

# TP2: Algoritmos genéticos

## Grupo 2

Tomás Álvarez Escalante (60127)  
Alejo Francisco Caeiro (60692)  
Lucas Agustín Ferreiro (61595)  
Román Gómez Kiss (61003)



- Tabla de contenidos



## 01 **ASCII drawer**

Resolución mediante  
algoritmos genéticos



## 02

### **Color searcher**

Estructura, Métodos  
de selección, crusa,  
mutación y criterios  
de corte



## 03

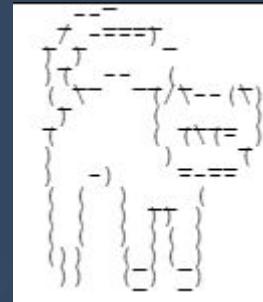
### **Conclusiones**

Datos y gráficos



# 01

## • ASCII drawer •



# • Introducción

Dada una imagen cuadrada, queremos implementar, mediante algoritmos genéticos, su representación en un mapa de caracteres ASCII de NxN.

Por ende planteamos el siguiente flujo de operaciones para obtener la solución que mejor aproxime la imagen.

# 1. Codificación del genotipo

Cada individuo en la población tendría un cromosoma que representaría la posible solución.

En este caso, el genotipo podría consistir en una **matriz de caracteres ASCII de tamaño NxN**, donde cada elemento de la matriz representa un carácter ASCII que se utilizará en la representación de la imagen.

## 2. Evaluación de la aptitud

Se evaluará la aptitud de cada individuo en la población mediante una función de evaluación de aptitud.

La función de fitness se encargará de **comparar los valores de los píxeles de la imagen original con los valores de los píxeles correspondientes en el mapa ASCII generado**. Cuanto menor sea la diferencia entre los valores, mayor será la aptitud de la solución.

**Ejemplo:** si el valor de un píxel en la imagen original es 100 y el valor correspondiente en el mapa ASCII es 97, la diferencia absoluta sería 3.

### 3. Inicialización de la población y selección

Se generarían una serie de individuos iniciales de forma aleatoria, cada uno con su propio cromosoma que representa una posible solución, luego se seleccionarán los individuos más aptos para la reproducción en función de su puntuación de aptitud.

La técnica de selección que utilizaríamos sería la **selección por torneo**, debido a que es simple y eficaz.

## 4. Cruza

Al momento de cruzar los padres para generar nuevos hijos, el método de **cruza de un solo punto** es una buena opción en nuestro caso porque es simple, preserva características importantes, introduce diversidad y es compatible con otros métodos para mejorar la diversidad en la población.



## 5. Mutación y recombinación

Se introduce una pequeña probabilidad de mutación en los hijos para introducir variabilidad en la población. Para este problema, se podrían cambiar aleatoriamente algunos caracteres en la cadena ASCII del hijo, utilizando el **método de mutación multigen limitado o uniforme**.

Luego se crea una nueva población a partir de los hijos generados y los miembros no seleccionados de la población actual.

Debido a que el método de recombinación **new over actual** prioriza tomar los hijos consideramos que es mejor dicho método en nuestro caso

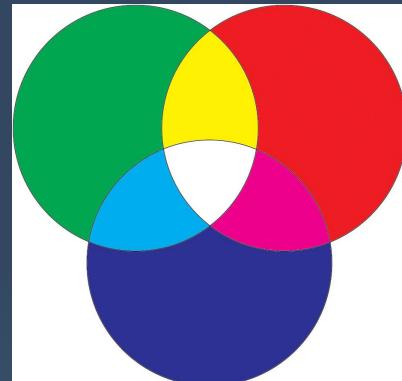
## 6. Convergencia y resultado

Los procesos de selección, crusa, mutación y recombinación planteados previamente se repetirán durante un número determinado de generaciones o hasta que se cumpla algún criterio de convergencia, como alcanzar una **aptitud objetivo** o no observar mejoras significativas en la aptitud de la población en un número determinado de generaciones.

Una vez que el algoritmo genético ha convergido, se obtendría la mejor solución encontrada, que sería la representación de la imagen en el mapa de caracteres ASCII correspondiente al cromosoma del individuo con la **aptitud más alta** en la última generación.

# 02

## • Color Searcher •



# • Introducción

Queremos ayudar a los artistas a lograr obtener un **color objetivo** con una paleta de colores finita

Para ello se implementó un sistema que, mediante **algoritmos genéticos**, mezcla proporciones de los diferentes colores de la paleta, a fin de encontrar el color deseado

# • Estructura del individuo



**Gen:** proporción de cada color de la paleta

$$\xrightarrow{\hspace{1cm}} \# \text{Genes} = \# \text{Colores en la paleta}$$

**Genotipo:** representación codificada de la información, es decir, una cadena de genes

**Alelos:** diferentes posibles proporciones en un determinado locus

Ejemplo con una paleta de 4 colores: [0,2, 0,4, 0,3, 0,1]

