



# Machine Learning

## Genetic Algorithm

---

Lecturer: Duc Dung Nguyen, PhD.

Contact: [nddung@hcmut.edu.vn](mailto:nddung@hcmut.edu.vn)

Faculty of Computer Science and Engineering  
Hochiminh city University of Technology

The background of the slide is a complex network of white lines and nodes on a dark gray background. The nodes are represented by circles of varying sizes, and the lines connect them in a web-like structure. A prominent horizontal line connects two nodes in the upper left quadrant.

1. Introduction

2. Genetic Algorithm



# Introduction

---

- **Evolution** is known to be a successful, robust method of nature
- How do we **search** the **space of hypotheses** containing complex interacting parts, where the impact of each part on overall hypothesis is **hard to model**.
- Computer programs are evolved to **certain fitness criteria**.
- Evolutionary computation = Genetic algorithms + genetic programming

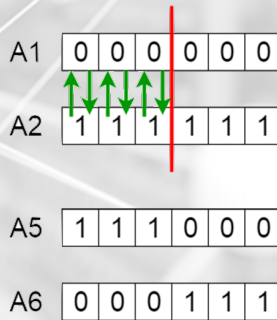
- **Learning as searching**
- Analogy to **biological evolution**
  - The best hypothesis is searched through **several generations** of hypotheses.
  - Next generation hypotheses are produced by **mutating** and **recombining** parts of the best current generation hypotheses
- It is not recommended to search from general-to-specific or from simple-to-complex hypotheses.



# Genetic Algorithm


---

# Genetic Algorithms



- **Initialize population:**  $P$  = randomly generated  $p$  hypotheses.
- **Evaluate fitness:** compute  $Fitness(h)$ , for each  $h \in P$
- While  $\max_{h \in P} Fitness(h) < Fitness_{threshold}$  do:
  - Create new generation
  - Evaluate fitness



- 
- The background of the slide features a complex network diagram. It is composed of numerous white circular nodes of varying sizes, interconnected by a web of thin white lines. Some nodes are significantly larger than others, and the connections form a dense, non-linear pattern across the entire slide area.
- Selection
  - Crossover
  - Mutation

- Selection:

Probabilistically select  $(1 - r)p$  hypotheses of  $P$  to add to the new generation.

The selection probability of a hypothesis

$$Pr(h_i) = \frac{Fitness(h_i)}{\sum_{h \in P} Fitness(h)}$$

- Crossover:
  - Probabilistically select  $(r/2)p$  pairs of hypotheses from  $P$  according to  $Pr(h)$
  - For each pair  $(h_1, h_2)$ , produce two offsprings by applying a Crossover operator.
  - Add all offspring to the new generation.

- Mutation:
  - Choose  $m$  percent of the added hypotheses with **uniform distribution**.
  - For each, **invert** one **randomly selected bit** in its representation.

- A classification rule as a bit string:

If  $expr(A_1) \wedge \dots \wedge expr(A_i) \wedge \dots \wedge expr(A_n)$  Then  $C = c$

- Example:


If *Wind = Strong* Then *PlayTennis = Yes*



1	1	1	1	0	1	0
---	---	---	---	---	---	---

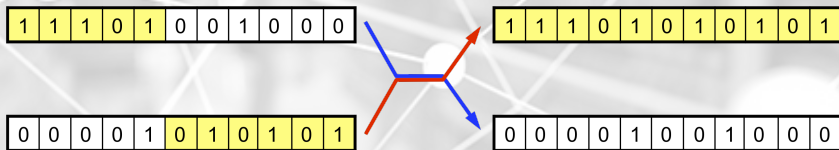
- A set of rules as concatenated bit strings:

1	1	1	1	0	1	0	1	0	0	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---

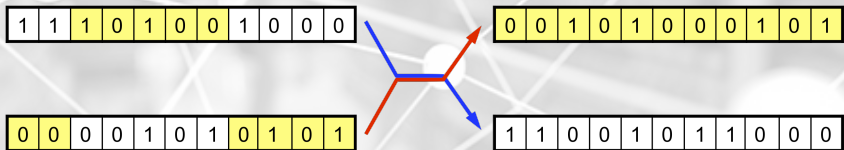
- 
- The background of the slide features a complex network diagram with numerous white nodes of varying sizes connected by thin white lines. A few nodes are highlighted with larger, solid white circles. A thick white horizontal line is drawn across the upper middle section of the network.
- Single-point
  - Two-point
  - Uniform



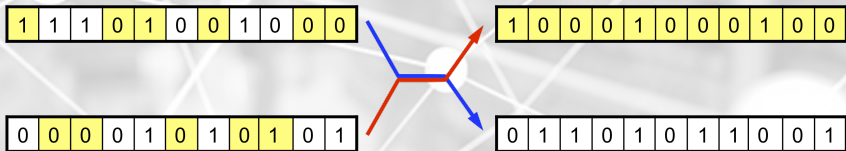
- Single-point



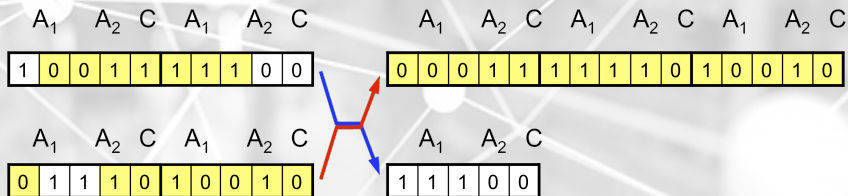
- Two-point



- Uniform



- Variable-length bit strings



- Example:

$$Fitness(h) = (correct(h))^2$$

$correct(h)$  = percent of all training examples correctly classified by hypothesis  $h$

- What is **inductive bias**?
- Where is inductive bias in GA?