

# Algoritmo de Evolución Diferencial

Majo Gil

Joshua Chicoj

# Evolución Diferencial

01

Es un algoritmo de evolución de optimización, que resuelve los problemas mediante la iteración y mejora de una posible solución.

02

A diferencia del algoritmo de Descenso Gradiente, el problema no necesita ser diferenciable.

03

Cabe resaltar que el resultado propuesto no necesariamente es el mínimo global del problema.

04

Se utiliza principalmente para resolver problemas no lineales, multiacotados de optimización

# Generalidades

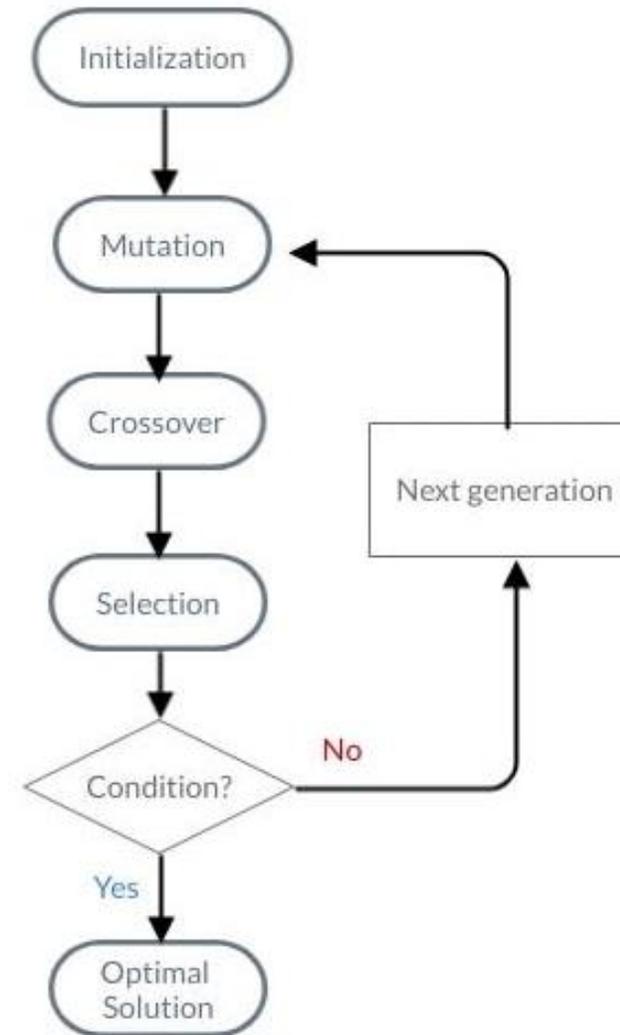
✓ Es parte de los algoritmos evolutivos

🧬 Es similar al algoritmo genético

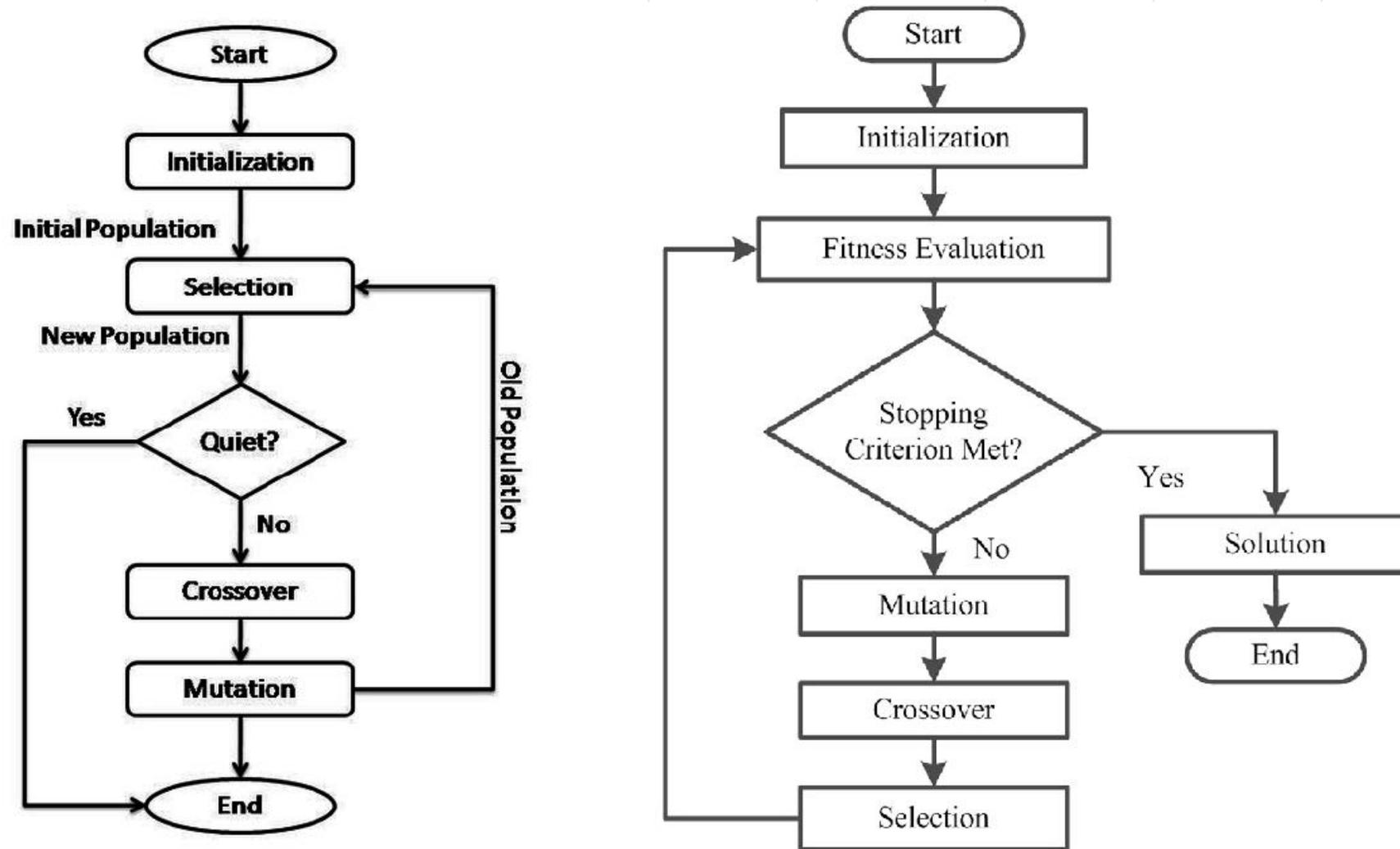
👑 Es de tipo “greedy”