# • Fitness del individuo

1. A partir de las proporciones de cada color de la paleta, se calcula el RGB correspondiente como la suma aditiva entre las distintas proporciones de cada color:

$$total\_proportion = \sum_0^{len(gen-1)} (color\_proportions)$$

$$r = round\left(\frac{\sum_0^{len(gen-1)} (gen[i] * palette[0][i])}{total\_proportion}\right)$$

$$g = round\left(\frac{\sum_0^{len(gen-1)} (gen[i] * palette[1][i])}{total\_proportion}\right)$$

$$b = round\left(\frac{\sum_0^{len(gen-1)} (gen[i] * palette[2][i])}{total\_proportion}\right)$$

# • Fitness del individuo

2. Una vez obtenido el RGB, se calcula el fitness como la distancia al color objetivo RGB con la siguiente fórmula:

$$fitness = 1 - \frac{distance(color, target\_color)}{MAX\_DISTANCE}$$

$$\bar{r} = \frac{C_{1,R} + C_{2,R}}{2}$$

$$\Delta R = C_{1,R} - C_{2,R}$$

$$\Delta G = C_{1,G} - C_{2,G}$$

$$\Delta B = C_{1,B} - C_{2,B}$$

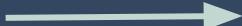
$$\Delta C = \sqrt{\left(2 + \frac{\bar{r}}{256}\right) \times \Delta R^2 + 4 \times \Delta G^2 + \left(2 + \frac{255 - \bar{r}}{256}\right) \times \Delta B^2}$$

# • Consideraciones

- Teniendo en cuenta las columnas R, G y B de la paleta de colores, es recomendable que haya alguna que sea **distinta de 0**, pues sino, al realizar la mezcla nunca se podrán obtener determinados colores.

Es decir, supongamos que tenemos en la paleta los siguientes 3 colores:

- 255,0,0
- 0,209,0
- 83,134,0



Al hacer la mezcla, el valor de B siempre será 0.  
Es decir, sólo podríamos obtener colores dentro del rango [0,0,0] y [255,255,0]

# • Métodos de selección

Utilizando la población inicial, o la nueva población de cada ciclo, se seleccionan K padres para cruzar

- **Elite:** se seleccionan K individuos de N individuos, se los ordena según el fitness y se elige cada uno  $n(i)$  veces, según  $n(i) = \lceil K \cdot i / N \rceil$
- **Por ruleta:** se generan K números uniformes y aleatorios  $r_i \in [0,1]$  y se eligen los K individuos que cumplen  $q_{i-1} < r_j \leq q_i$ , donde  $q_i$  es la aptitud relativa acumulada.
- **Universal:** análogo a por ruleta, pero se genera únicamente un único valor aleatorio y uniforme  $r \in [0,1]$  y se realiza la sucesión  $r_j = (r+j)/K$ , donde  $j \in [0, (K-1)]$
- **Por torneos determinísticos:** se eligen  $M \in [1, N]$  individuos al azar. De ellos se elige el mejor y se repite el proceso hasta conseguir los K individuos.
- **Por torneos probabilísticos:** primero se elige un valor de threshold  $\in [0.5, 1]$  (uniforme y aleatorio). Luego se eligen 2 individuos al azar y se toma un valor  $r \in [0,1]$  (uniforme y aleatorio). Entonces, si  $r < \text{Threshold}$  se selecciona el más apto, sino se selecciona el menos apto. Se repite el proceso hasta conseguir los K individuos.

# • Métodos de crusa

Utilizando la población ya seleccionada, se eligen dos padres consecutivos, y se crean 2 hijos a partir de ellos.

- **Cruza de un punto:** Se elige un locus al azar y luego se intercambian las proporciones de los padres a partir de dicho locus para generar dos hijos.
- **Cruza de dos puntos:** Se eligen dos locus al azar y se intercambian los genes entre ellos.
- **Cruza angular:** Se elige una locus al azar  $P$  y una longitud  $L$  entre  $[0, \lceil \text{len(cromosoma)}/2 \rceil]$ . Luego se intercambian  $L$  proporciones a partir de  $P$ .
- **Cruza uniforme:** Se intercambia con probabilidad 0,5 cada proporción.



# • Métodos de mutación

Sea un hijo y una probabilidad  $P_m$  previamente estipulada, se modifican las proporciones de dicho hijo de la siguiente manera:

1. Establecemos un  $u = \text{gen} * 0,5$
2. Generamos un número aleatorio  $\delta = \text{Uniforme}(-u, u)$
3. Reasignamos  $\text{gen} = \text{gen} + \delta$

- **Un gen:** se altera una sola proporción con una probabilidad  $P_m$
- **Mutigen limitado:** se selecciona una cantidad  $[1, M]$  de proporciones para mutar, cada una con probabilidad  $P_m$
- **Mutigen uniforme:** cada proporción tiene una probabilidad  $P_m$  de ser mutado.
- **Completa:** con una probabilidad  $P_m$  se mutan todas las proporciones del cromosoma.



- **Métodos de selección de nueva población**

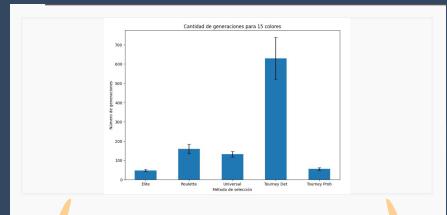
- **Use all:** Se seleccionan N individuos del conjunto de parents + children.
- **New over actual:** Se seleccionan N individuos del conjunto de children. Si no fuera suficiente, se completa con los mejores parents hasta llegar a N.

