Evolutionary Algorithms:
Estimation of Distribution Algorithms
A.K.A. Probabilistic Model-Building GAs

November 25, 2019

Prof. Chang Wook Ahn

Meta-Evolutionary Machine Intelligence (MEMI) Lab.
Electrical Eng. & Computer Sci.
Gwangju Inst. Sci. & Tech. (GIST)

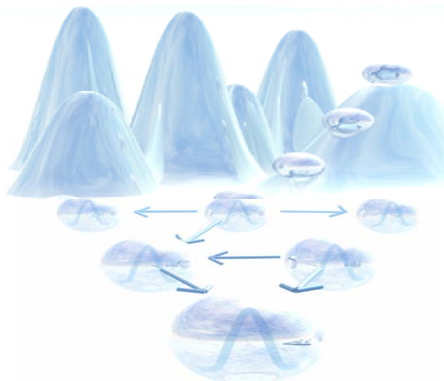
Estimation of Distribution Algorithms

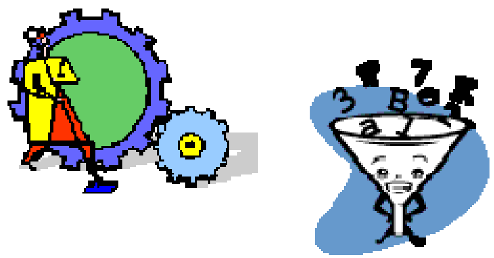
Background

❖ Estimation of Distribution Algorithms (EDAs)

- ➢ **A new paradigm** in terms of **intelligent optimization tools**
- ➢ **Cross-fertilization:** Evolutionary computation + (Probabilistic) Machine learning
- ➢ **Practical use:** Automatic learning, Robust & scalable performance

Evolutionary Computation → EDA (since 1998) ← Machine Learning





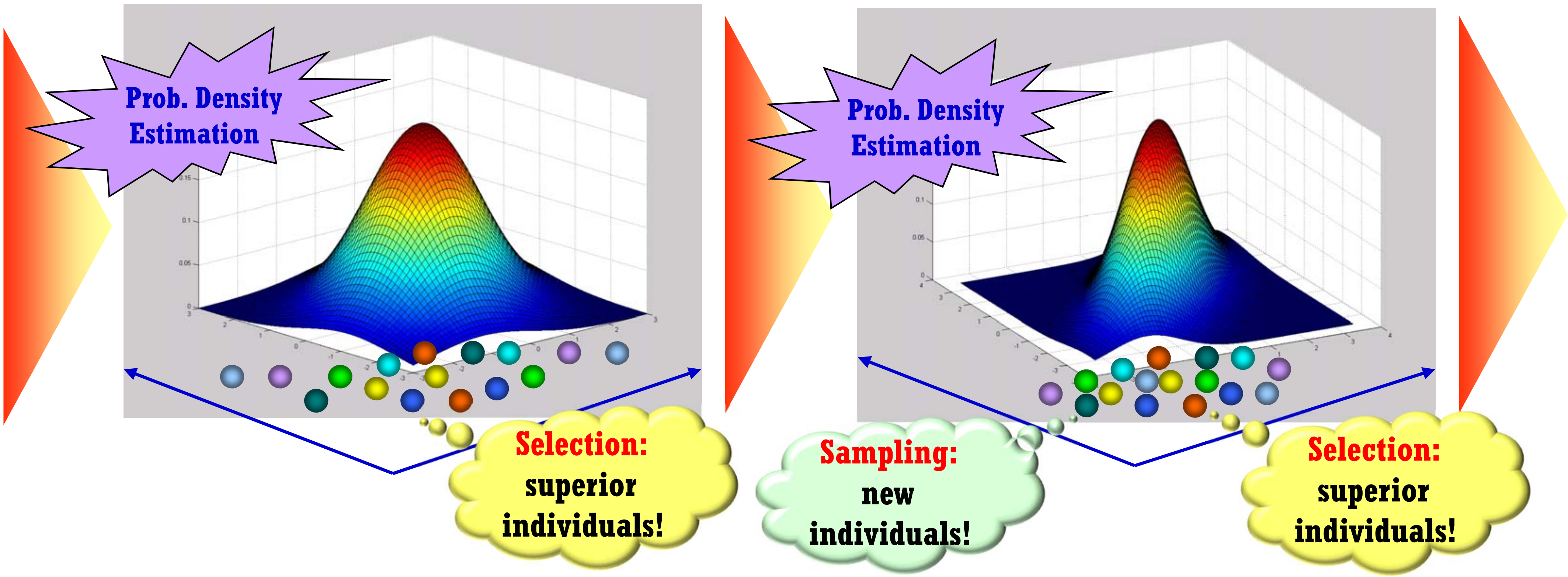
Evolutionary Computation	EDA (since 1998)	Machine Learning
• Ad hoc Operators • Great for Toy Problems • Not Scale Well • Powerful for Search • GAs, GP, ESs, …	• Multiple Search of EC • Problem Knowledge of ML • Intelligent Operators by EC and ML • Quadratic Scalability	• Efficient for Extracting Information from Data • Powerful for Learning & Recognition • BNs, GM, Classifier, …

