



# Machine Learning

(Course Code: 18B1WCI634 / 18B11BI611)

## Inductive Classification

**Mr. Sandeep Kumar Patel**

Assistant Professor

Jaypee University of Information Technology

Waknaghat, Solan, HP-173234

# Concept Learning Task

A concept learning task is the problem of **finding a general rule (concept)** that correctly classifies instances.

The goal is to learn a function that maps **instances**  $\rightarrow$  **class labels**.

i.e. learn a target concept (function) from a set of training examples.

Therefore,

## Definition

Concept learning is the task of inferring a Boolean-valued function from training examples of its inputs and outputs.

# Notations

- **Instance Space ( $X$ ):** All possible examples.

$$X = \{x_1, x_2, \dots, x_n\}$$

- **Target Concept ( $c$ ):** Unknown function to be learned.

$$c : X \rightarrow \{0, 1\}$$

- **Training Data ( $D$ ):** Labeled examples.

$$D = \{\langle x_i, c(x_i) \rangle\}$$

- **Hypothesis Space ( $H$ ):** Set of possible rules.

$$h : X \rightarrow \{0, 1\}$$

# Learning Objective

## Goal

Find a hypothesis  $h \in H$  such that:

$$h(x) \approx c(x), \quad \forall x \in X$$

**Hypotheses  $H$ :** Each hypothesis is a conjunction of constraints on the attributes. Constraints can be:

- ? (any value acceptable),
- $\phi$  (no value acceptable), or
- A specific value

# Concept Learning as a Search Problem

Concept learning can be viewed as a **search problem** where:

- The search space is the **hypothesis space  $H$** .
- Each hypothesis represents a possible concept.
- The goal is to find a hypothesis that is consistent with training data.

## Example

- **Attributes:** (Sky, Temp, Humidity, Wind, Water, Forecast)
- **Target concept ( $c$ ):** “Play Tennis”
- **Training example:**

$\langle (Sunny, Warm, Normal, Strong, Warm, Same), Yes \rangle$

- **Hypothesis ( $h$ ):**

$$h(x) = (Sky = Sunny) \wedge (Temp = Warm)$$

## Problem:

In The given Attribute table,

Attribute	Possible Values
Sky	{Sunny, Rainy, Cloudy} $\rightarrow 3$
Temp	{Warm, Cold} $\rightarrow 2$
Humidity	{Normal, High} $\rightarrow 2$
Wind	{Strong, Weak} $\rightarrow 2$
Water	{Warm, Cool} $\rightarrow 2$
Forecast	{Same, Change} $\rightarrow 2$

**Q.** Find Instance Space, Total number of concepts, Numbers of syntactic Hypothesis and semantic Hypothesis.

# Number of Concepts

**Instance Space Size:**

$$|X| = 3 \times 2 \times 2 \times 2 \times 2 \times 2 = 96$$

**Definition of a Concept:**

$$c : X \rightarrow \{0, 1\}$$

Each instance can be labeled independently as positive or negative.

**Total Number of Concepts:**

$$\text{Number of concepts} = 2^{|X|} = 2^{96}$$



## Number of Hypotheses

- **Syntactically:** For each attribute:

$$\text{Choices} = (\text{number of values}) + 2(\phi, ?)$$

$$|H| = (3 + 2)(2 + 2)^5 = 5 \times 4^5 = 4 \times 1024$$

$$|H| = 5120$$

- **Semantically:** For each attribute:

$$\text{Choices} = (\text{number of values}) + 1$$

$$|H| = (3 + 1)(2 + 1)^5 = 4 \times 3^5 = 4 \times 243 = 972$$

$$|H| = 972 + 1(h_0) = 973$$

# General-to-Specific Ordering of Hypotheses

**Specific hypothesis:** only matches very few positive examples.

**General hypothesis:** matches many positive examples.

## Definition

A hypothesis  $h_j$  is **more general than or equal to** hypothesis  $h_k$  (written  $h_j \geq_g h_k$ ) if:

$$\forall x \in X, (h_k(x) = 1 \implies h_j(x) = 1)$$

Example with Attributes: (*Sky, Temp, Humidity, Wind*)

$$h_1 = (?, \text{Warm}, ?, ?)$$

$$h_2 = (\text{Sunny}, \text{Warm}, \text{High}, ?)$$

So,

$$h_1 \geq_g h_2$$

# Concept Learning Task: Find-S Algorithm

## Training Examples:

Ex.	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

## Question:

Using the **Find-S Algorithm**, determine the final hypothesis  $h$  after processing all the above examples.