# 03

## • Conclusiones •

# • Consideraciones generales

- Todas las pruebas fueron realizadas tomando un valor variable y todos los demás constantes, con el objetivo de poder compararlas entre ellas.
- Se utiliza la paleta de colores con todas las posibles combinaciones de 0 y 255 en RGB.
- Se generan 50 colores al azar para buscar, y se mantiene la población inicial de individuos.
- Se toma siempre  $N = K = 100$
- Se tomaron 2 condiciones de corte,  $\delta=0,01$  o más de 1000 generaciones.



# Variando métodos de selección

Parámetros fijos:  
Cruza Uniforme  
Mutación Uniforme  
Porcentaje de mutación = 0,5



# • Métodos de selección

En orden de convergencia a la solución utilizando use all:

1. **Elite:** prioriza aptitud y algunos padres pasan a la siguiente población.
  2. **Torneo Probabilístico:** genera alta diversidad genética y mucha fluctuación en los valores.
  3. **Ruleta:**
  4. **Universal:**
  5. **Torneo Determinístico:** puede mantener durante muchas generaciones un mismo individuo y converger prematuramente.
- Muchas fluctuaciones en un rango acotado.  
En algunos casos puede tardar en converger.



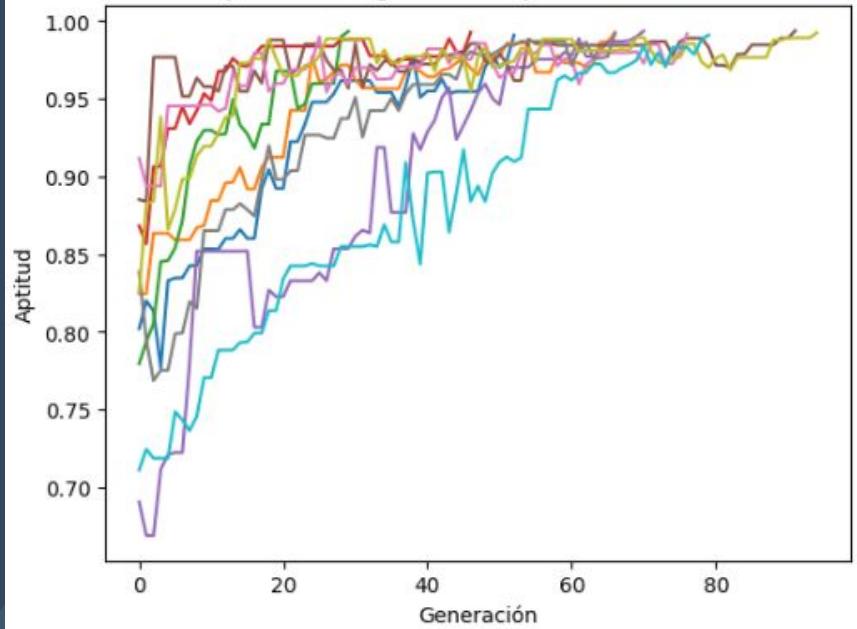
# • Métodos de selección

En orden de convergencia a la solución utilizando **new over actual**:

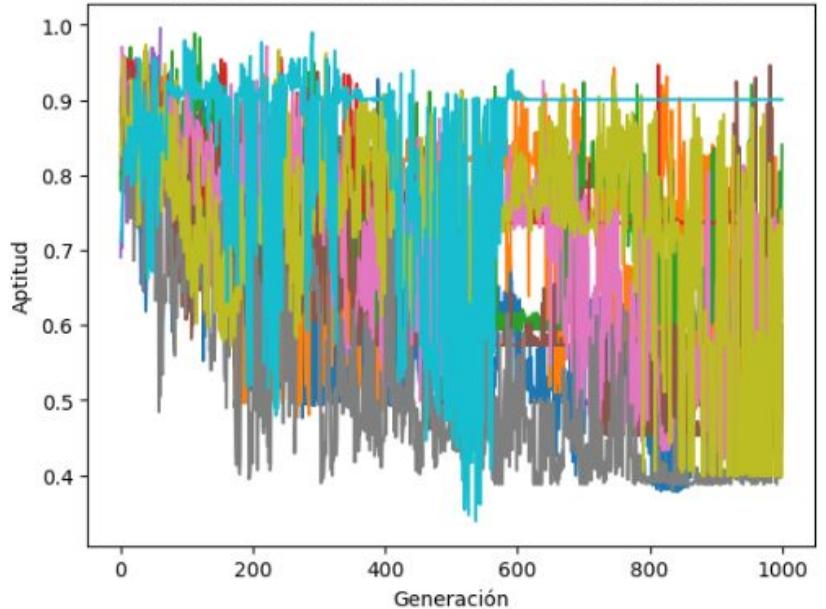
1. **Torneo Determinístico:** se pierde el riesgo de mantener un parent por varias generaciones y se persisten las mejores características.
2. **Torneo Probabilístico:** nuevamente se genera gran diversidad genética.
3. **Ruleta:** se mantiene similar que con use all.
4. **Universal:** pierde aleatoriedad con respecto a ruleta, tarda más y puede caer en convergencia prematura.
5. **Elite:** cae en convergencia prematura por presión de selección muy alta.