Key Principle

❖ **Evolution of Probability Distribution of Population**

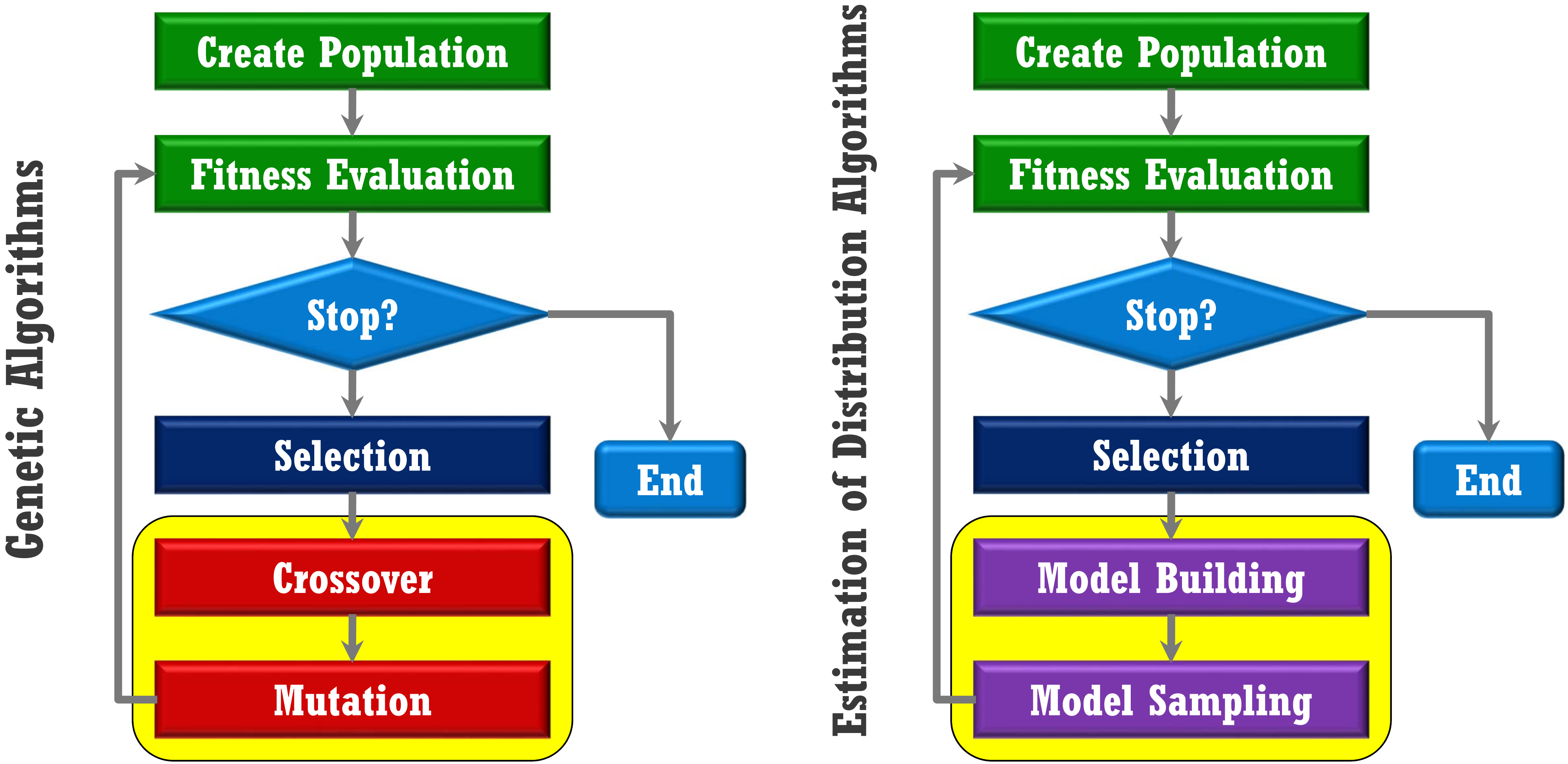
- **Selection:** Leads to probability density evolution
- **Sampling:** Generates new individuals
  - ✓ It replaces crossover and mutation!
  - ✓ But the probability density itself is identical!

☞ **Note:** Incorporating problem knowledge offers better performance.



