MACHINE LEARNING

- Symbol Based Learning
- Framework for Symbol-Based Learning
- The ID3 Decision Tree Induction Algorithm
- Inductive bias and Learnability
- Knowledge and Learning
- Unsupervised learning
- Reinforcement Learning

MACHINE LEARNING

- Machine learning is **a branch of Artificial Intelligence (AI) & Computer Science that focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.**
- The three machine learning types are **supervised, unsupervised, and reinforcement learning**
- **Examples:**
  - Facial recognition.
  - Product recommendations.
  - Email automation and spam filtering.

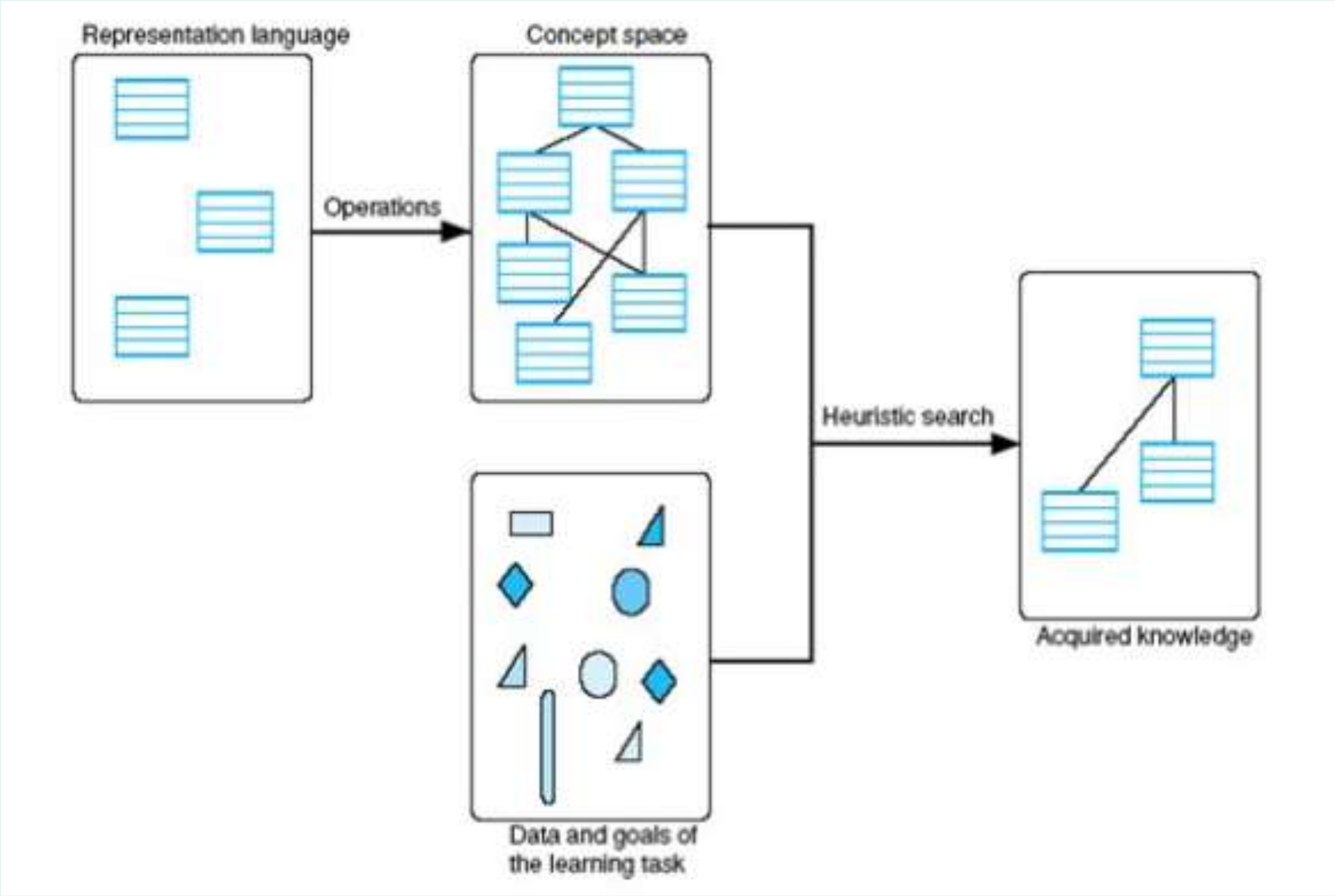
SYMBOL-BASED LEARNING

- Symbol-based learning methods, begin with a set of symbols that represent the entities and relationships of a problem domain
- Symbolic algorithms attempt to infer novel, valid, and useful generalizations that can be expressed using these symbols.

Learning:-

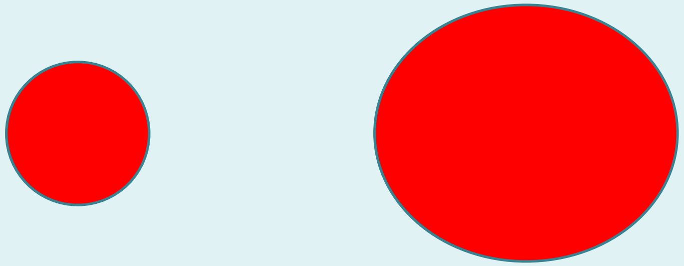
1. *allows the system to perform better the second time*
2. *involves changes in the learner*

THE FRAMEWORK FOR SYMBOL-BASED LEARNING



THE FRAMEWORK EXAMPLE

- Data



- The representation:
  - Size(small)^color(red)^shape(round)
  - Size(large)^color(red)^shape(round)

THE FRAMEWORK EXAMPLE

A set of operations:

Based on

- Size(small)^color(red)^shape(round)

Replace a single constant with a variable produces the generalizations:

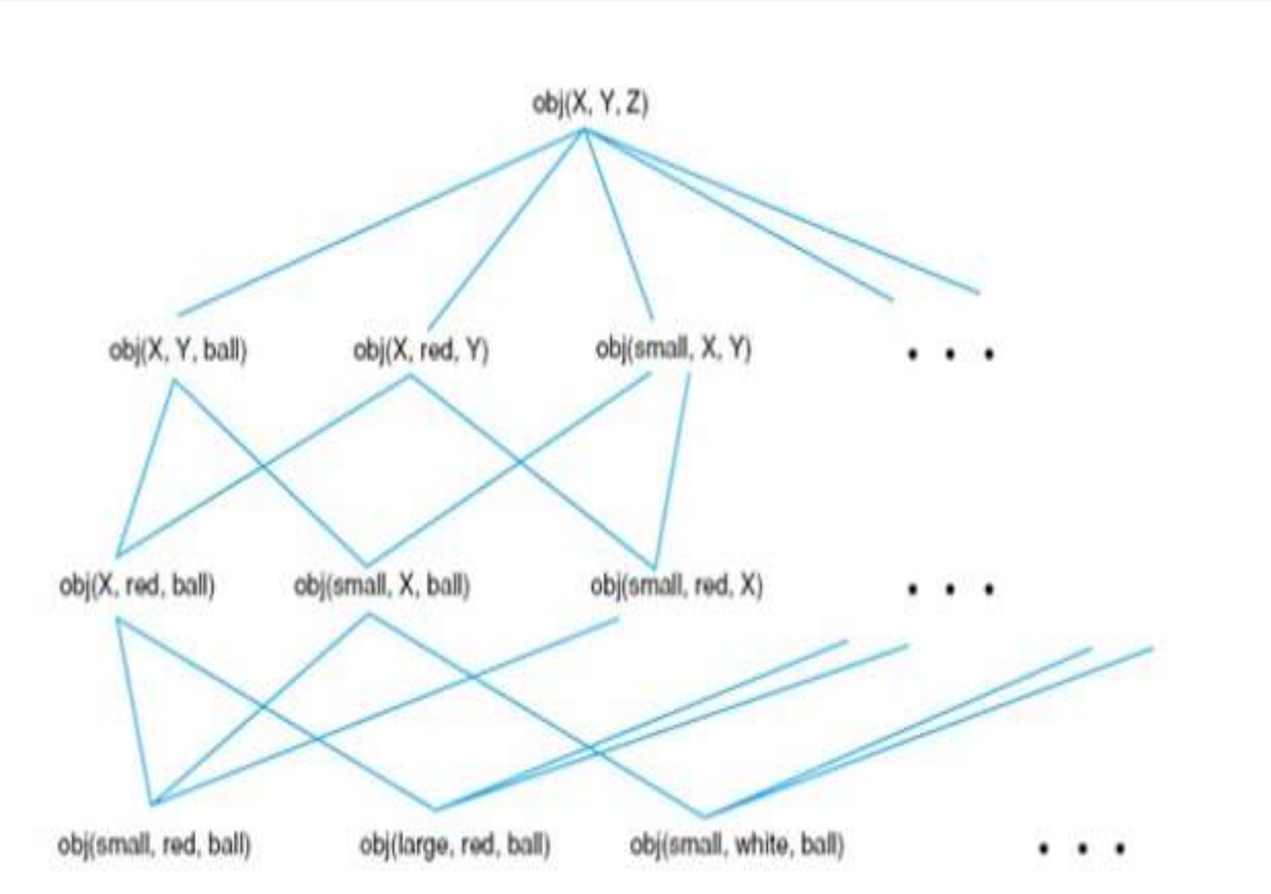
- Size(X)^color(red)^shape(round)
- Size(small)^color(X)^shape(round)
- Size(small)^color(red)^shape(X)