# Cómo funciona

- DE optimiza un problema creando una población, mutándola, haciendo un cross over de ellas mismas, y seleccionando las mejores opciones de acuerdo a una fitness function



# Genético vs ED



# Pasos

## Inicialización:

- La población inicial es generada de manera aleatoria y cubre el espacio completo

## Mutación:

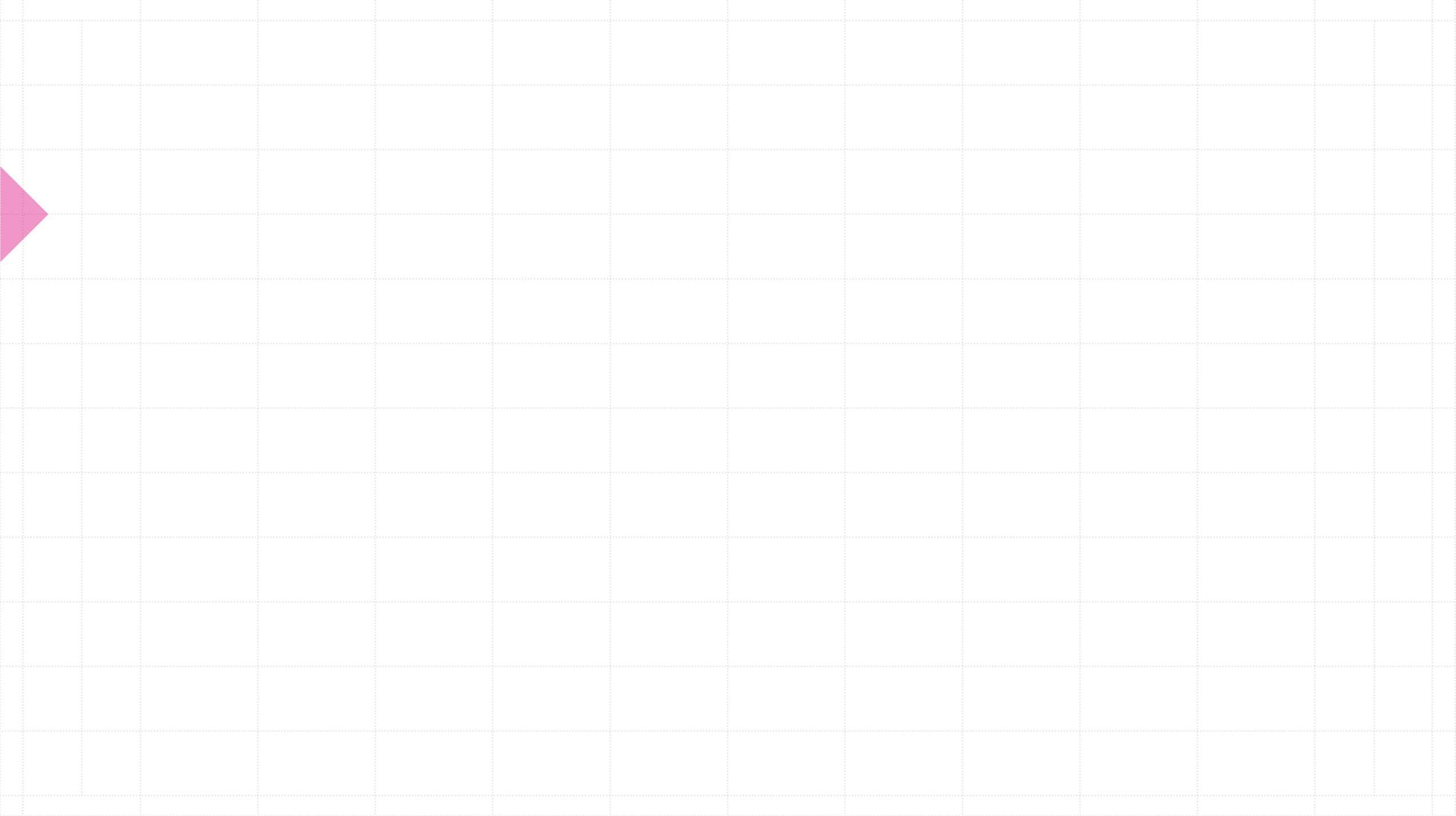
- Por cada individuo (vector), se seleccionan otros 3 individuos de manera aleatoria, y se calcula el vector mutante de la siguiente manera
- $v = x_3 + F(x_1 - x_2)$  donde  $F$  es un factor entre 0 y 1 que controla la amplificación de la mutación y  $x_1 \neq x_2 \neq x_3$  y pertenecen al espacio.