GAs vs. EDAs

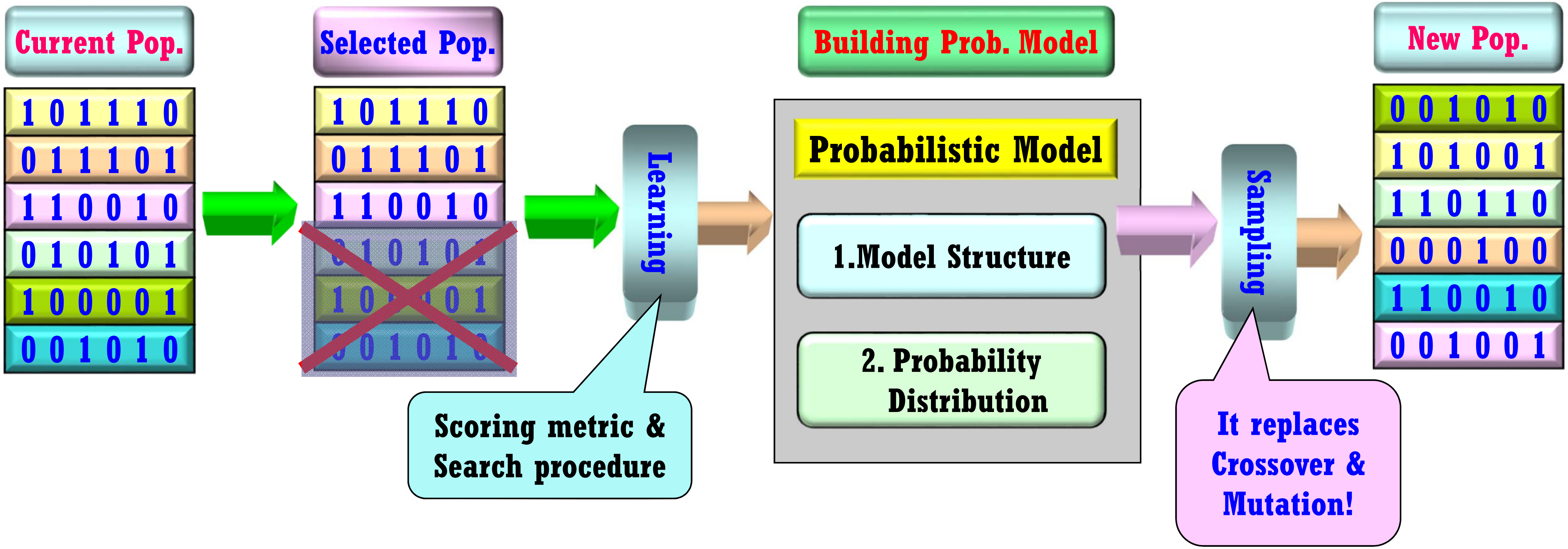
- **EDAs conduct GAs with Probability Distribution**
  - **GAs:** Crossover & Mutation are applied to explicit individuals
  - **EDAs:** Mixing process is conducted by sampling the *PDF* of population



General Procedures

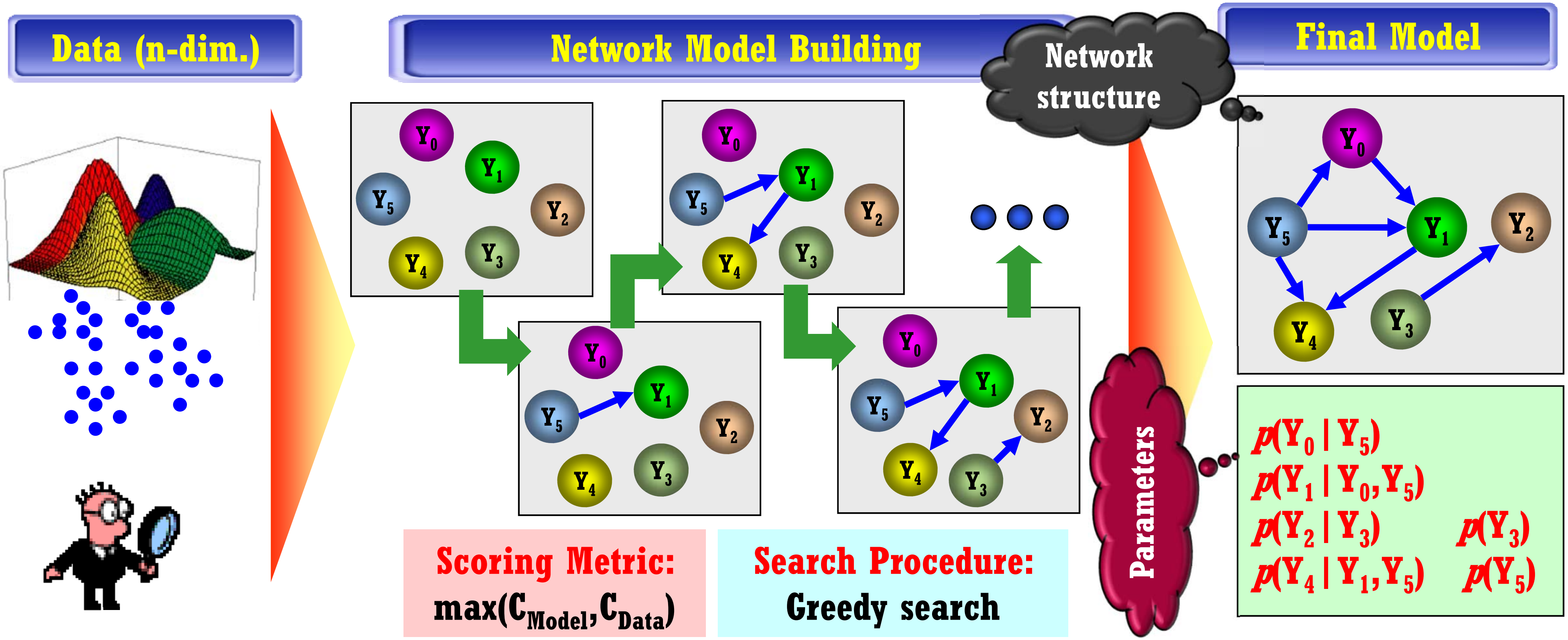
❖ Details of Estimation of Distribution Algorithms