THE FRAMEWORK EXAMPLE

The concept space:

- The learner must search this space to find the desired concept.
- The complexity of this concept space is a primary measure of the difficulty of a learning problem

THE FRAMEWORK EXAMPLE:
Heuristic search



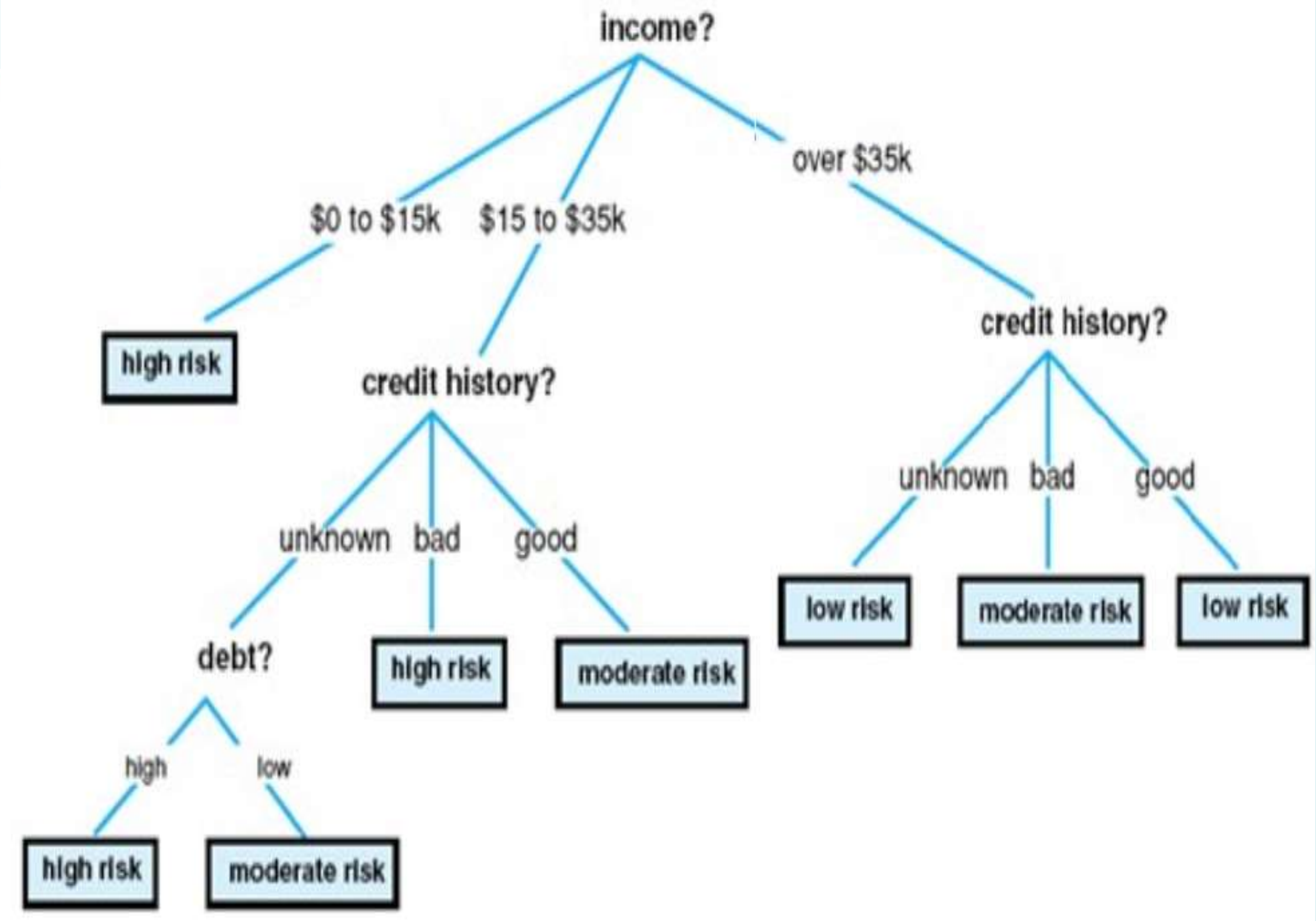
- Based on
  - Size(small)^color(red)^shape(round)
- The learner will make that example a candidate “ball” concept; this concept correctly classifies the only positive instance
- If the algorithm is given a second positive instance
  - Size(large)^color(red)^shape(round)
- The learner may generalize the candidate “ball” concept to
  - Size(Y)^color(red)^shape(round)

LEARNING PROCESS

- The training data is a series of positive and negative examples of the concept:
- Examples: world structures that fit a category, along with near misses.
- The latter are instances that almost belong to the category but fail on one property or relation

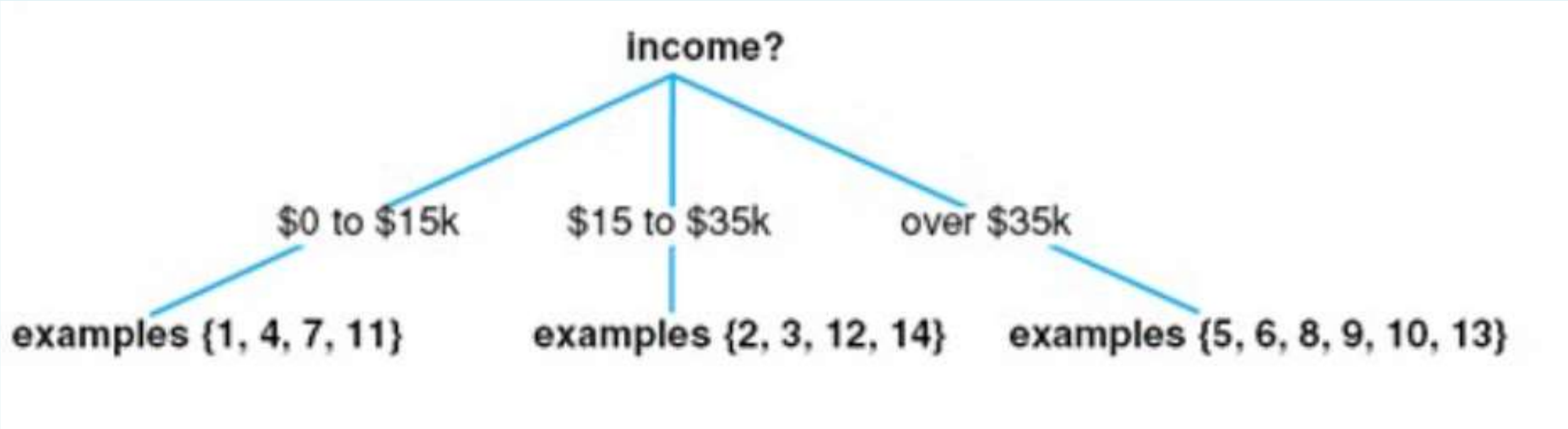
THE ID3 DECISION TREE INDUCTION ALGORITHM

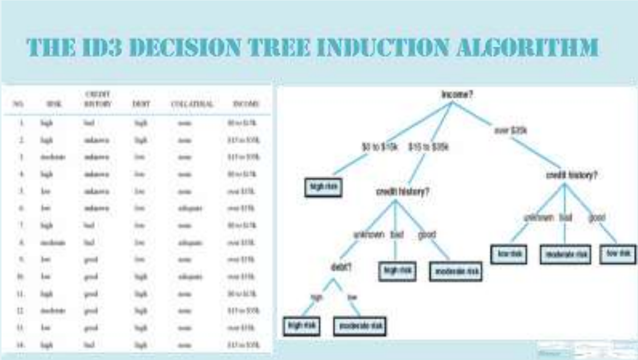
NO.	RISK	CREDIT HISTORY	DEBT	COLLATERAL	INCOME
1.	high	bad	high	none	$0 to $15k
2.	high	unknown	high	none	$15 to $35k
3.	moderate	unknown	low	none	$15 to $35k
4.	high	unknown	low	none	$0 to $15k
5.	low	unknown	low	none	over $35k
6.	low	unknown	low	adequate	over $35k
7.	high	bad	low	none	$0 to $15k
8.	moderate	bad	low	adequate	over $35k
9.	low	good	low	none	over $35k
10.	low	good	high	adequate	over $35k
11.	high	good	high	none	$0 to $15k
12.	moderate	good	high	none	$15 to $35k
13.	low	good	high	none	over $35k
14.	high	bad	high	none	$15 to $35k