## Crossover

- Se utiliza un factor de probabilidad para determinar qué elementos de los vectores se cruzan para formar uno nuevo, así como cuántos vectores originales permanecen en la población.

## Selección

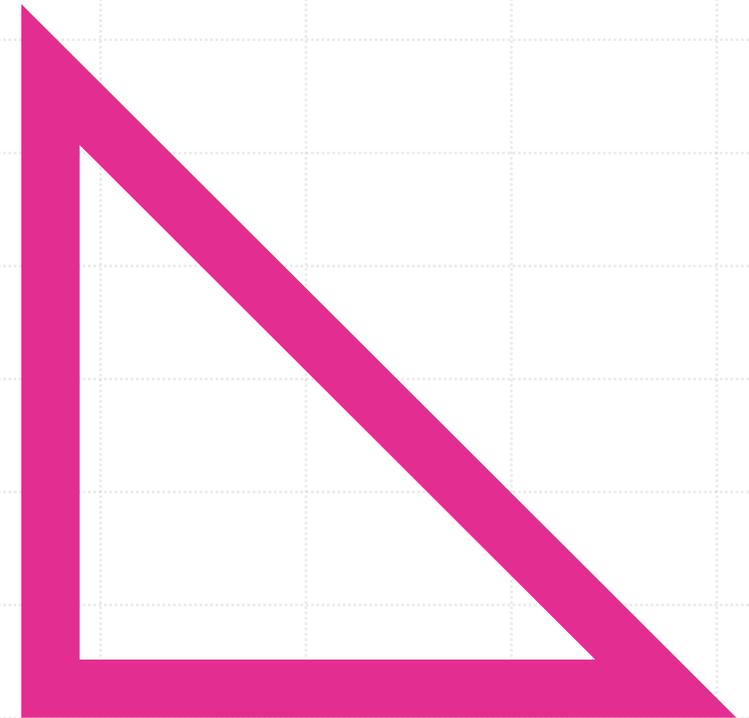
- Los integrantes de la población son comparados con la función objetivo y se seleccionan los mejores





# Condiciones de frontera

- Veamos que los nuevos elementos de la población generados luego del crossover, deben permanecer dentro del espacio del problema. Para asegurarse de esto, cualquier parámetro que infrinja estas condiciones de frontera es reemplazado de manera aleatoria por valores que pertenezcan a este rango



# Restricciones y penalizaciones

- En los ED es usual utilizar funciones de penalización para volver un problema con restricciones un problema de optimización sin restricciones. Una de las maneras más comunes de penalizar a un individuo es que el peso de la penalización sea la suma de las violaciones a las condiciones cometidas





# Ejemplos



# Referencias

Direct, S. (ed.) (no date) *Differential Evolution Algorithm*, *Science Direct*. Available at: [https://www.sciencedirect.com/topics/computer-science/differential-evolution-algorithm#:~:text=The%20differential%20evolution%20\(DE\)%20algorithm,to%20the%20environment%20are%20preserved](https://www.sciencedirect.com/topics/computer-science/differential-evolution-algorithm#:~:text=The%20differential%20evolution%20(DE)%20algorithm,to%20the%20environment%20are%20preserved). (Accessed: 21 November 2023).

R. Shaji, "Differential Evolution : What it is and how does it work?," Medium, [https://medium.com/@reshma\\_shaji/differential-evolution-what-it-is-and-how-does-it-work-81d3415c2367](https://medium.com/@reshma_shaji/differential-evolution-what-it-is-and-how-does-it-work-81d3415c2367) (accessed Nov. 21, 2023).

Wikipedia Contributors, "Differential evolution," *Wikipedia*, Nov. 14, 2019. [https://en.wikipedia.org/wiki/Differential\\_evolution](https://en.wikipedia.org/wiki/Differential_evolution) (accessed Nov. 21, 2023).