- Powerful **Intelligent Stochastic Optimization Technique** inspired from both **Genetic Algorithms** and **(Probabilistic) Machine Learning**
- **Automatic identification** and **exploitation** of problem regularities
- Robust and **Scalable** performance on broad classes of challenging problems



Probabilistic Machine Learning

- **Graphical Model** (A Representative Probabilistic Machine Learning Technique)
  - **Representation:** Dependencies among variables by a structure
  - **Structure:** A directed acyclic graph that represents a factorization of variables
  - **Factorization:** A joint distribution by a product of conditional distributions
  - **Scoring metric** & **Search procedure** are required

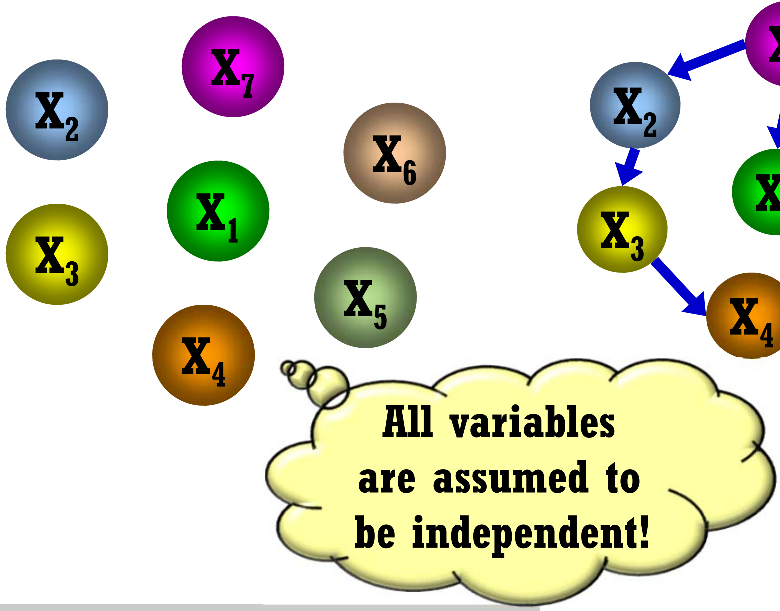


Classification

❖ Categories of EDAs

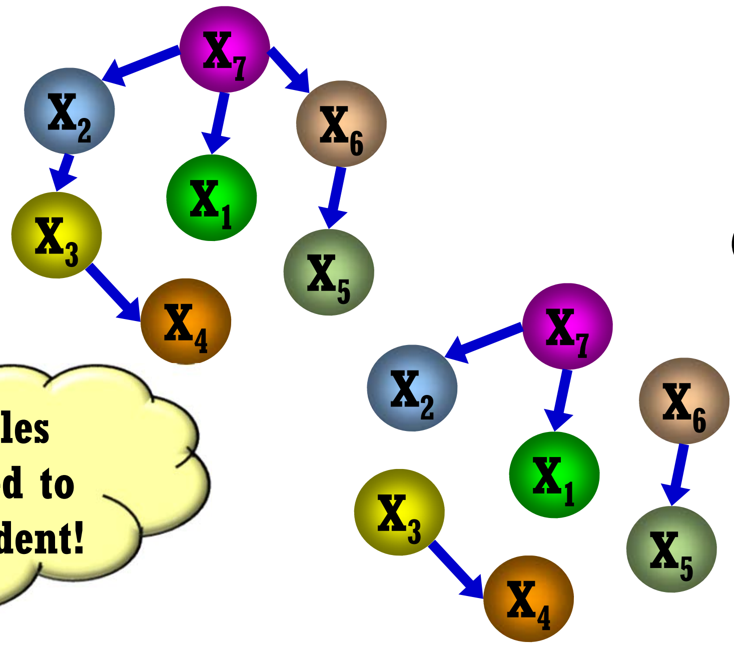
➢ **Three Classes** exist in accordance with **Model Dependencies**

Univariate Model



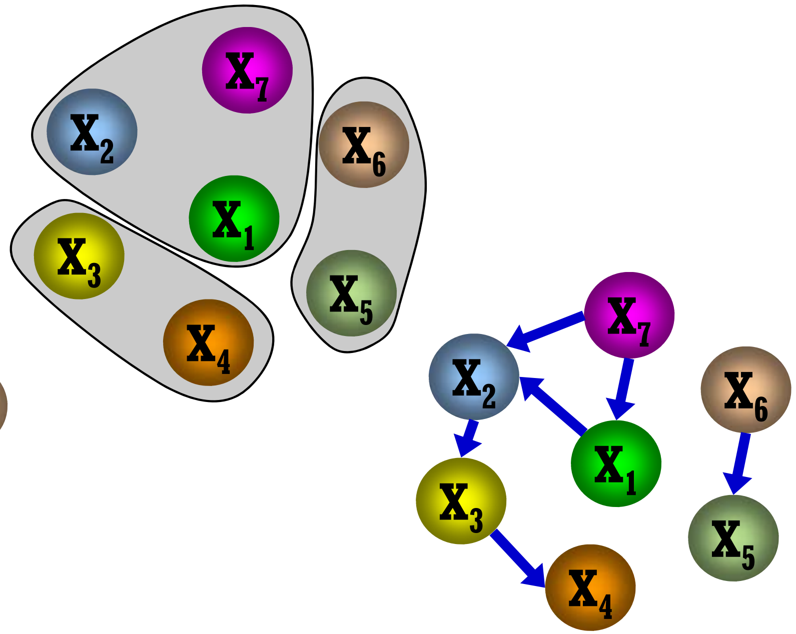
- PBIL (Baluja '95)
- UMDA (Muhlenbein '96)
- Compact GA (Harik '98)
- QEA (K.H. Han '02)