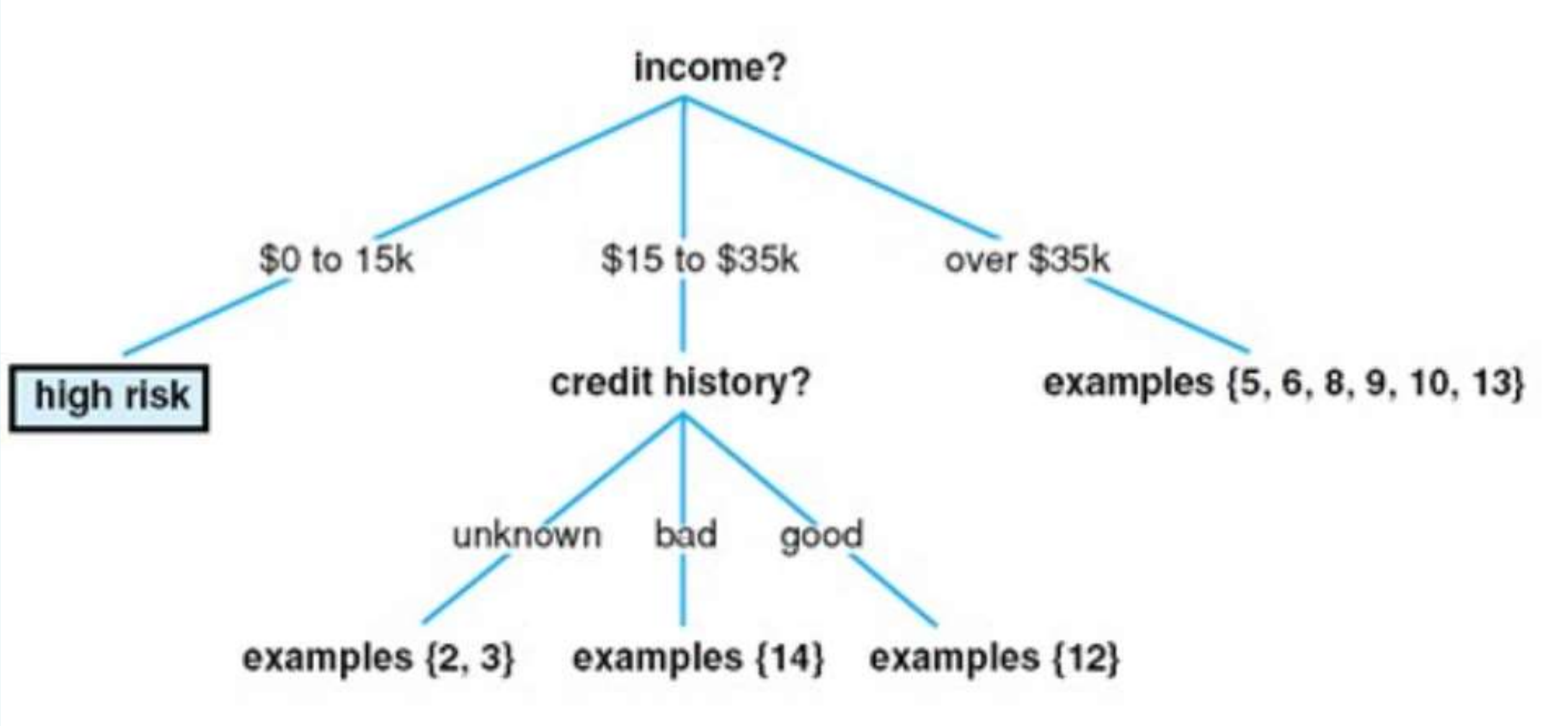
THE ID3 DECISION TREE INDUCTION ALGORITHM

- **ID3 constructs decision trees in a top-down fashion.**
  - ID3 selects a property to test at the current node of the tree and uses this test to partition the set of examples
  - The algorithm recursively constructs a sub-tree for each partition
  - This continues until all members of the partition are in the same class
- For example, ID3 selects income as the root property for the first step
- **How to select the 1st node? (and the following nodes)**
  - ID3 measures the information gained by making each property the root of current subtree
  - It picks the property that provides the greatest information gain





ID3 – ATTRIBUTE SELECTION



ID3 – ATTRIBUTE SELECTION

- If all the examples in the table occur with equal probability, then:
- P(risk is high)=6/14
- P(risk is moderate)=3/14
- P(risk is low)=5/14

$$I(M)=\sum_{i=1}^{n}-p(m_i)\log_2(p(m_i))$$

$$Info(D)=I(6,3,5)=-\frac{6}{14}\log_2(\frac{6}{14})-\frac{3}{14}\log_2(\frac{3}{14})-\frac{5}{14}\log_2(\frac{5}{14})=1.531$$

NO.	RISK	CREDIT HISTORY	DEBT	COLLATERAL	INCOME
1.	high	bad	high	none	\$0 to \$15k
2.	high	unknown	high	none	\$15 to \$35k
3.	moderate	unknown	low	none	\$15 to \$35k
4.	high	unknown	low	none	\$0 to \$15k
5.	low	unknown	low	none	over \$35k
6.	low	unknown	low	adequate	over \$35k
7.	high	bad	low	none	\$0 to \$15k
8.	moderate	bad	low	adequate	over \$35k
9.	low	good	low	none	over \$35k
10.	low	good	high	adequate	over \$35k
11.	high	good	high	none	\$0 to \$15k
12.	moderate	good	high	none	\$15 to \$35k
13.	low	good	high	none	over \$35k
14.	high	bad	high	none	\$15 to \$35k

ID3 – ATTRIBUTE SELECTION

$$Info_{income}(D) = \frac{4}{14} I(4,0,0) + \frac{4}{14} I(2,2,0) + \frac{6}{14} I(0,1,5) = 0.564$$

$$Info(D) = I(4,0,0) = -\frac{4}{4}\log_2(\frac{4}{4}) - \frac{0}{4}\log_2(\frac{0}{4}) - \frac{0}{4}\log_2(\frac{0}{4}) = 0$$

$$Info(D) = I(2,2,0) = -\frac{2}{4}\log_2(\frac{2}{4}) - \frac{2}{4}\log_2(\frac{2}{4}) - \frac{0}{4}\log_2(\frac{0}{4}) = 1.0$$

$$Info(D) = I(0,1,5) = -\frac{0}{6}\log_2(\frac{0}{6}) - \frac{1}{6}\log_2(\frac{1}{6}) - \frac{5}{6}\log_2(\frac{5}{6}) = 0.650$$

NO.	RISK	CREDIT HISTORY	DEBT	COLLATERAL	INCOME
1.	high	bad	high	none	$0 to $15k
2.	high	unknown	high	none	$15 to $35k
3.	moderate	unknown	low	none	$15 to $35k
4.	high	unknown	low	none	$0 to $15k
5.	low	unknown	low	none	over $35k
6.	low	unknown	low	adequate	over $35k
7.	high	bad	low	none	$0 to $15k
8.	moderate	bad	low	adequate	over $35k
9.	low	good	low	none	over $35k
10.	low	good	high	adequate	over $35k
11.	high	good	high	none	$0 to $15k
12.	moderate	good	high	none	$15 to $35k
13.	low	good	high	none	over $35k
14.	high	bad	high	none	$15 to $35k

The information gain form income is:

Gain(income)= I[Risk]-I_{Income}[Risk]=1.531-0.564 =0.967

Similarly,

Gain(Credit history)=0.266

Gain(Debt)=0.063

Gain(Collateral)=0.206

Since income provides the greatest information gain, ID3 will select it as the root of the tree

THE ID3 DECISION TREE INDUCTION ALGORITHM

- ❖Select the attribute with the highest information gain
- ❖Let pi be the probability that an arbitrary tuple in D belongs to class Ci, estimated by |Ci, D|/|D|
- ❖Expected information (entropy) needed to classify a tuple in D:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

