# Genetic Algorithms

**Dr. Mahmoud Nabil Mahmoud**
*mnmahmoud@ncat.edu*

North Carolina A & T State University

March 31, 2021

# Outline

# Introduction

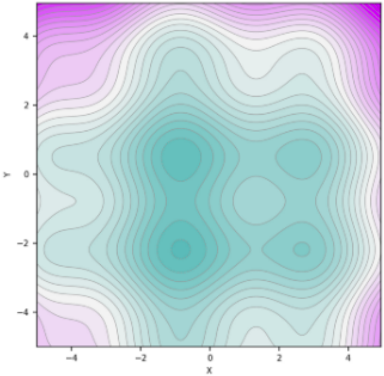
- PSO is originally attributed to Kennedy, Eberhart in 1995.
- It solves a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space using the position and velocity of particles.
- Inspired by the social behavior of birds
- Advantages
    - Very few hyperparameters.
    - Idea very similar to GA
    - It can be parallelized.

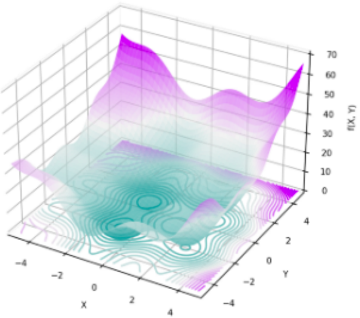
# Outline

1. Introduction
2. How it works?
3. Optimization

# Example

Function to minimize  2D and 3D view



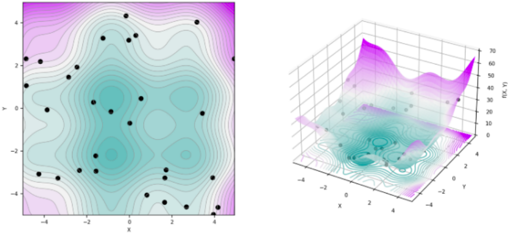
# Particles

First we define a group of particles (potential solutions) over the search space.

$$P_i^t = [x_{0,i}^t, x_{1,i}^t, x_{2,i}^t, \ldots, x_{n,i}^t]$$

- n is the number of dimensions
- t is the generation
- i is the index of the particle

All particles have fitness values evaluated by the fitness function to be optimized.



# Velocity

Each of these particles is in movement with a velocity allowing them to update their position over the iterations to find the global minimum.

$$V_i^t = [v_{0,i}^t, v_{1,i}^t, v_{2,i}^t, \ldots, v_{n,i}^t]$$

- n is the number of dimensions
- t is the generation
- i is the index of the particle

Note that, positions and velocities of particles are assigned randomly.

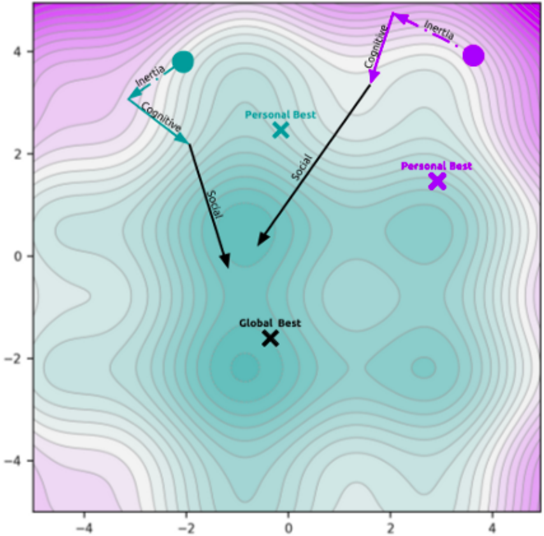
# Swarm

- Unlike GA, PSO has no evolution operators such as crossover and mutation
- Each particle is randomly accelerated towards:
  - its previous best position (personal best)
  - the best solution of the group (global best).
- Thus, the velocity is subject to inertia and is governed by the two best values found so far.

$$V_i^{t+1} = \underbrace{wV_i^{t+1}}_{\texttt{Inertia}} + \underbrace{c_1 r_1 \left( P_{best(i)}^t - P_i^t \right)}_{\texttt{Cognitive Personal}} + \underbrace{c_2 r_2 \left( P_{bestglobal}^t - P_i^t \right)}_{\texttt{Social Global}}$$

$$P_i^{t+1} = P_i^t + V_i^{t+1}$$

# Swarm


Inertia
Cognitive
Social
Personal Best
Personal Best
Global Best
4
2
0
−2
−4
−4
−2
0
2
4

# Outline

# Optimization

$$V_i^{t+1} = \underbrace{wV_i^{t+1}}_{\texttt{Inertia}} + \underbrace{c_1 r_1 \left(P_{best(i)}^t - P_i^t\right)}_{\texttt{Cognitive Personal}} + \underbrace{c_2 r_2 \left(P_{bestglobal}^t - P_i^t\right)}_{\texttt{Social Global}}$$

- $w \in \mathbb{R}^+$ : Inertia coefficient.
- $c_1 \mathbb{R}^+$ and $r_1 \in [0, 2]$ Cognitive coefficients.
- $c_2 \mathbb{R}^+$ and $r_2 \in [0, 2]$ Social coefficients.

- These coefficients control the levels of exploration and exploitation.

# Effects of Coefficients

- A low coefficient w facilitates the exploitation of the best solutions found so far
- A high coefficient w facilitates the exploration around these solutions.
- Note that it is recommended to avoid $w > 1$ which can lead to a divergence of our particles.

# Effects of Coefficients

When c1,r1 are high and c2,r2 are low:

- Swarm are more individualistic
- Therefore, no convergence

When c2, r2 are high and c1,r1 are low:

- Swarm are more more influenced by the others.
- May converge to local minima.

The coefficients c1 and c2 are consequently complementary. A combination of the two increases both exploration and exploitation.

# Auto hyperparameters

Coefficients are usually updated automatically over the iterations.

$$
\begin{gathered}
w^t = 0.4\frac{t-N}{N^2} + 0.4 \\
c_1^t = -3\frac{t}{N} + 3.5 \\
c_2^t = +3\frac{t}{N} + 0.5
\end{gathered}
$$

Starting with a strong c1, strong w, and weak c2 to increase the exploration of the search space, we want to tend towards a weak c1, weak w, and strong c2 to exploit the best results after exploration by converging towards the global minimum.

# References


- Goldenberg, D.E., 1989. Genetic algorithms in search, optimization and machine learning.
- Michalewicz, Z., 2013. Genetic algorithms + data structures= evolution programs. Springer Science & Business Media


Questions