Bivariate Model



- MIMIC (DeBonet '96)
- COMIT (Baluja '97)
- BMDA (Pelikan '98)

Multivariate Model

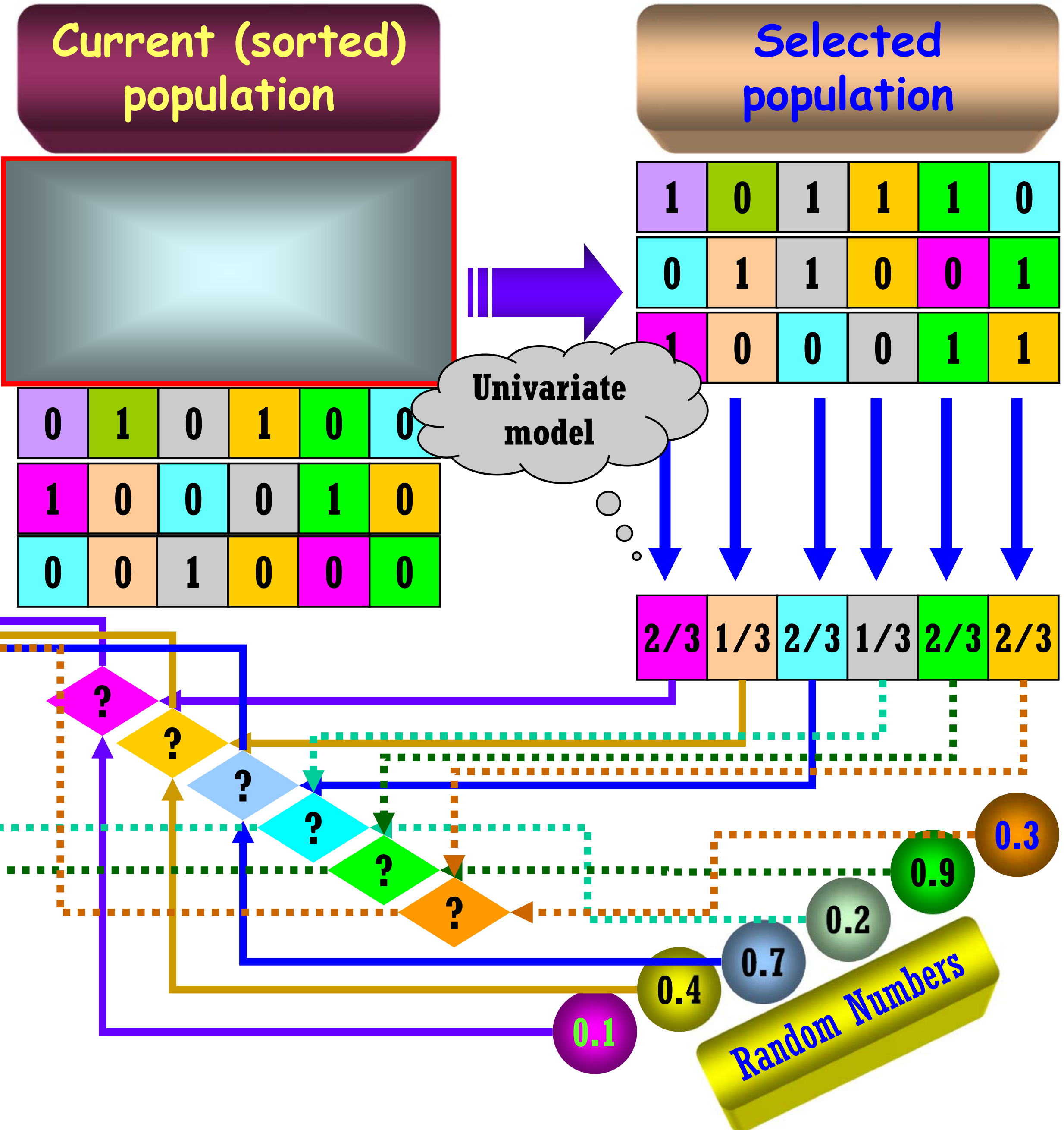


- FDA (Muhlenbein '99)
- ECGA (Harik '99) , mIDEA(Bosman '03)
- BOA (Pelikan '99), rBOA (C.W. Ahn '04)
- Mixed BOA (Ocenassek '02)

Example: UMDA

Simulation by Hand

- Population size, N = 6
- Individual length, l = 6
- Problem: OneMax (Optimum=111111)
- Probability model, $p = (p_1, \ldots, p_l)$