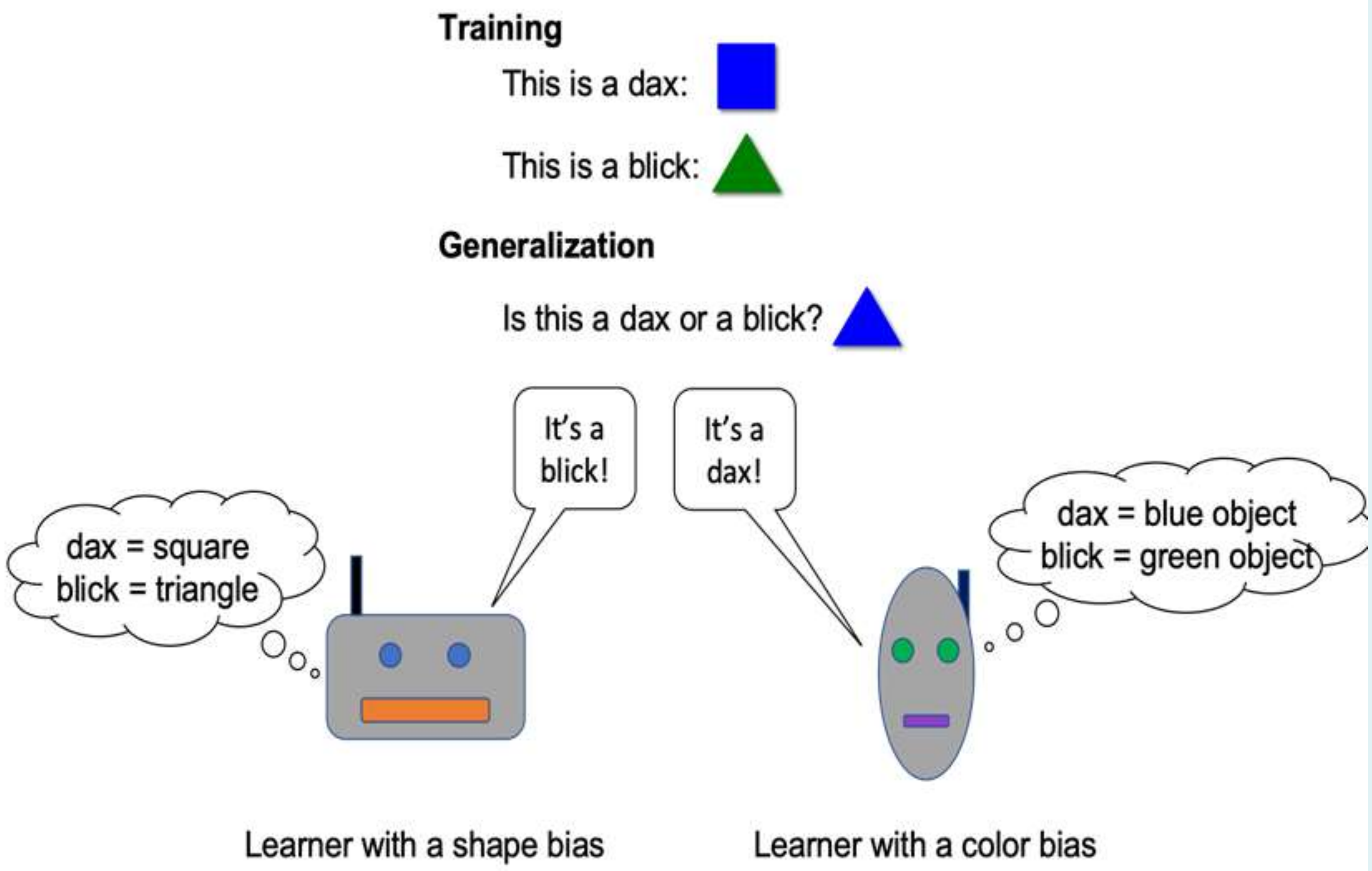
- ❖Information needed (after using A to split D into v partitions) to classify D:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times I(D_j)$$

- ❖Information gained by branching on attribute A

$$Gain(A) = Info(D) - Info_A(D)$$

INDUCTIVE BIAS



INDUCTIVE BIAS

- *Inductive bias* refers to a set of (explicit or implicit) assumptions made by a learning algorithm in order to perform induction, that is, to generalize a finite set of observations (training data) into a general model of the domain
- For eg. in the case of decision trees, the depth of the tress is the inductive bias.
- If the depth of the tree is too low, then there is too much generalization in the model.
- Similarly, if the depth of the tree is too much, there is too less generalization and while testing the model on a new example, we might reach a particular example used to train the model.

LEARNABILITY

❖Learnability is **the Mathematical Analysis of Machine Learning**.

❖It is also employed in language acquisition in arguments within linguistics.

❖A class of concepts is PAC (Probably Approximately Correct) learnable if an algorithm exists that executes efficiently

❖ a high probability of finding an approximately correct concept.

KNOWLEDGE & LEARNING

EXPLANATION BASED LEARNING (EBL)

❖Deals with the idea of single-example learning.

❖Requires a substantial number of training instances but there are two difficulties in this:

i. it is difficult to have such a number of training instances

ii. It may help us to learn certain things effectively, especially when we have enough knowledge.

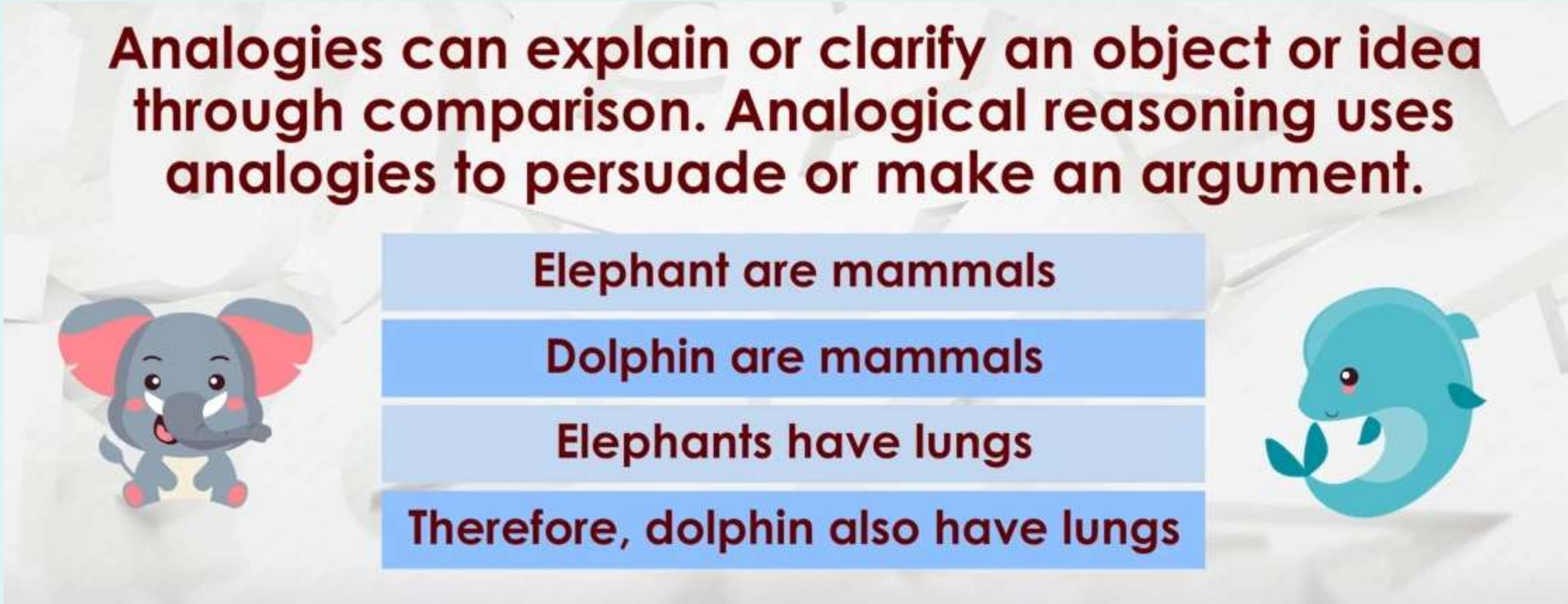
Hence, it is clear that instance-based learning is more data-intensive, and data-driven while EBL is more knowledge-intensive, and knowledge-driven.

❖Initially, an EBL system accepts a training example.

❖On the basis of the given goal concept, operationality criteria, and domain theory, it

ANALOGICAL REASONING

❖The ability to perceive and use relational similarity between two situations or events – is a fundamental aspect of human cognition.



