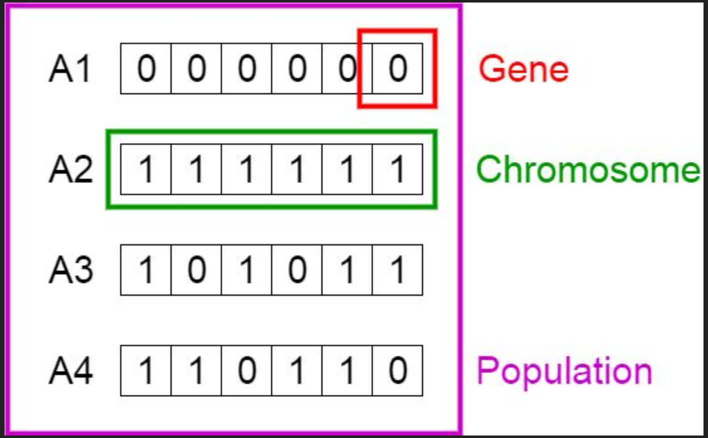
Genetic Algorithms

Sean Peng, Ron Guter, Quintin Buckley

What are Genetic Algorithms (GAs)?

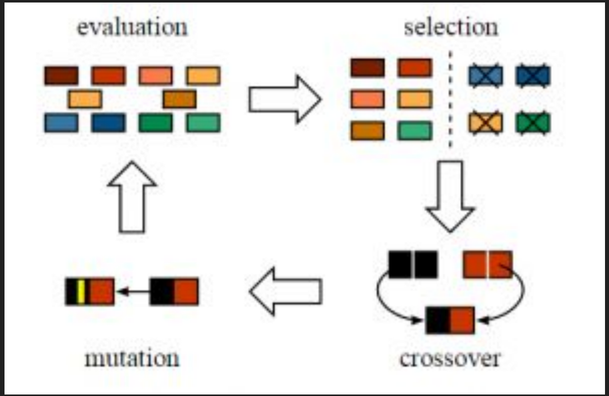
- Method of solving optimization problems, inspired by natural selection
- A ***heuristic*** algorithm, not ***analytic***
- Each candidate solution is an array called a ***chromosome***
- The initial ***population*** is randomly generated
- The solutions improve over ***generations***



Four Stages of GA

For each generation:

- **Evaluation:** calculate a score for each chromosome
- **Selection:** the best chromosomes become parents
- **Crossover:** the parents combine and reproduce
- **Mutation:** random change of part of a chromosome at a very low probability (flipping a bit).



Evaluation

- Calculate the ***fitness*** for each chromosome
- Ex. For an algorithm where each chromosome has 2 parameters, a fitness function could be

$$f(p_1, p_2) = \pi^{p_1} + p_2$$

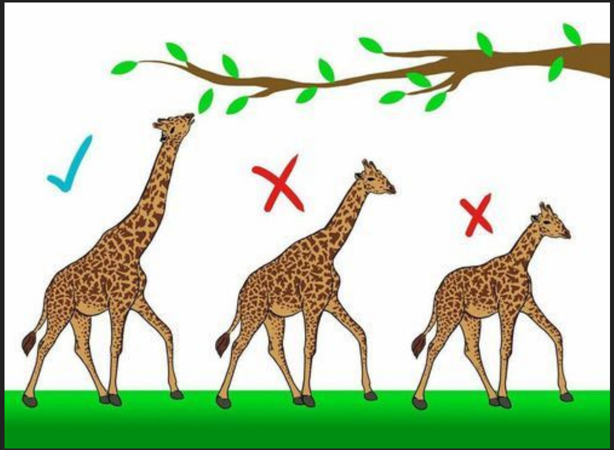
- For maximizing, higher fitness indicates a better solution
- For minimizing, lower fitness indicates a better solution
- **Better solutions dictate the general direction of the evolution process**