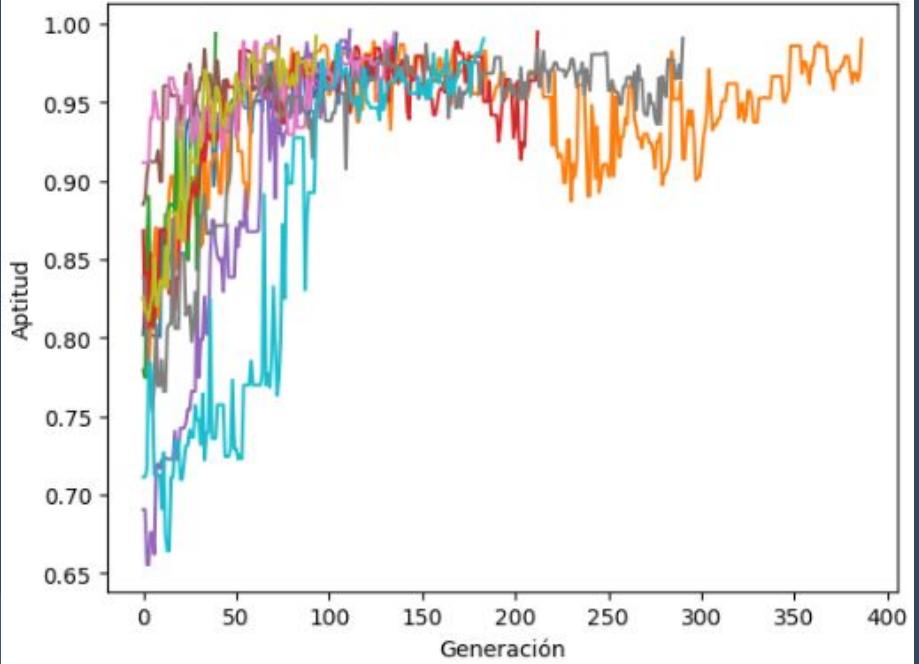
Mejor de cada generación para selección Elite



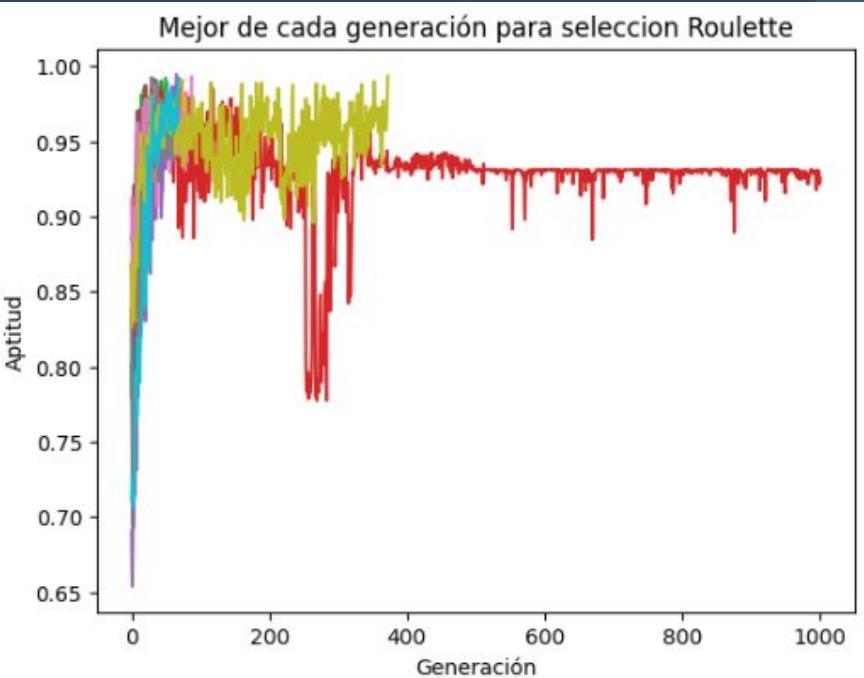
Mejor de cada generación para selección Elite