FRAMEWORK FOR ANALOGICAL REASONING

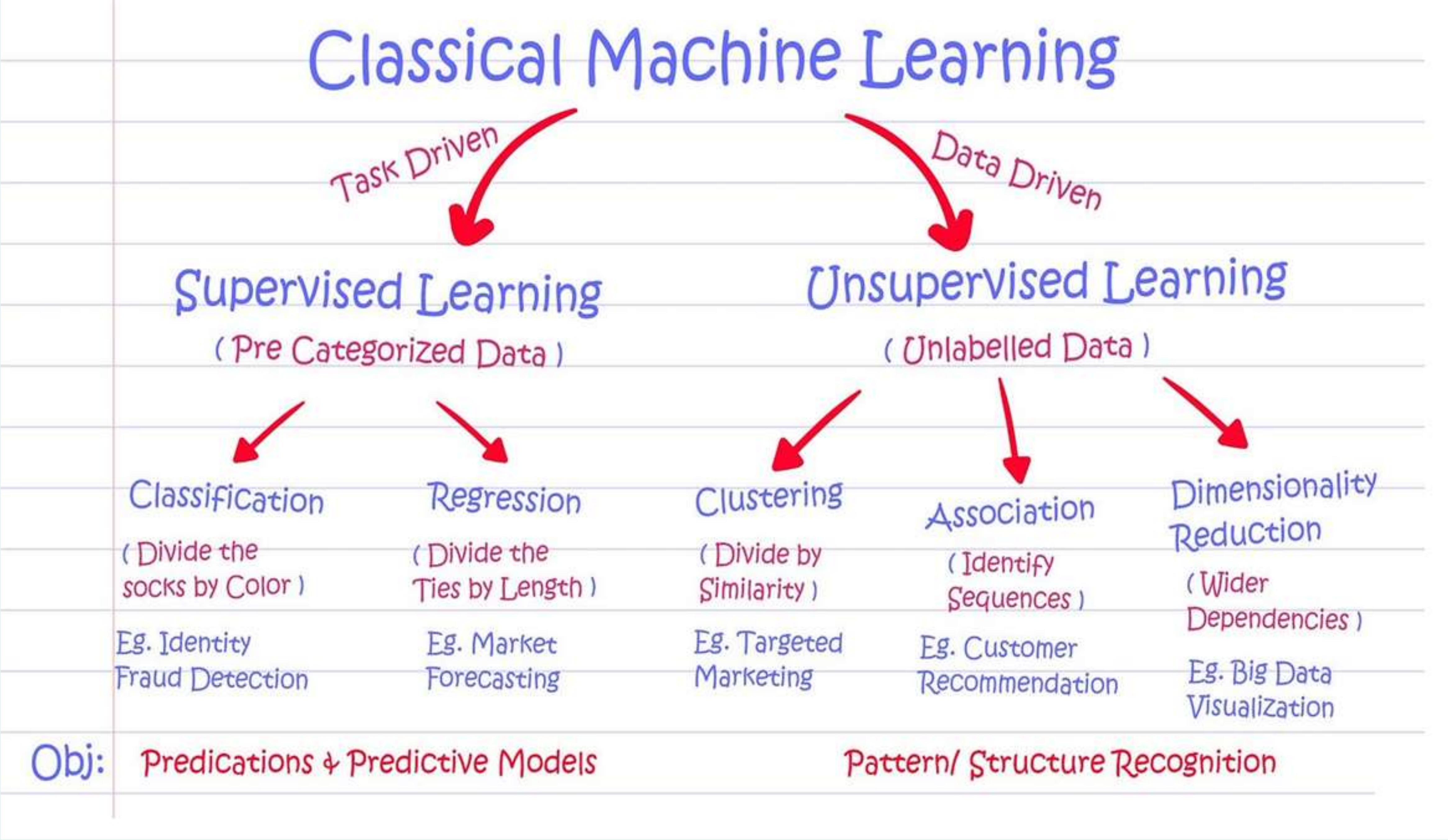
Retrieval. Given a target problem, it is necessary to select a potential source analog

Elaboration. Once the source has been retrieved, it is often necessary to derive additional features and relations of the source.

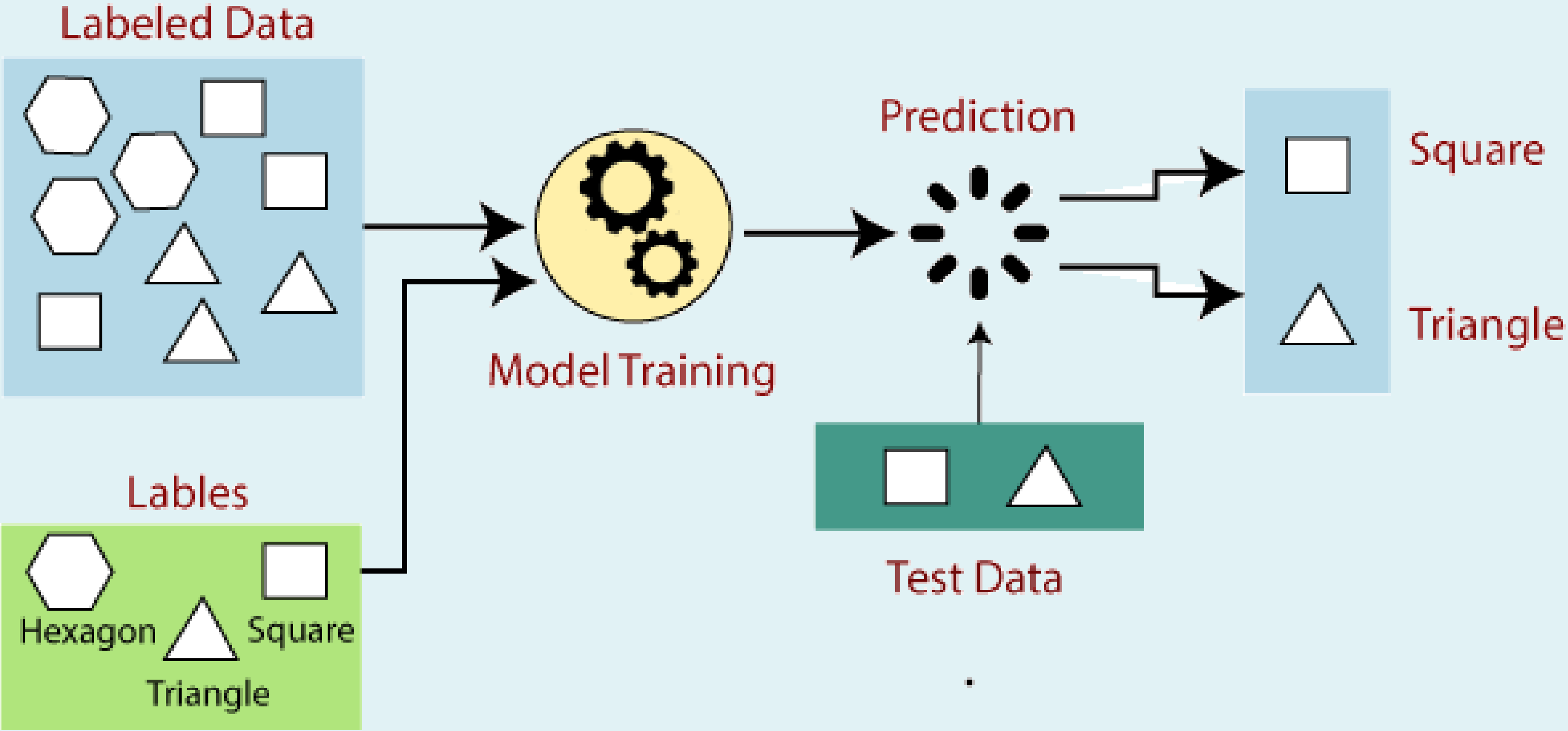
Mapping and inference. This stage involves developing the mapping of source attributes into the target domain.

Justification. Determine that the mapping is indeed valid

Learning. In this stage, the acquired knowledge is stored in a form that will be useful in the future


Classical Machine Learning
Task Driven
Data Driven
Supervised Learning
(Pre Categorized Data)
Unsupervised Learning
(Unlabelled Data)
Classification
(Divide the socks by Color)
Eg. Identity Fraud Detection
Regression
(Divide the Ties by Length)
Eg. Market Forecasting
Clustering
(Divide by Similarity)
Eg. Targeted Marketing
Association
(Identify Sequences)
Eg. Customer Recommendation
Dimensionality Reduction
(Wider Dependencies)
Eg. Big Data Visualization
Obj:
Predications & Predictive Models
Pattern/ Structure Recognition