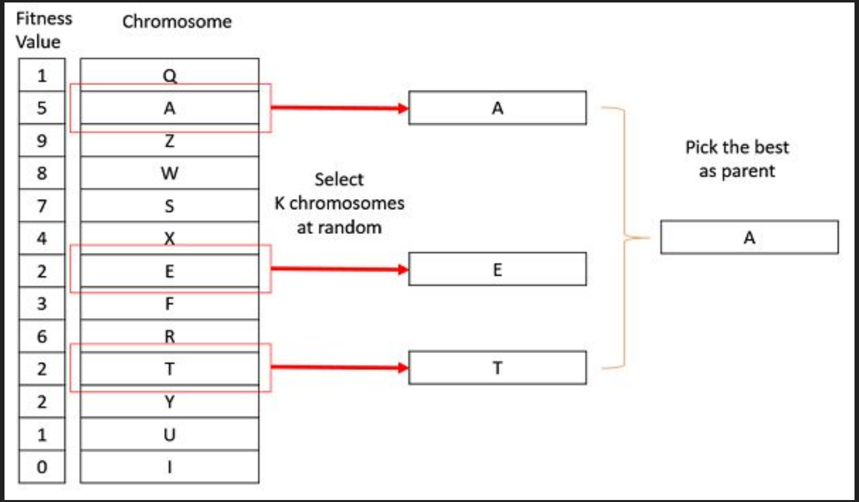
Selection

- Some high fitness chromosomes in the current generation become parents
- Many different ***selection methods***
- Roulette Wheel Selection (for the ith chromosome)

  - $$\text{prob. of selection}_i = \frac{\text{fitness}_i}{\sum_{j=0}^{n} \text{fitness}_j}$$

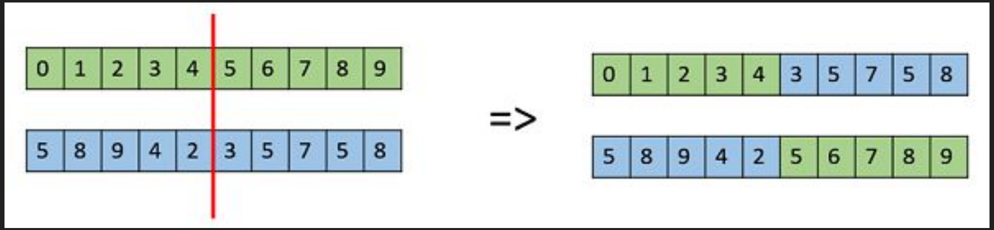
- Tournament Selection
  - Break population into groups of k
  - The fittest of each group is selected

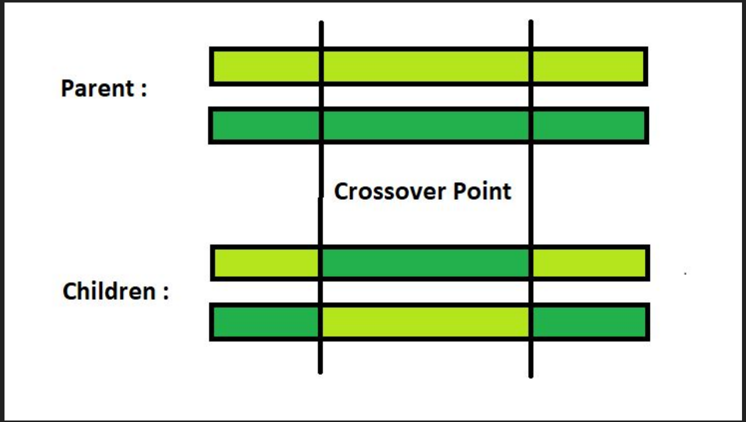




Crossover

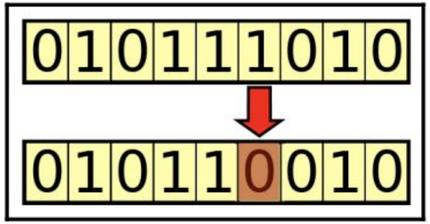
- Selected parents will "reproduce" by swapping segments of their genes, based on randomized crossover point(s).
- Ex. P1 = 11111, P2 = 00000, one crossover point at bit 2
    - Child 1 = 11000, Child 2 = 00111
- Potentially combine the best features of the parents





Mutation

- Randomly flips bits in a chromosome at a very low probability
- Inspired by nature, maintains genetic diversity
- Prevents the algorithm from getting stuck at local optima before finding the global optimum
    - Advantage over other algorithms, e,g. gradient descent



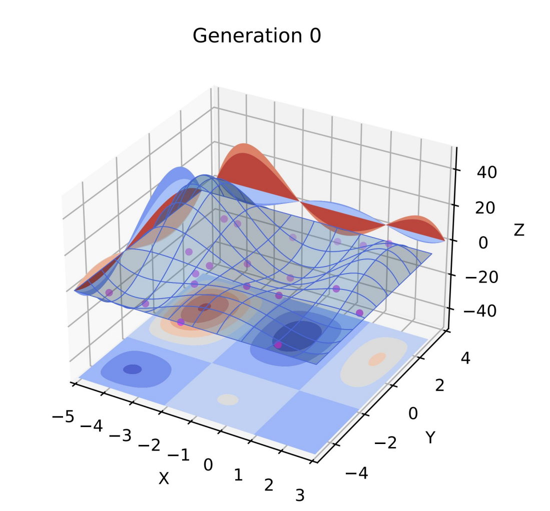
Mutation operator applied to a binary-coded chromosome