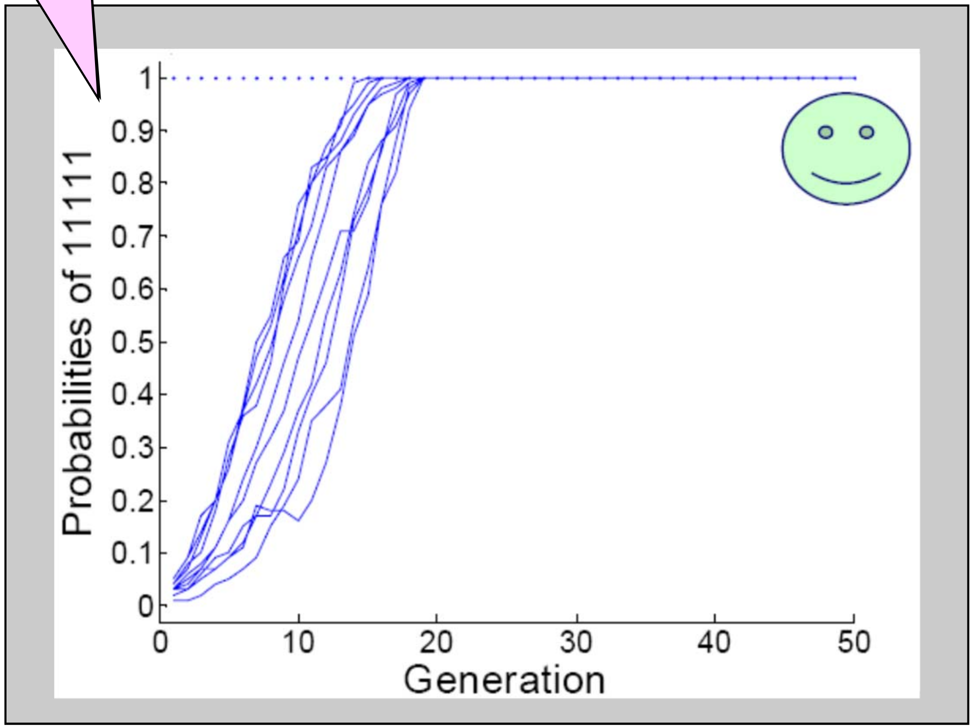
Towards Higher-Order Statistics

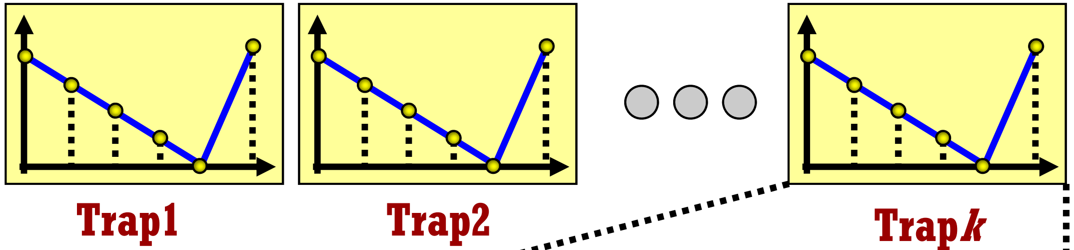
❖ Why Higher-Order Statistics?

- ➢ One-Max Problem: Optimum is '111…1'
  - ✓ Single bit '1' is better than '0' on average.
- ➢ Ex) 5-bit Trap Function: optimum is '11111'
  - ✓ But, f(1****)=1.375, f(0****)=2;
  - ✓ Single bits are misleading!
  - ✓ Thus, consider 5-bit statistics!

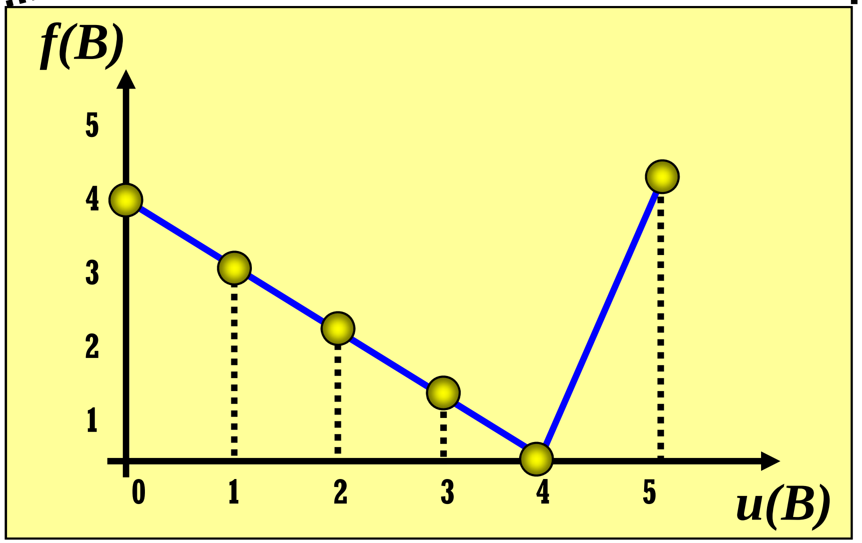
Ex) Artificial Deceptive Problem;
i.e., Concatenated Traps
$F(X) = \sum f(B)$

Via 5-bit statistics!