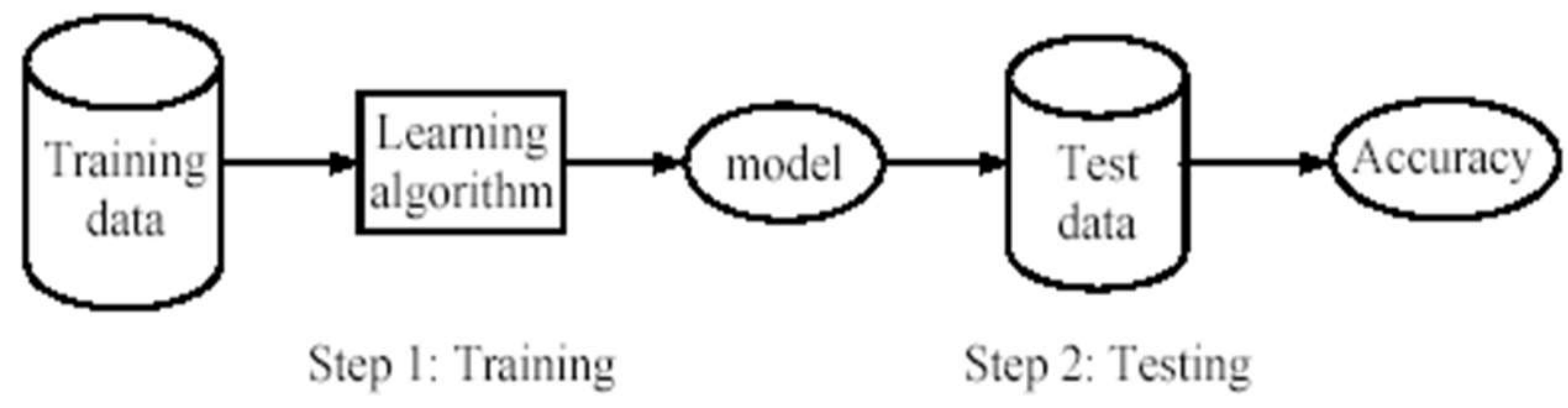
SUPERVISED LEARNING

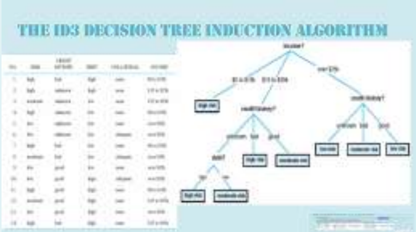


SUPERVISED LEARNING

Supervision:

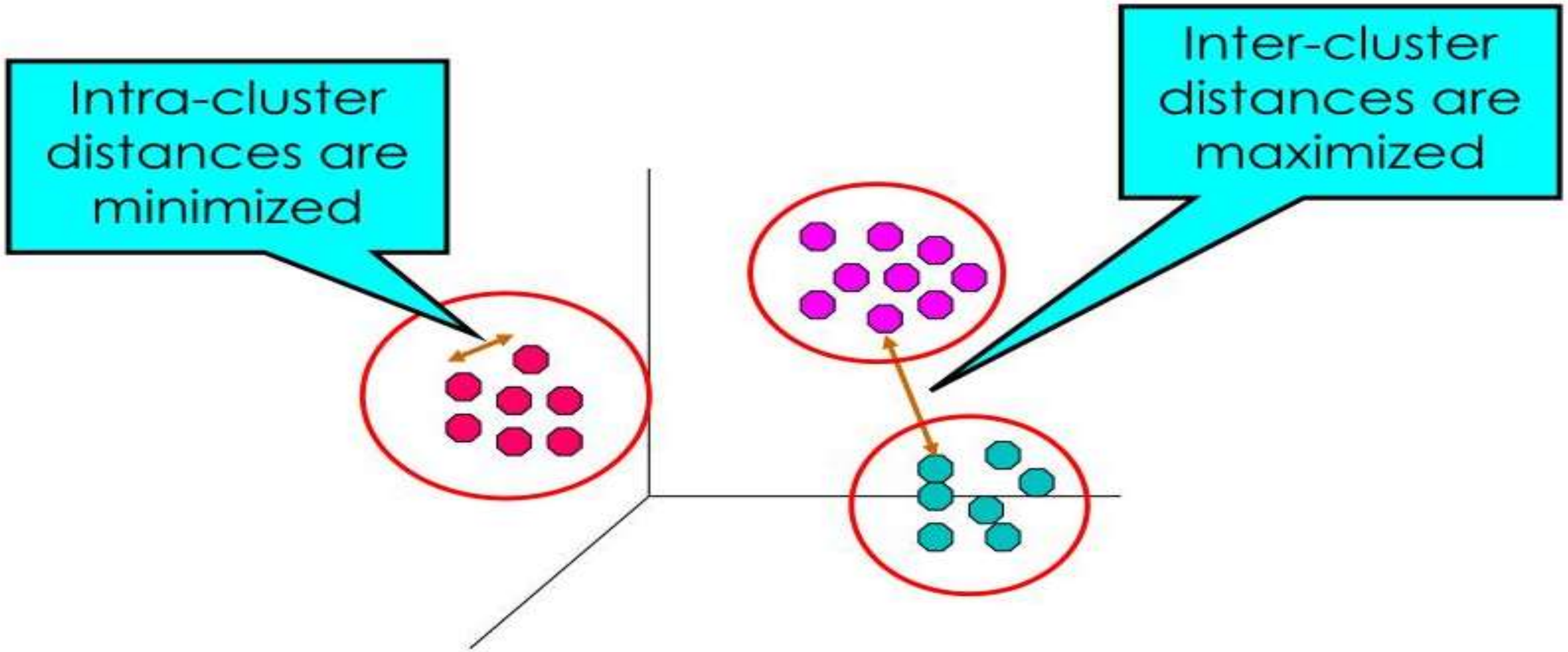
- ❖ The data (observations, measurements, etc.) are labeled with pre-defined classes. It is like a "teacher" gives the classes (supervision).
- ❖ Test data are classified into these classes too.
- ❖ Learning (training): Learn a model using the training data
- ❖ Testing: Test the model using unseen test data to assess the model accuracy



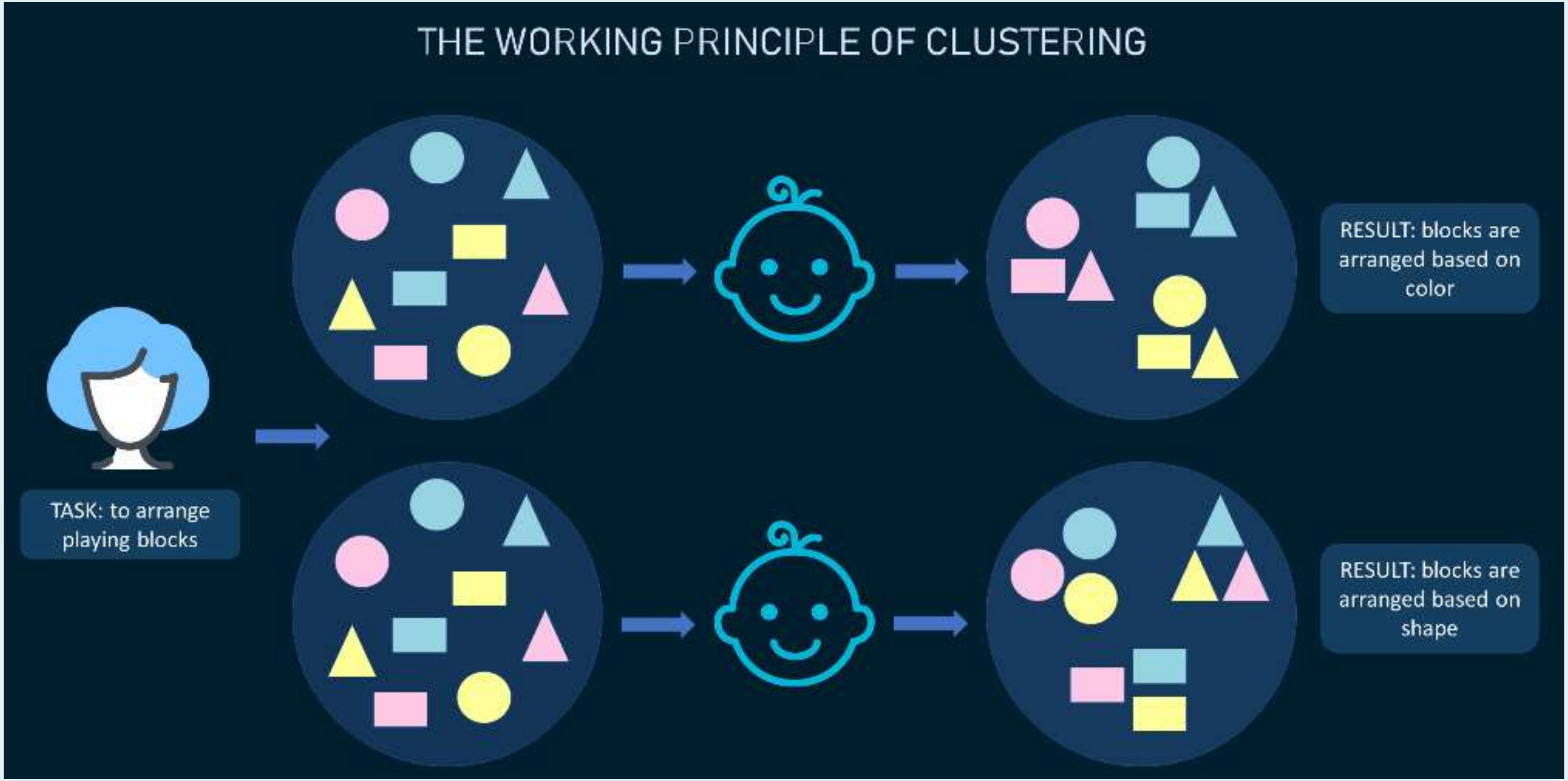


UNSUPERVISED LEARNING

- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



UNSUPERVISED LEARNING



AGGLOMERATIVE CLUSTERING

- ✓Examining all pairs of objects, select the pair with the highest degree of similarity, and mark that pair a cluster
- ✓Defining the features of the cluster as some function (such as average) of the features of the component members and then replacing the component objects with this cluster definition
- ✓Repeat this process on the collection of objects until all objects have been reduced to a single cluster
- ✓Result is a Binary Tree whose leaf nodes are instances and whose internal nodes are clusters of increasing size