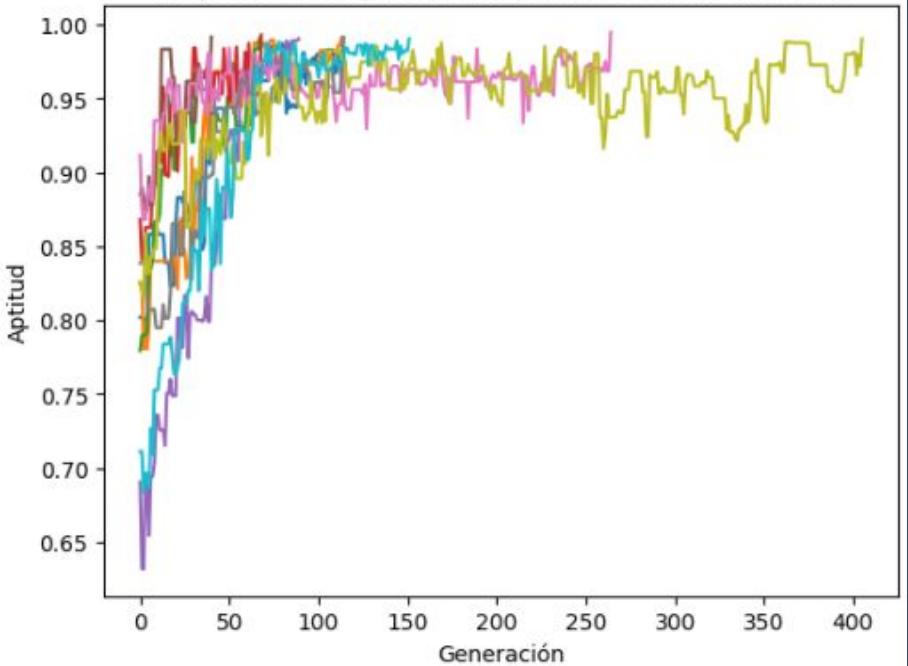
Mejor de cada generación para selección Roulette



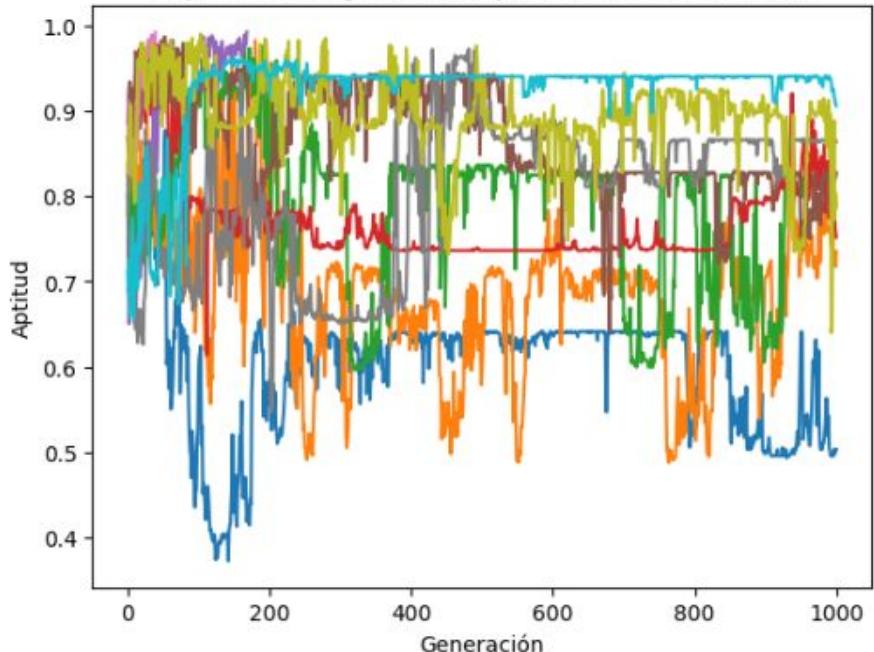
Mejor de cada generación para selección Roulette



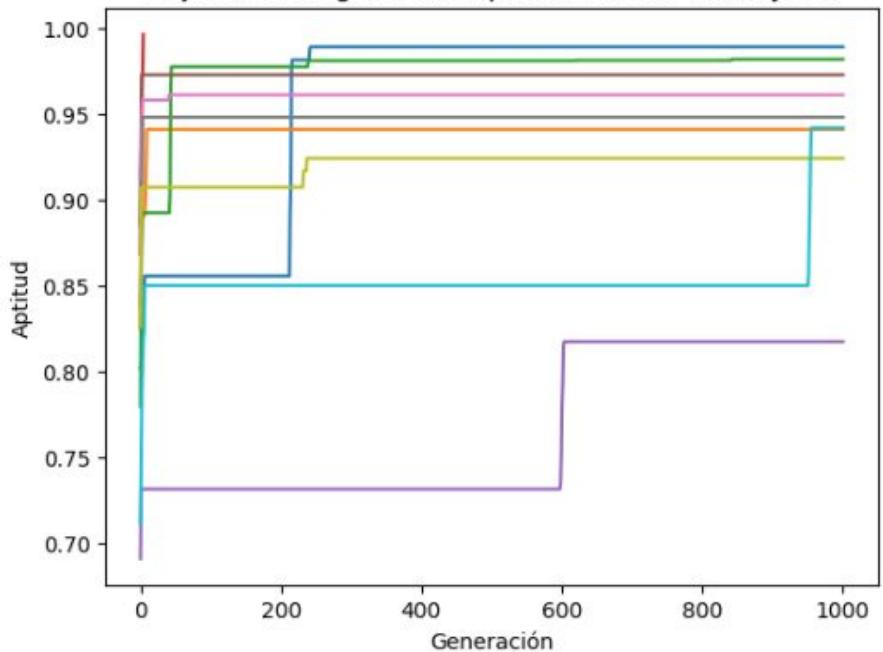
Mejor de cada generación para selección Universal