# Find-S Algorithm: Step 1 & Step 2

**Goal:** Find the most specific hypothesis  $h$  consistent with all positive training examples.

## Step 1: Initialization

- Start with the most specific hypothesis:

$$h \leftarrow \langle \emptyset, \emptyset, \dots, \emptyset \rangle$$

## Step 2: For each training example $(x, y)$

- If  $y$  is positive, then for each attribute  $i$ :
  - If  $h[i] = \emptyset$ , set  $h[i] \leftarrow x[i]$ .
  - If  $h[i] \neq x[i]$ , set  $h[i] \leftarrow ?$ .
  - Otherwise leave  $h[i]$  unchanged.

# Find-S Algorithm: Step 3 & Step 4

## Step 3: Processing negative examples

- If  $y$  is negative:
  - Do nothing (hypothesis is not updated).

## Step 4: Final hypothesis

- After all training examples are processed, return  $h$ .

**Output:** The most specific hypothesis consistent with all positive training examples.

## Find-S Steps:

$$S_0 = \langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$$

$$S_1 = \langle \text{Sunny}, \text{Warm}, \text{Normal}, \text{Strong}, \text{Warm}, \text{Same} \rangle \quad (\text{after Example 1})$$

$$S_2 = \langle \text{Sunny}, \text{Warm}, ?, \text{Strong}, \text{Warm}, \text{Same} \rangle \quad (\text{after Example 2})$$

$$S_3 = S_2 \quad (\text{Example 3 is negative, ignored})$$

$$S_{\text{final}} = \langle \text{Sunny}, \text{Warm}, ?, \text{Strong}, ?, ? \rangle \quad (\text{after Example 4})$$

## Final Hypothesis:

$$h(x) = \text{EnjoySport if } (Sky = \text{Sunny}) \wedge (AirTemp = \text{Warm}) \wedge (Wind = \text{Strong})$$

# Concept Learning Task: Loan Approval

Ex.	Income	Credit	Collateral	Age	Employment	Location	LoanApproval
1	High	Good	Yes	Young	Salaried	Urban	Yes
2	High	Good	No	Young	Salaried	Urban	Yes
3	Low	Bad	Yes	Old	SelfEmp	Rural	No
4	High	Good	Yes	Young	Salaried	Urban	Yes

**Q:** Using the **Find-S Algorithm**, what is the final hypothesis  $h$  after processing all the above examples?

# Find-S Solution: Loan Approval (Part 1)

Start with the most specific hypothesis:

$$S_0 = \langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle$$

Example 1 (Positive) : (High, Good, Yes, Young, Salaried, Urban)

$$S_1 = \langle \textit{High}, \textit{Good}, \textit{Yes}, \textit{Young}, \textit{Salaried}, \textit{Urban} \rangle$$

Example 2 (Positive) : (High, Good, No, Young, Salaried, Urban)

Collateral differs  $\rightarrow$  generalize to ?:

$$S_2 = \langle \textit{High}, \textit{Good}, ?, \textit{Young}, \textit{Salaried}, \textit{Urban} \rangle$$



## Find-S Solution: Loan Approval (Part 2)

Example 3 (Negative): (Low, Bad, Yes, Old, SelfEmp, Rural)

Ignore negative example; hypothesis unchanged:

$$S_3 = S_2 = \langle \textit{High}, \textit{Good}, ?, \textit{Young}, \textit{Salaried}, \textit{Urban} \rangle$$