UNSUPERVISED LEARNING

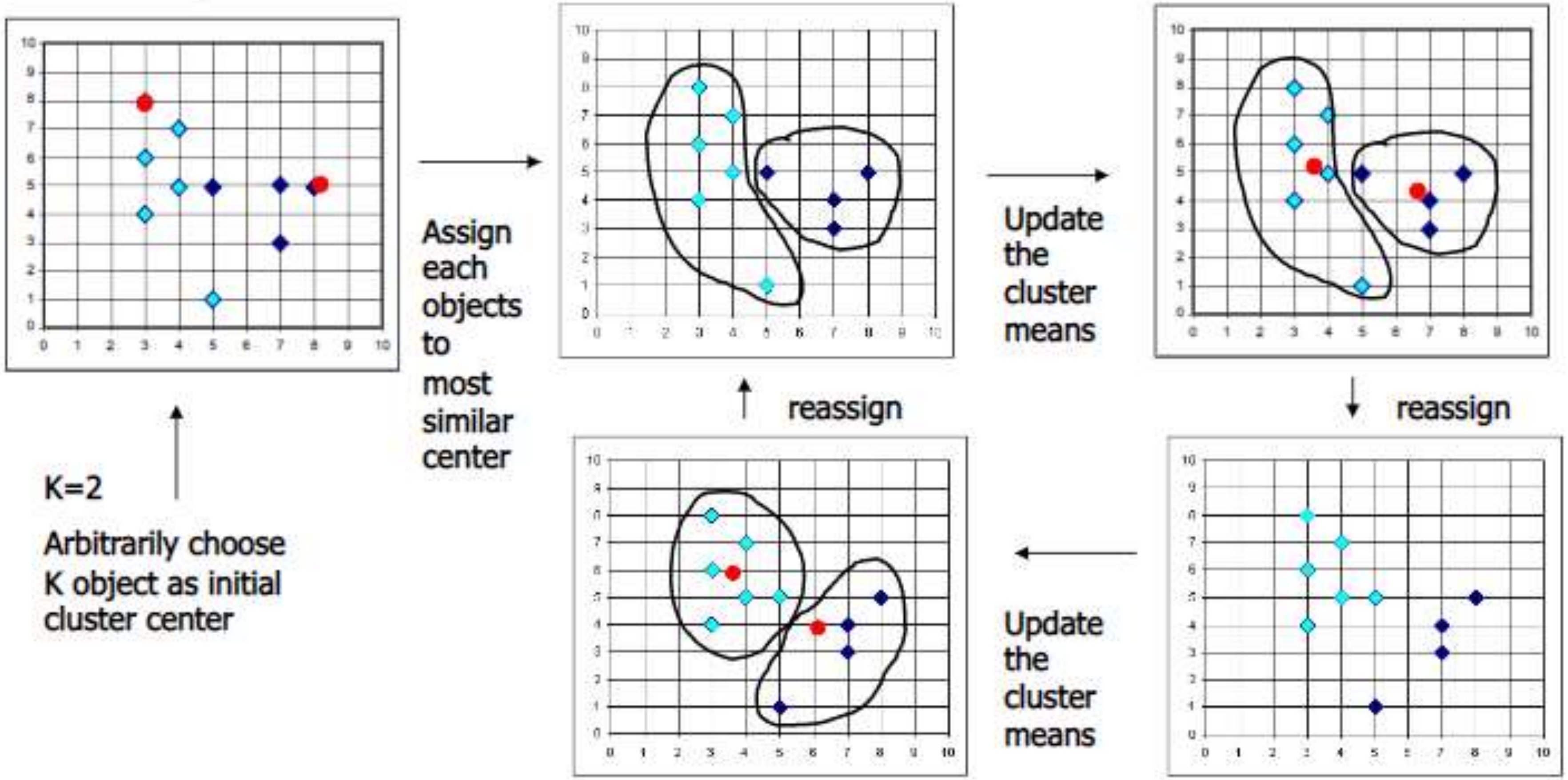
- Object1={small, red, rubber, ball}
- Object**2**={small, blue, rubber, ball}
- Object**3**={large, black, wooden, ball}

- This metric would compute the similary values:
  - Similarity(object1, object2)= ¾
  - Similarity(object1, object3)=1/4

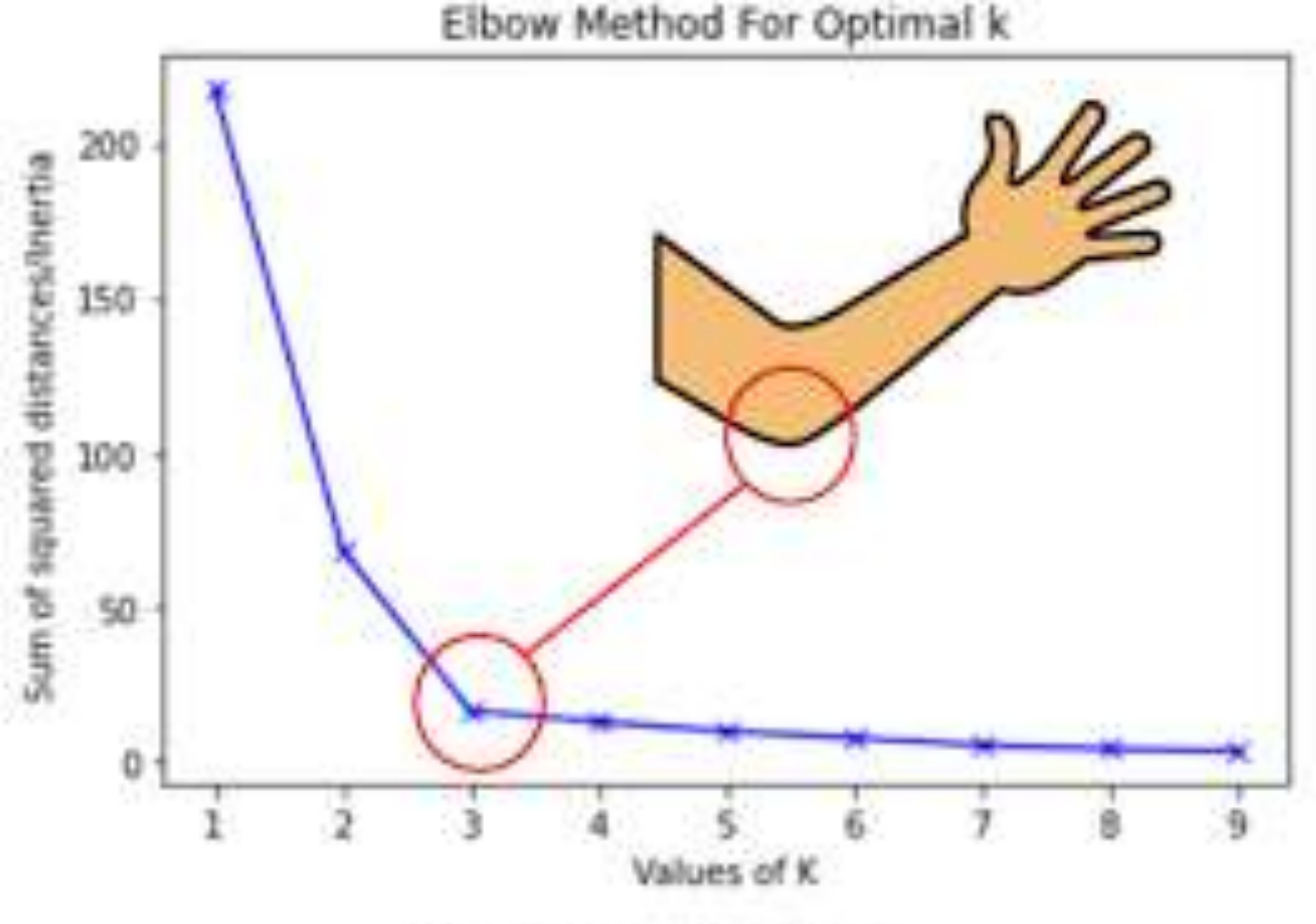
Partitioning Algorithms

- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: *k-means* and *k-medoids* algorithms
    - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
    - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