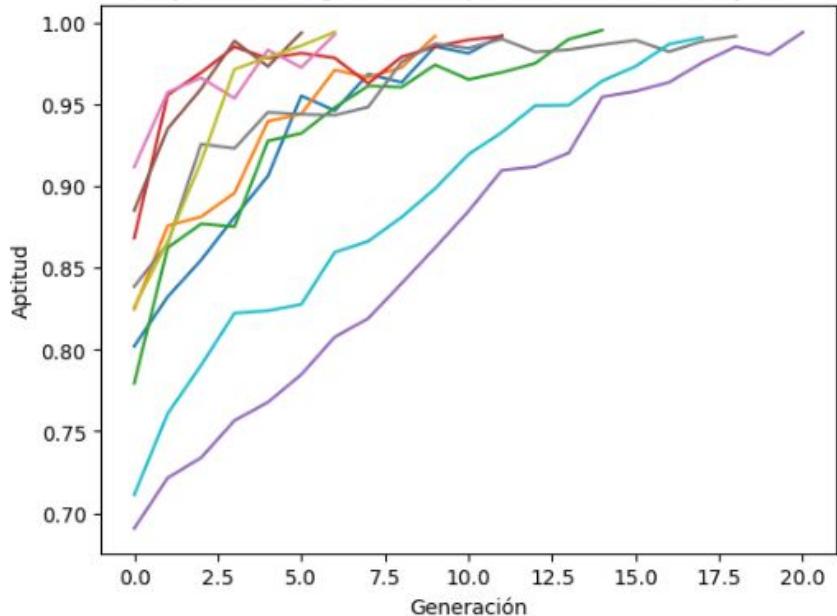
Mejor de cada generación para selección Universal

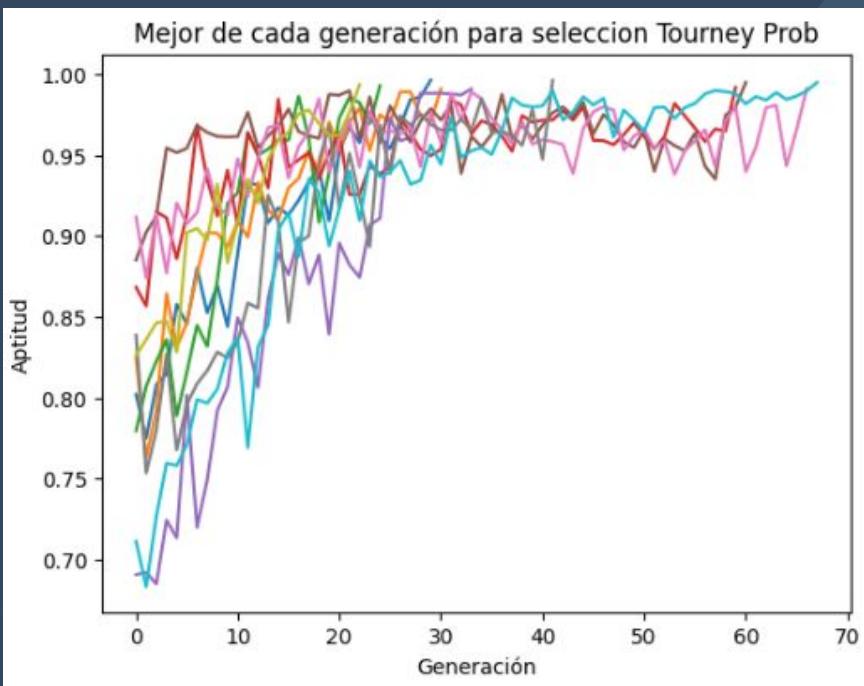
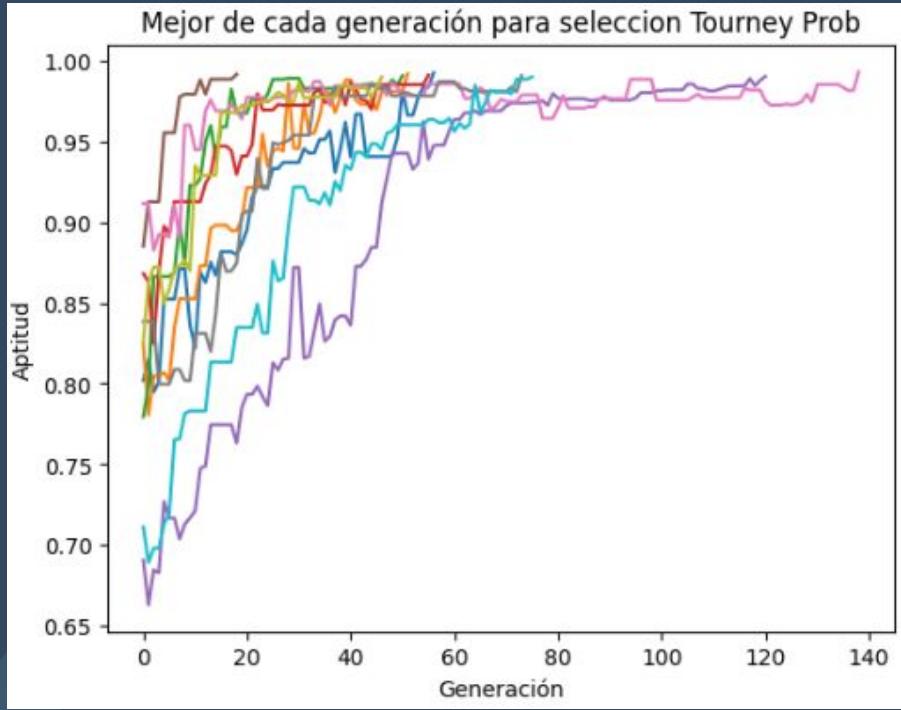