Example 4 (Positive): (High, Good, Yes, Young, Salaried, Urban)

Already consistent, final hypothesis:

$$S = \langle \textit{High}, \textit{Good}, ?, \textit{Young}, \textit{Salaried}, \textit{Urban} \rangle$$

# Drawbacks of Find-S Algorithm

- **Ignores Negative Examples:** Updates hypothesis only on positive examples; negative examples are not used.
- **Single Hypothesis Output:** Produces only one final hypothesis, even if multiple consistent hypotheses exist.
- **Dependence on Initial Example:** The first positive example strongly influences the hypothesis, limiting flexibility.

# Concept Learning Task: Candidate Elimination Algorithm

## Training Examples:

Ex.	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

**Question:** Using the **Candidate Elimination Algorithm**, compute the **final Version Space**  $VS_{H,D}$  by updating the  $S$  (most specific) and  $G$  (most general) boundaries after each example.

# Prerequisites for Candidate Elimination Algorithm

- **Hypothesis Space ( $H$ ):** The set of all possible hypotheses. Each hypothesis is a conjunction of attribute constraints.
- **General-to-Specific Ordering:** A hypothesis  $h_j$  is more general than or equal to  $h_k$  if:

$$\forall x \in X, (h_k(x) = 1 \implies h_j(x) = 1)$$

- **Boundary Sets:**
  - $S$  — the set of most specific hypotheses consistent with the data.
  - $G$  — the set of most general hypotheses consistent with the data.
- **Version Space:** The set of all hypotheses in  $H$  that are consistent with the training data:

$$VS_{H,D} = \{h \in H \mid h(x) = c(x), \forall (x, c(x)) \in D\}$$

# Candidate Elimination Algorithm: Step 1 – Initialization

**Goal:** Learn the version space (all hypotheses consistent with the training data) by maintaining:

- **S** — the set of most specific hypotheses
- **G** — the set of most general hypotheses

## Step 1: Initialization

- Start with the most specific hypothesis:

$$S = \langle \emptyset, \emptyset, \dots \rangle$$

- Start with the most general hypothesis:

$$G = \langle ?, ?, \dots \rangle$$

# Candidate Elimination: Positive Example Update

**For a positive example**  $(x, y = 1)$ :

- ① Remove from  $G$  any hypotheses inconsistent with  $x$
- ② For each  $s \in S$  inconsistent with  $x$ :
  - Generalize  $s$  minimally so that  $s(x) = 1$
- ③ Remove any hypothesis in  $S$  that is more general than another hypothesis in  $S$

**Effect:**  $S$  becomes more general to include positive examples,  $G$  stays consistent

# Candidate Elimination: Negative Example Update

**For a negative example**  $(x, y = 0)$ :

- ① Remove from  $S$  any hypotheses inconsistent with  $x$
- ② For each  $g \in G$  inconsistent with  $x$ :
  - Specialize  $g$  minimally so that  $g(x) = 0$
- ③ Remove any hypothesis in  $G$  that is more specific than another hypothesis in  $G$

**Effect:**  $G$  becomes more specific to exclude negative examples,  $S$  stays consistent

# Candidate Elimination: Output

**After processing all training examples:**

$$VS_{H,D} = \{h \mid S \leq_g h \leq_g G\}$$

- All hypotheses in the version space are consistent with all training examples
- $S$  = most specific hypotheses
- $G$  = most general hypotheses
- The version space represents all possible solutions



# Candidate Elimination Algorithm: Step-by-Step

## Initialization:

- $S_0 = \{\langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle\}$
- $G_0 = \{\langle ?, ?, ?, ?, ?, ? \rangle\}$

**Example 1 (Positive):** (Sunny, Warm, Normal, Strong, Warm, Same)

- $S_1 = \{\langle \text{Sunny}, \text{Warm}, \text{Normal}, \text{Strong}, \text{Warm}, \text{Same} \rangle\}$
- $G_1 = \{\langle ?, ?, ?, ?, ?, ? \rangle\}$