Ex) 5-bit Trap Function;
if $u(B) = 4$, then $f(B) = 5$;
else $f(B) = 4 - u(B)$;

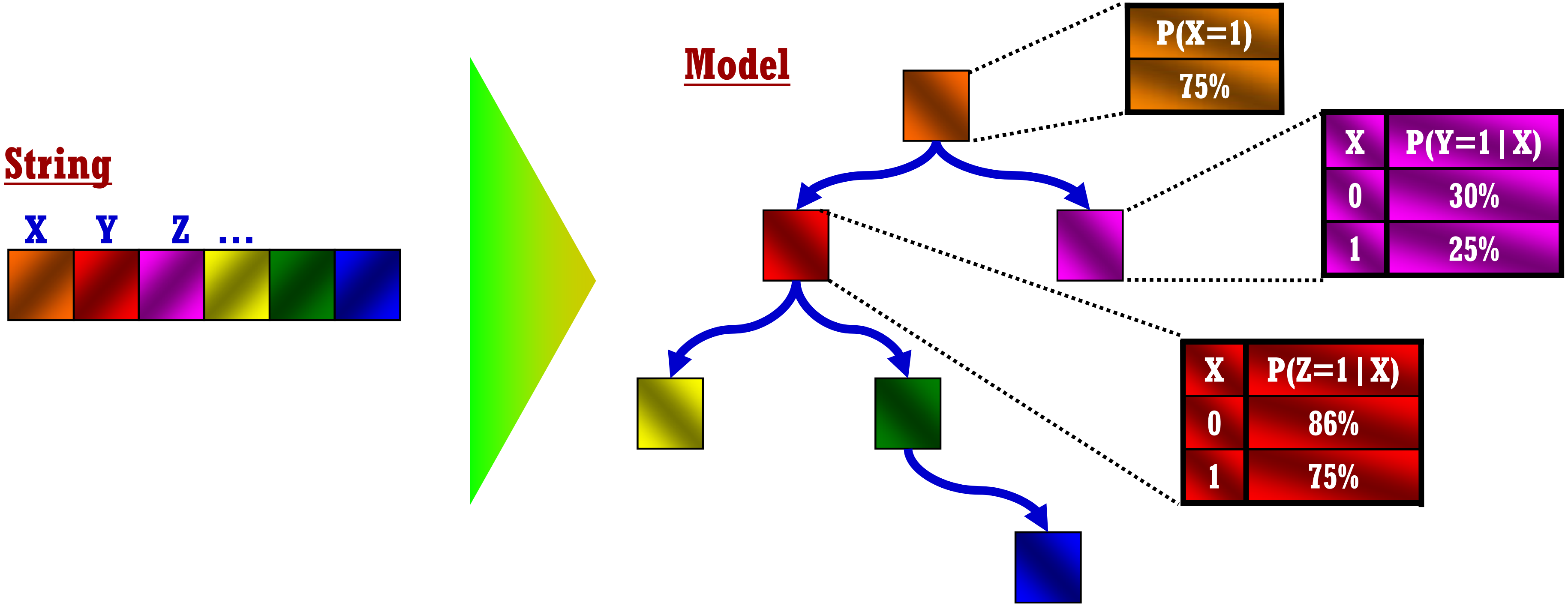


Example: COMIT

- **Combined Optimizers with Mutual Information Trees [Baluja&Davis'97]**
  - **Use a tree model**
    - **Find a tree that maximizes mutual information between connected nodes**

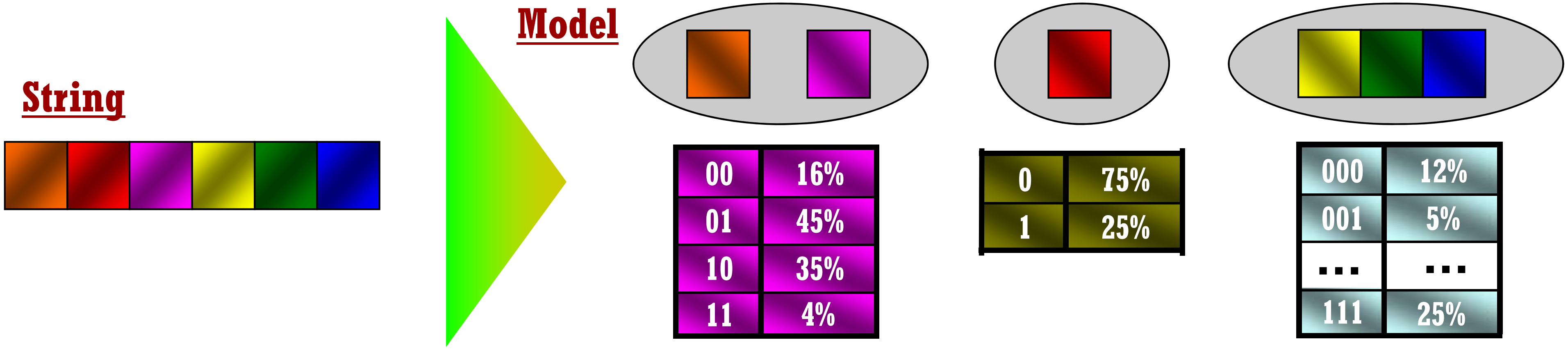
$$I(X_i, X_j) = \sum_{a,b} P(X_i = a, X_j = b) \log \frac{P(X_i = a, X_j = b)}{P(X_i = a) P(X_j = b)}$$