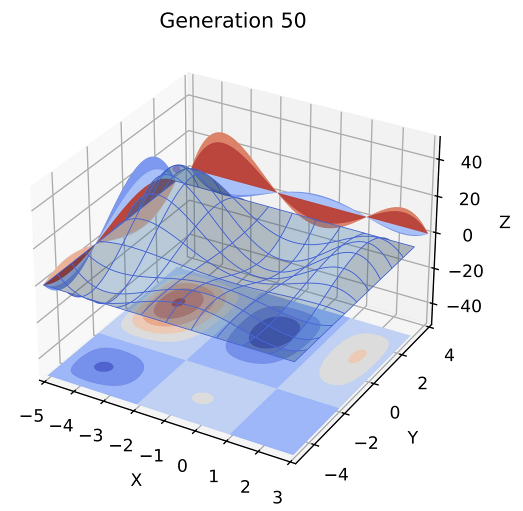
Algorithm 1 GA

$g = 0$ ▷ Initialize generation 0
P_g = population of n randomly generated chromosomes
Compute $f(p_i)$ for all $p_i \in P_g$ ▷ Compute initial fitness
while population has not converged **do**
    Selection
    Crossover
    Mutation
    $g = g + 1$
    Compute $f(p_i)$ for all $p_i \in P_g$
return fittest chromosome in P_g

Simple Function Optimization

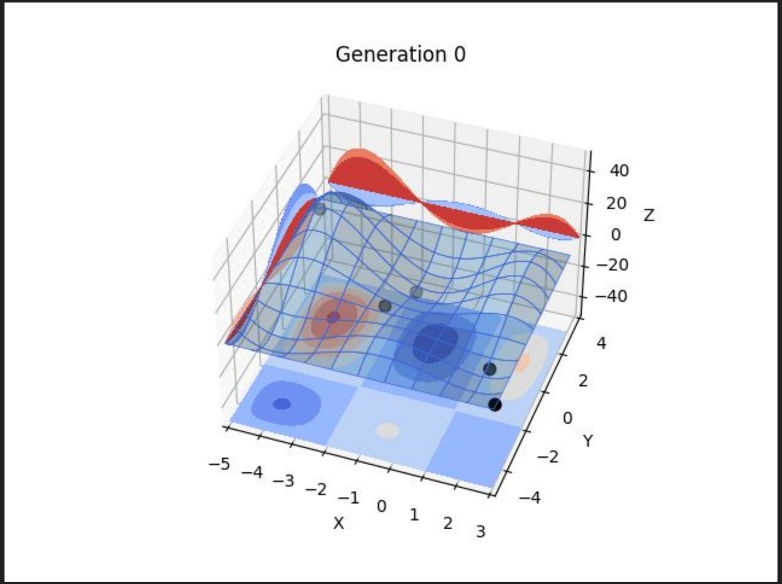
- Maximize $f(x,y) = 0.01(x-1)(x+2)(x+5)(x-3)(y-3)(y+2)(y+5)$
  - Constrained domain: $x, y \in [-5, 3]$
  - This is our fitness function
- Population size: 20 chromosomes
- Tournament selection: k = 3, Crossover rate = 0.8, Mutation rate = 0.1





Improving GA

- Genetic algorithms follow a simple set of rules
- This makes them easier to understand and modify
- What if we combined GA with other kinds of optimization methods?



Particle Swarm Optimization (PSO)