# Candidate Elimination Algorithm

**Example 2 (Positive):** (Sunny, Warm, High, Strong, Warm, Same)

- Generalize  $S_1$  on Humidity  $\rightarrow ?$
- $S_2 = \{\langle \text{Sunny}, \text{Warm}, ?, \text{Strong}, \text{Warm}, \text{Same} \rangle\}$
- $G_2 = \{\langle ?, ?, ?, ?, ?, ? \rangle\}$

**Example 3 (Negative):** (Rainy, Cold, High, Strong, Warm, Change)

- Specialize  $G$  to exclude negative while  $\geq S$
- $G_3 = \{\langle \text{Sunny}, ?, ?, ?, ?, ? \rangle, \langle ?, \text{Warm}, ?, ?, ?, ? \rangle, \langle ?, ?, ?, ?, ?, \text{Same} \rangle\}$
- $S_3 = S_2$

# Candidate Elimination Algorithm

**Example 4 (Positive):** (Sunny, Warm, High, Strong, Cool, Change)

- Remove inconsistent hypotheses from  $G_3$
- $G_4 = \{\langle \text{Sunny}, ?, ?, ?, ?, ? \rangle, \langle ?, \text{Warm}, ?, ?, ?, ? \rangle\}$
- Generalize  $S_3$  to cover Water and Forecast differences
- $S_4 = \{\langle \text{Sunny}, \text{Warm}, ?, \text{Strong}, ?, ? \rangle\}$

# Final Version Space

**Most Specific Boundary  $S$ :**

$$S = \{\langle \text{Sunny}, \text{Warm}, ?, \text{Strong}, ?, ? \rangle\}$$

**Most General Boundary  $G$ :**

$$G = \{\langle \text{Sunny}, ?, ?, ?, ?, ? \rangle, \langle ?, \text{Warm}, ?, ?, ?, ? \rangle\}$$

**Final Version Space:**

$$VS_{H,D} = \{h \in H \mid S \preceq h \preceq G\}$$

$$G = \{\langle \text{Sunny}, \text{Warm}, ?, ?, ?, ? \rangle, \langle \text{Sunny}, ?, ?, \text{Strong}, ?, ? \rangle, \langle ?, \text{Warm}, ?, \text{Strong}, ?, ? \rangle\}$$

# Concept Learning Task: Car Buying Decision

## Training Examples:

Ex.	Make	Doors	Engine	Color	Manual	Buy?
1	Toyota	4	Petrol	Red	Yes	Yes
2	Honda	2	Petrol	Blue	Yes	Yes
3	Ford	4	Diesel	Red	No	No
4	Toyota	2	Petrol	Red	Yes	Yes

**Q.** Use the **Candidate Elimination Algorithm** to update the  $S$  and  $G$  boundaries after each example.

## Step 0: Initialization

- Most specific hypothesis:  $S_0 = \langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle$
- Most general hypothesis:  $G_0 = \langle ?, ?, ?, ?, ? \rangle$

## Step 1: Example 1 (Positive)

- Example 1:  $\langle \text{Toyota}, 4, \text{Petrol}, \text{Red}, \text{Yes}, \text{Yes} \rangle$
- Update  $S$  to first positive example:

$$S_1 = \langle \text{Toyota}, 4, \text{Petrol}, \text{Red}, \text{Yes} \rangle$$

- $G$  remains maximally general:

$$G_1 = \langle ?, ?, ?, ?, ? \rangle$$

## Step 2: Example 2 (Positive)

- Example 2:  $\langle \text{Honda}, 2, \text{Petrol}, \text{Blue}, \text{Yes}, \text{Yes} \rangle$
- Generalize  $S$  where attributes differ:

$$S_2 = \langle ?, ?, \text{Petrol}, ?, \text{Yes} \rangle$$

- $G$  unchanged:

$$G_2 = \langle ?, ?, ?, ?, ? \rangle$$



## Step 3: Example 3 (Negative)

- Example 3:  $\langle \text{Ford}, 4, \text{Diesel}, \text{Red}, \text{No}, \text{No} \rangle$
- Specialize  $G$  to exclude negative example while including  $S$ :

$$G_3 = \langle ?, ?, \text{Petrol}, ?, ? \rangle$$

- $S$  remains unchanged:

$$S_3 = \langle ?, ?, \text{Petrol}, ?, \text{Yes} \rangle$$

## Step 4: Example 4 (Positive)

- Example 4:  $\langle \text{Toyota}, 2, \text{Petrol}, \text{Red}, \text{Yes}, \text{Yes} \rangle$
- Update  $S$  to include this positive example (already covered by  $S_3$ ):

$$S_4 = \langle ?, ?, \text{Petrol}, ?, \text{Yes} \rangle$$

- $G$  unchanged:

$$G_4 = \langle ?, ?, \text{Petrol}, ?, ? \rangle$$

# Final Version Space

- Most Specific Hypothesis:

$$S = \langle ?, ?, \text{Petrol}, ?, \text{Yes} \rangle$$

- Most General Hypothesis:

$$G = \langle ?, ?, \text{Petrol}, ?, ? \rangle$$

- Version space narrowed down to **Engine = Petrol** and **Manual = Yes**.

# Concept Learning Task: Fruit Edibility Decision

## Training Examples:

Ex.	Color	Size	Shape	Taste	Sweet	Edible
1	Red	Small	Round	Sour	Yes	Yes
2	Yellow	Small	Round	Sweet	Yes	Yes
3	Green	Large	Oval	Bitter	No	No
4	Red	Small	Oval	Sweet	Yes	Yes

*Use the Candidate Elimination Algorithm to update the  $S$  and  $G$  boundaries after each example.*

## Step 0: Initialization

- Most specific hypothesis:  $S_0 = \langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle$
- Most general hypothesis:  $G_0 = \langle ?, ?, ?, ?, ? \rangle$

## Step 1: Example 1 (Positive)

- Example 1:  $\langle \text{Red, Small, Round, Sour, Yes, Yes} \rangle$
- Update  $S$  to first positive example:

$$S_1 = \langle \text{Red, Small, Round, Sour, Yes} \rangle$$

- $G$  remains maximally general:

$$G_1 = \langle ?, ?, ?, ?, ? \rangle$$

## Step 2: Example 2 (Positive)

- Example 2:  $\langle \text{Yellow, Small, Round, Sweet, Yes, Yes} \rangle$
- Generalize  $S$  where attributes differ:

$$S_2 = \langle ?, \text{Small, Round}, ?, \text{Yes} \rangle$$

- $G$  unchanged:

$$G_2 = \langle ?, ?, ?, ?, ? \rangle$$

## Step 3: Example 3 (Negative)

- Example 3:  $\langle \text{Green, Large, Oval, Bitter, No, No} \rangle$
- Specialize  $G$  to exclude negative example while including  $S$ :

$$G_3 = \langle ?, \text{Small}, ?, ?, \text{Yes} \rangle$$

- $S$  remains unchanged:

$$S_3 = \langle ?, \text{Small}, \text{Round}, ?, \text{Yes} \rangle$$



## Step 4: Example 4 (Positive)

- Example 4:  $\langle \text{Red, Small, Oval, Sweet, Yes, Yes} \rangle$
- Update  $S$  to include this positive example (already partially generalized):

$$S_4 = \langle ?, \text{Small}, ?, ?, \text{Yes} \rangle$$

- $G$  remains:

$$G_4 = \langle ?, \text{Small}, ?, ?, \text{Yes} \rangle$$

# Final Version Space

- Most Specific Hypothesis:

$$S = \langle ?, \text{Small}, ?, ?, \text{Yes} \rangle$$

- Most General Hypothesis:

$$G = \langle ?, \text{Small}, ?, ?, \text{Yes} \rangle$$

- Version space narrowed down to: **Size = Small** and **Sweet = Yes**.