  - **Prim algorithm is used for finding a maximum spanning tree**



Example: ECGA

- **Extended Compact GA** [Harik'99]
  - Consider groups of genes
    - ✓ Use **Minimum Description Length** { $\log(N)\sum_I 2^{S[I]} + N\sum Entropy(M_I)$ }
  - **Greedy algorithm** is used for model search



00	16%
01	45%
10	35%
11	4%

0	75%
1	25%

000	12%
001	5%
…	…
111	25%

Selected population

[0]	[1]	[2]	[3]
1	0	0	0
1	1	0	1
0	1	1	1
1	1	0	0
0	0	1	0
0	1	1	1
1	0	0	0
1	0	0	1

Model	MDL
[0] [1] [2] [3]	44

→

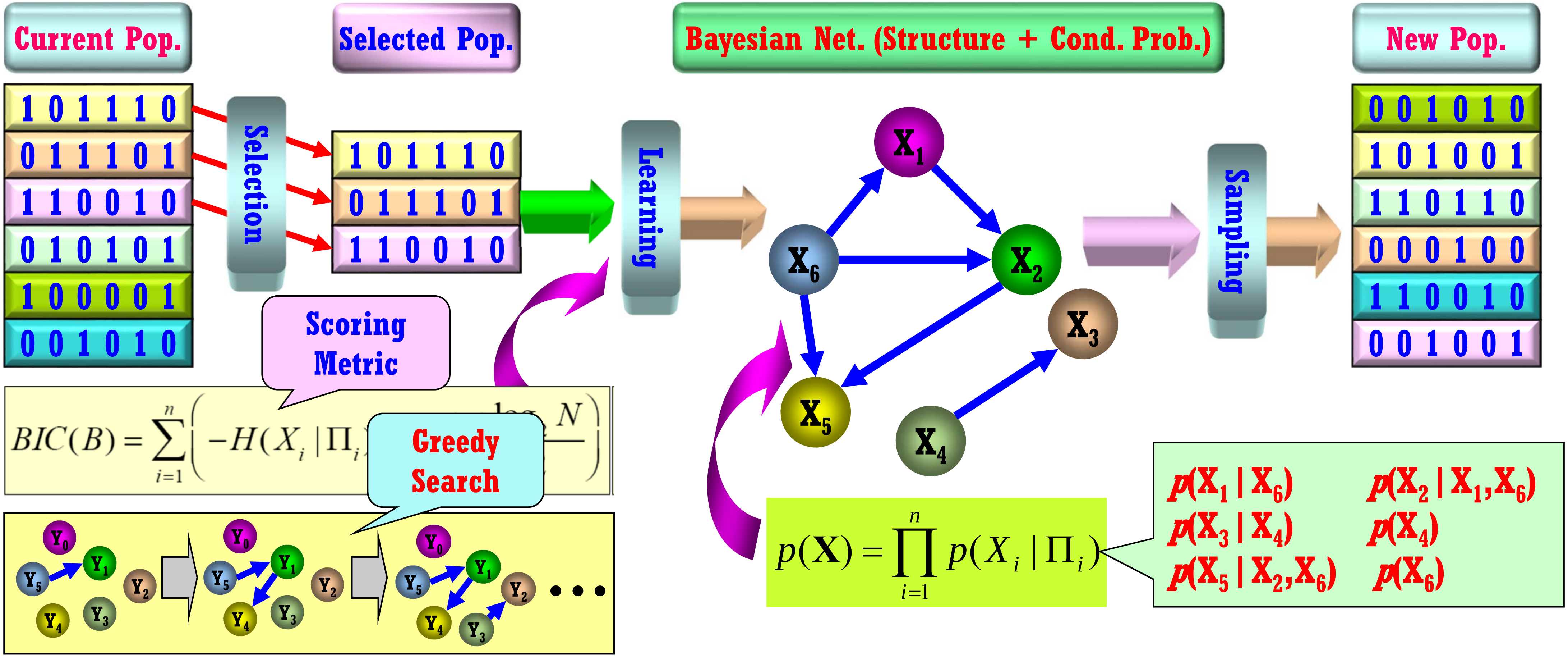
Model	MDL
[0 1] [2] [3]	46.7
[0 2] [1] [3]	39.8
[0 3] [1] [2]	46.7
[0] [1 2] [3]	46.7
[0] [1 3] [2]	45.6
[0] [1] [2 3]	46.7

→

Model	MDL
[0 2 1] [3]	48.6
[0 2 3] [1]	48.6
[0 2] [1 3]	41.4

Example: BOA

- **Bayesian Optimization Algorithm: $O(n^{1.55}) \sim O(n^2)$ [Pelikan'99]**
  - Exploiting multiple interactions using **Bayesian Networks**
    - ✓ **Learning the structure**: 1) Scoring metrics, 2) Search procedure
    - ✓ **Learning the conditional probabilities**
  - Sampling the Bayesian network based on forward simulation



Summary

- **Estimation of Distribution Algorithms (EDAs)**
  - A new paradigm of EAs: Evolutionary Computation + Machine Learning
  - They discover problem knowledge and then exploit it for evolution
  - They are very powerful for solving quite hard problems.
  - ☞ EDAs are the most notable research issue in EC community.