alternative measures for selecting attributes

Attribute with Different Cost



Attributes: Temperature, BiopsyResult, Pulse, BloodTestResults

Have Different Cost (monetary and patient comfort)

Use low cost attribute where possible, high cost only when required to produce reliable classification

Cost is considered in calculating Gain of each attribute

Approaches

Tan and Schlimmer (1990) and Tan (1993):

$$\frac{Gain^2(S,A)}{Cost(A)}$$

Nunez (1988):

$$\frac{2^{Gain(S,A)}-1}{(Cost(A)+1)^w}$$

Where $w \in [0,1]$ determine importance of cost

Exercise

Outlook	Temp	Humidity	Windy	Class
sunny	75	70	TRUE	Play
sunny	80	90	TRUE	Don't Play
sunny	85	85	FALSE	Don't Play
sunny	72	95	FALSE	Don't Play
sunny	69	70	FALSE	Play
?	72	90	TRUE	Play
overcast	83	78	FALSE	Play
overcast	64	65	TRUE	Play
overcast	81	75	FALSE	Play
rain	71	80	TRUE	Don't Play
rain	65	70	TRUE	Don't Play
rain	75	80	FALSE	Play
rain	68	80	FALSE	Play
rain	70	96	FALSE	Play

1. What is GainRatio for Outlook?

2. What are Examples (instances) for Outlook = sunny and what is the weight for instance with outlook=? related with outlook=sunny

3. Based on the result of question (2), define the threshold in "Humidity" discretization, and the leaf node for each branch.

THANK YOU

INSTITUT TEKNOLOGI BANDUNG
1920