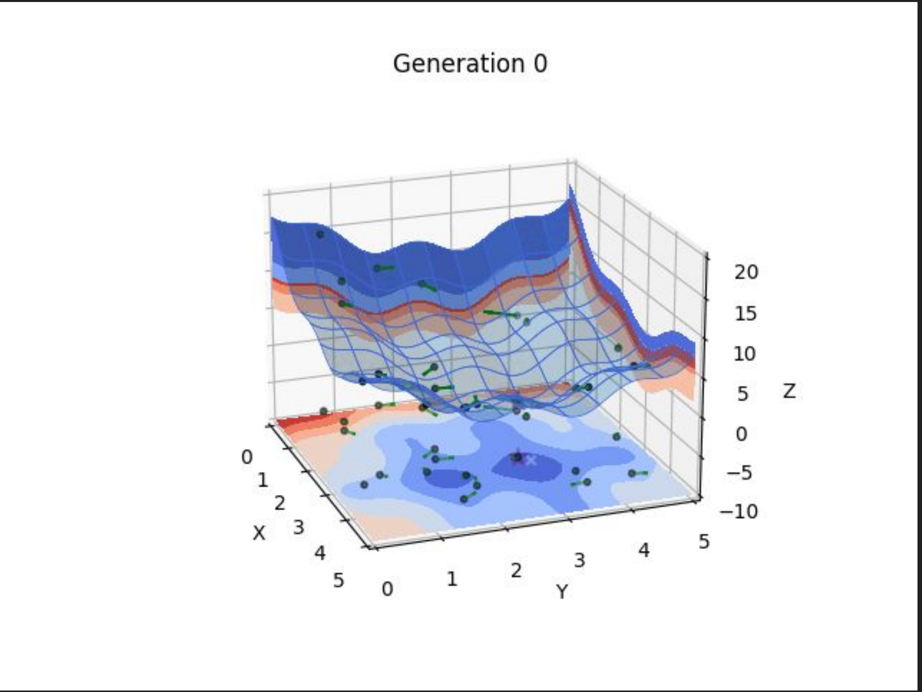
- Another ***heuristic*** algorithm
- Instead of “chromosomes,” we have a “swarm” of random “***particles***”
- Each particle starts with a random initial ***velocity*** in a random direction
- At every iteration (t), every particle (X_i) steps in the direction of its velocity
- At every iteration, every particle calculates its new velocity (V_i) using the equation

$$V_i(t+1) = \omega \cdot V_i(t)$$
$$+c_1\phi_1(\mathrm{pbest}_i - X_i(t)) + c_2\phi_2(\mathrm{gbest} - X_i(t))$$

* ω, c_1, and c_2 are preset hyperparameters on [0, 1]
* ϕ_1 and ϕ_2 are randomly selected numbers on [0, 1]

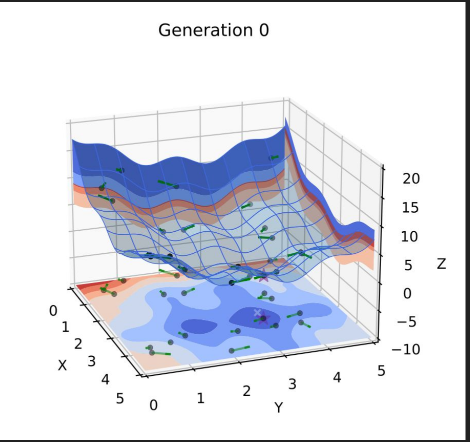
What does this mean?

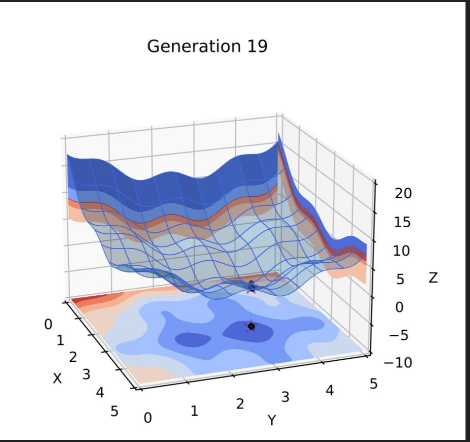
- Every particle tracks its own best position over time ($pbest_i$) and the swarm's overall best (gbest)
- At every step, a particle's velocity will gradually shift towards both its ***personal best*** and the swarm's ***global best***
- We both "***explore***" a particle's individual results and "***exploit***" the knowledge of the entire swarm



A Hybrid of Genetic Algorithms and Particle Swarm (HGAPSO)

- Apply the principles of ***exploration*** and ***exploitation*** to a genetic algorithm
- Give each chromosome a velocity, and for every generation:
  1. Step ("***enhance***") each chromosome in the direction of its velocity
  2. Compute every chromosome's new velocity with the same equation as in PSO
  3. Apply selection, crossover, and mutation to these "***enhanced chromosomes***"
  4. Repeat