The *K-Means* Clustering Method

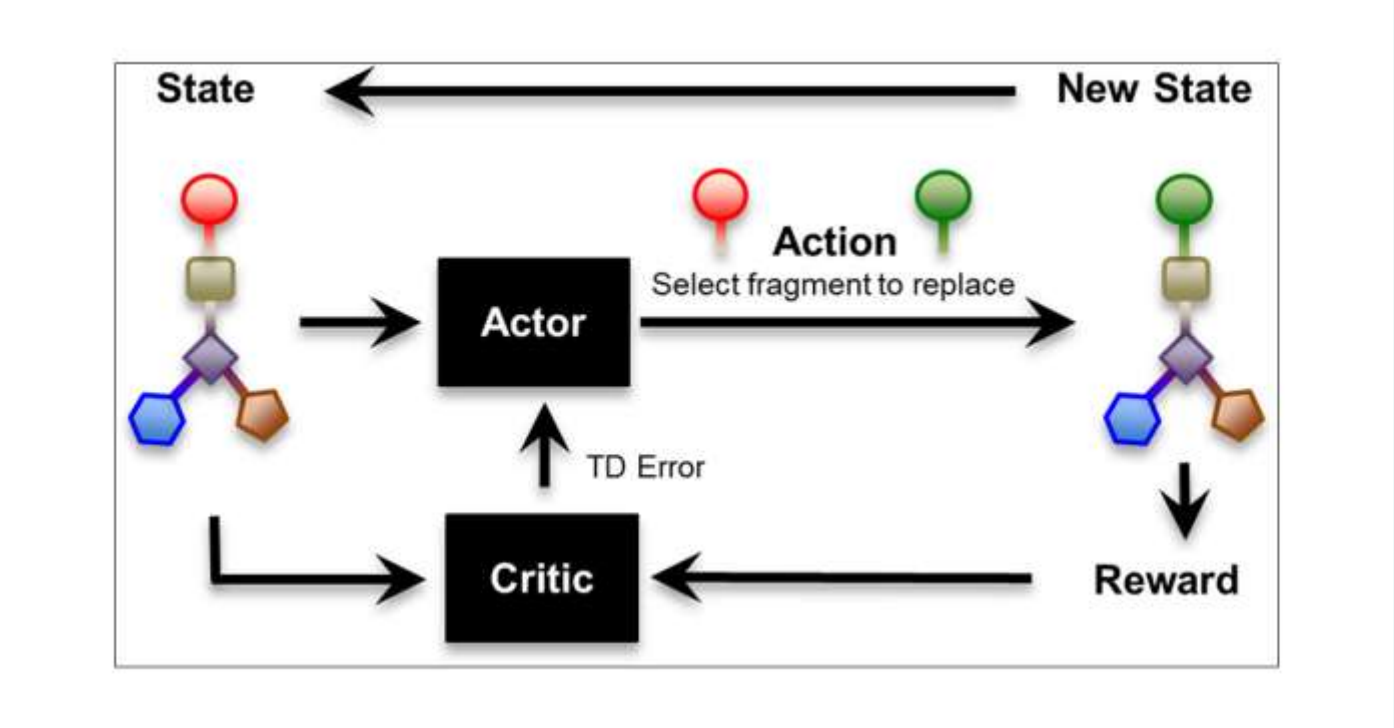


Choosing the optimal k



Line plot between K and inertia

REINFORCEMENT LEARNING



- Reinforcement learning is a machine learning training method based on rewarding desired behaviors and/or punishing undesired ones.
- A reinforcement learning agent is able to perceive and interpret its environment, take actions and learn through trial and error.
- The "reinforcement" in reinforcement learning **refers to how certain behaviors are encouraged, and others discouraged**.

REINFORCEMENT LEARNING

- Input: The input should be an initial state from which the model will start
- Output: There are many possible outputs as there are a variety of solutions to a particular problem
- Training: The training is based upon the input, the model will return a state and the user will decide to reward or punish the model based on its output. The model continues to learn.
- The best solution is decided based on the maximum reward.

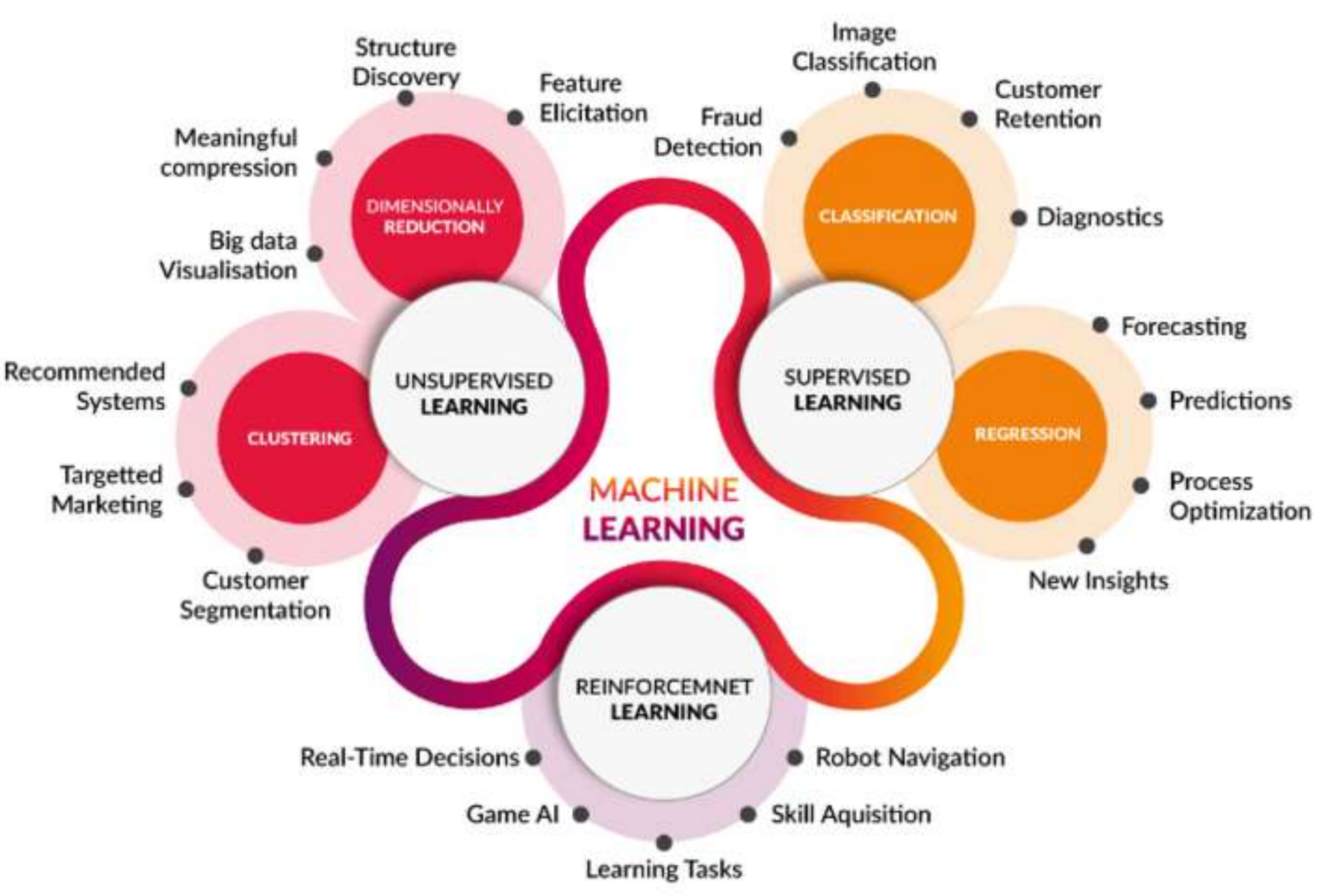
ELEMENTS OF REINFORCEMENT LEARNING

Policy: Policy defines the learning agent’s behavior for a given time period. It is a mapping from perceived states of the environment to actions to be taken when in those states.

Reward function: The reward function is used to define a goal in a reinforcement learning problem. It also maps each perceived state of the environment to a single number.

Value function: Value functions specify what is good in the long run. The value of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state.

APPLICATIONS



Criteria	Supervised ML	Unsupervised ML	Reinforcement ML
Definition	Machine Learns by using labelled data	Machine is trained using unlabelled data without any guidance.	Agent interacts with the environment by performing action. Learns by errors and rewards.
Type of data	Labelled data	Unlabelled data	No – predefined data.
Type of problems	Regression and classification	Association and Clustering	Reward and error based.
Supervision	External supervision	No supervision	No supervision
Algorithms	Linear Regression, Logistic Regression, Naïve Byes Decision trees	K – Means clustering, KNN (K-nearest neighbours) Principle Component Analysis Neural Networks	Monte Carlo, Q-Learning, SARSA
Aim	Calculate outcomes	Discover underlying patterns	Learn a series of action
Approach	Maps labelled inputs to the known outputs	Understands patterns & discover the output	Follow the trial and error method
Application	Risk Evaluation, Forecast Sales	Recommendation System, Anomaly Detection	Self-Driving Cars, Gaming, Healthcare