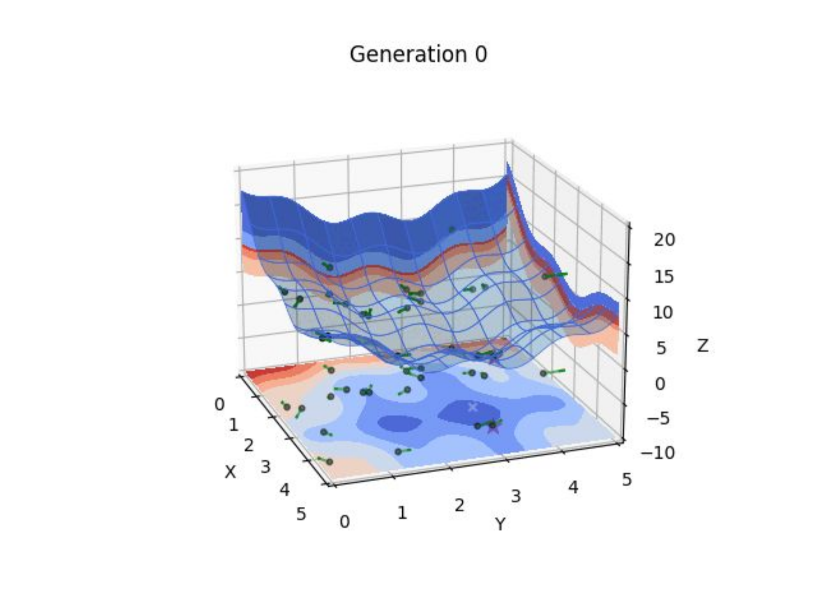
Standard GA vs. HGAPSO

- Both algorithms tested on the function

$$f(x, y) = (x - 3.14)^2 + (y - 2.72)^2 + \sin(3x + 1.41) + \sin(4y - 1.73)$$

$$x, y \in [0, 5]$$

- On average, HGAPSO was able to find a solution within one-hundredth of the true global minimum in fewer generations than the standard GA could
- Even better results could be achieved by finding optimal hyperparameters



Application to Portfolio Optimization

- A ***portfolio*** is a partition of the total capital, with each portion is allocated to a different asset
- We can build an optimal portfolio based off some metric (fitness function) using GAs
- We used data sets from Yahoo Finance to get the stock prices of companies from 2019–2021.

	ADBE	BTC-USD	IRX	AMZN	TXN	MSFT	NVDA
Date							
2019-05-31	270.899994	8574.501953	2.293	1775.069946	98.278275	121.004944	33.731499
2019-06-01	270.899994	8564.016602	2.293	1775.069946	98.278275	121.004944	33.731499
2019-06-02	270.899994	8742.958008	2.293	1775.069946	98.278275	121.004944	33.731499
2019-06-03	259.029999	8208.995117	2.280	1692.689941	99.088554	117.247993	33.313152
2019-06-04	268.709991	7707.770996	2.295	1729.560059	102.810135	120.496185	35.609070
...	...	...	...	...	...	...	...

Defining the Stages

- **Chromosome:** each chromosome is a partition that sums to 1 (the total capital)
- **Fitness function:** expected return
- **Selection:** top third of the current generation
- **Crossover:** Random constant k, the computation $k \cdot a + (1 - k) \cdot b$ for parents a, b
- **Mutation:** Randomly transferring capital from one asset to another

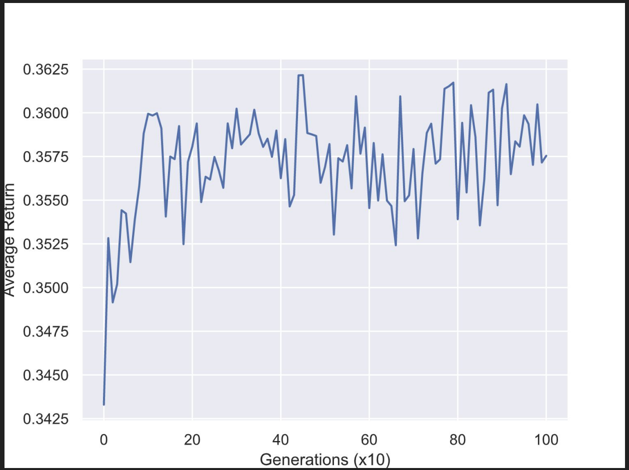


TABLE II
BEST CHROMOSOME FOR MAX RETURN

Stock	Weight for Max Return
AAPL	0.12
ADBE	0.01
AMZN	0.06
BTC-USD	0.07
FB	0.12
GC=F	0.05
IRX	0.17
MSFT	0.21
NVDA	0.06
QCOM	0.03
TSLA	0.03
TXN	0.08

Improvement Using the Sharpe Ratio

- Original fitness function focused on returns, but doesn't account for risk
  - resulted in high volatility
- Better metric: Sharpe ratio (reward-to-variability ratio)
  - $$\frac{\text{Expected Return} - \text{Risk-free Rate}}{\text{Std. of returns}}$$
- Achieved 0.5738 Sharpe ratio with 6.21% returns
  - > 0.5 means well-diversified portfolio