Mejor de cada generación para selección Turney Det

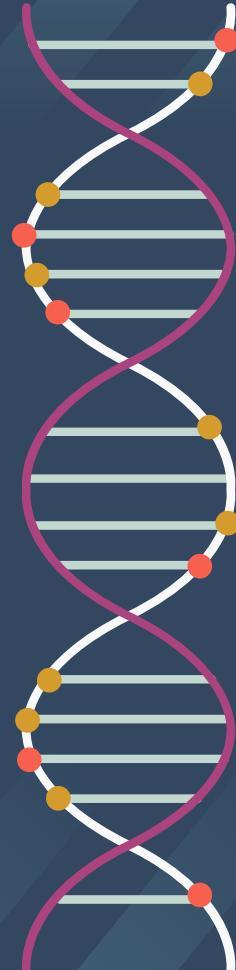


Mejor de cada generación para selección Turney Det

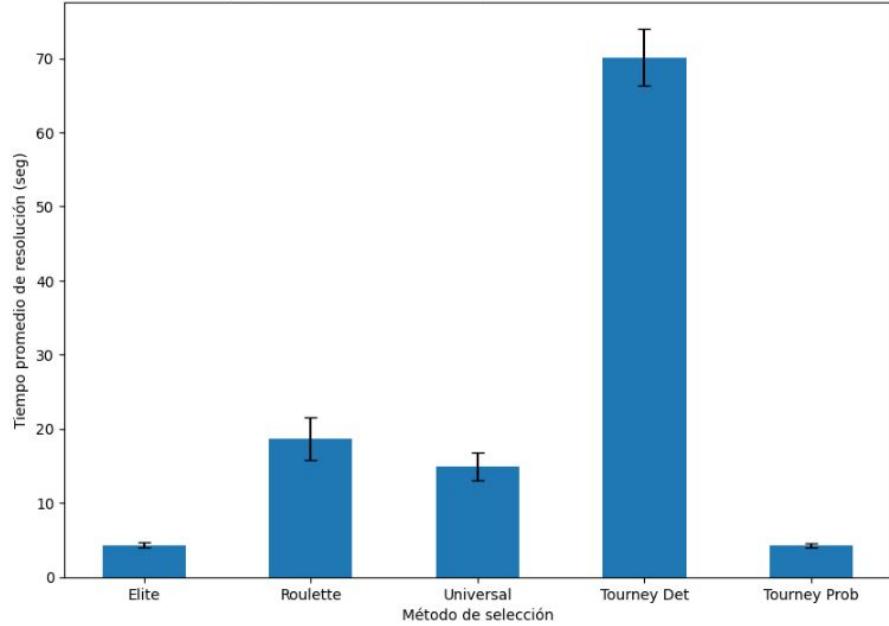




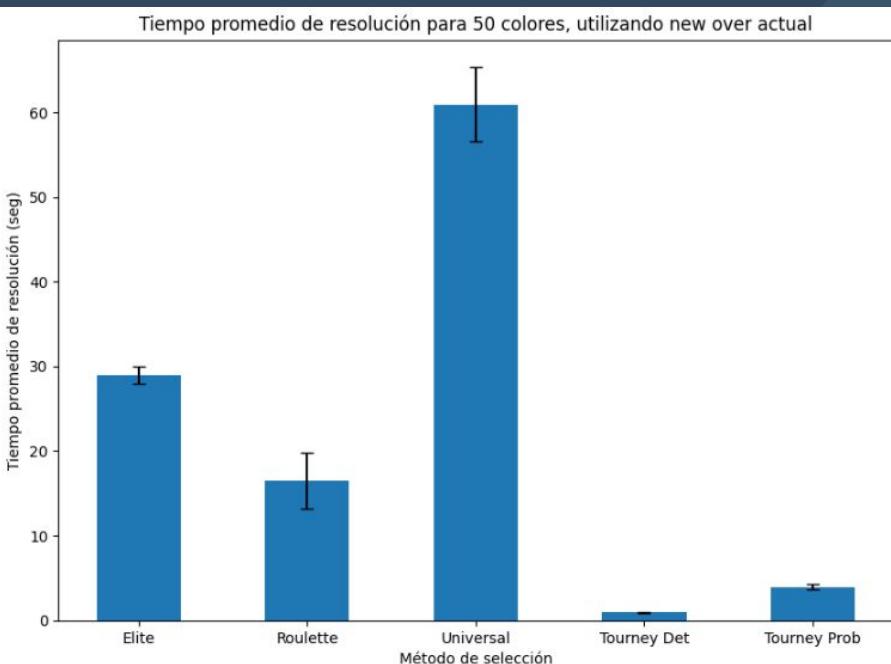
# Tiempo promedio de resolución



Tiempo promedio de resolución para 50 colores, utilizando use all



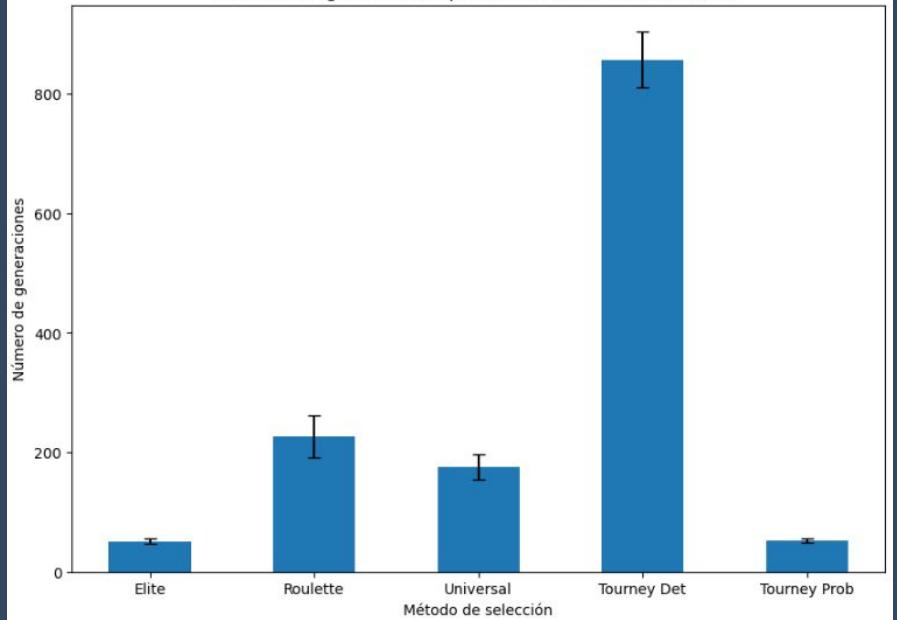
Tiempo promedio de resolución para 50 colores, utilizando new over actual



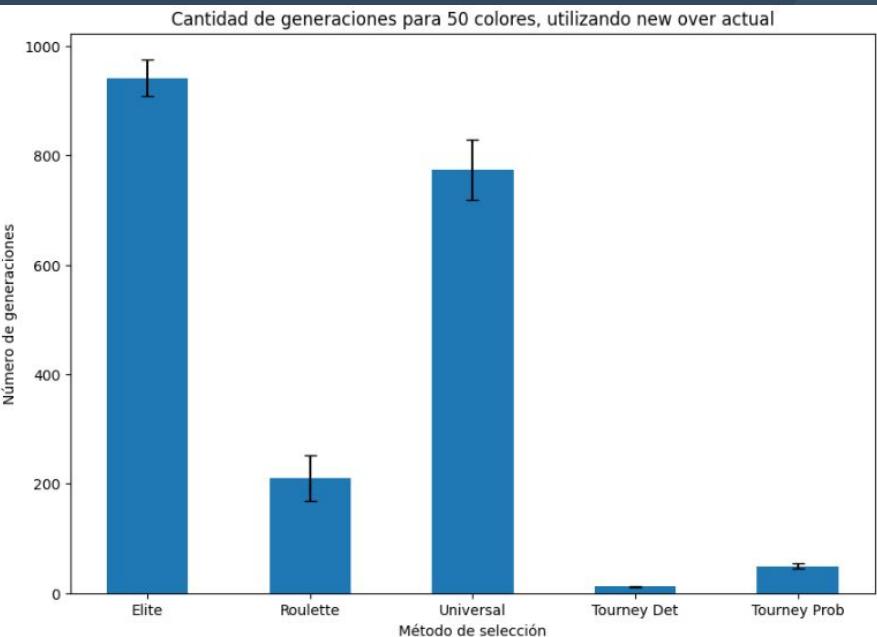
# Cantidad de generaciones



Cantidad de generaciones para 50 colores, utilizando use all



Cantidad de generaciones para 50 colores, utilizando new over actual



# Variando métodos de cruza

Parámetros fijos:  
Selección por Torneo Probabilístico  
Mutación Uniforme  
Porcentaje de mutación = 0,5



# • Métodos de crusa

En orden de convergencia a la solución utilizando use all:

1. **Uniforme:** genera gran diversidad genética de buenas características de los padres
  2. **Un punto:**
  3. **Doble punto:**
  4. **Angular:** puede variar de  $[0, \lceil S/2 \rceil]$  (0-50% de crusa)
- } Puede variar de  $[1, S]$  genes, y resulta en diversidad genética similar



# • Métodos de crusa

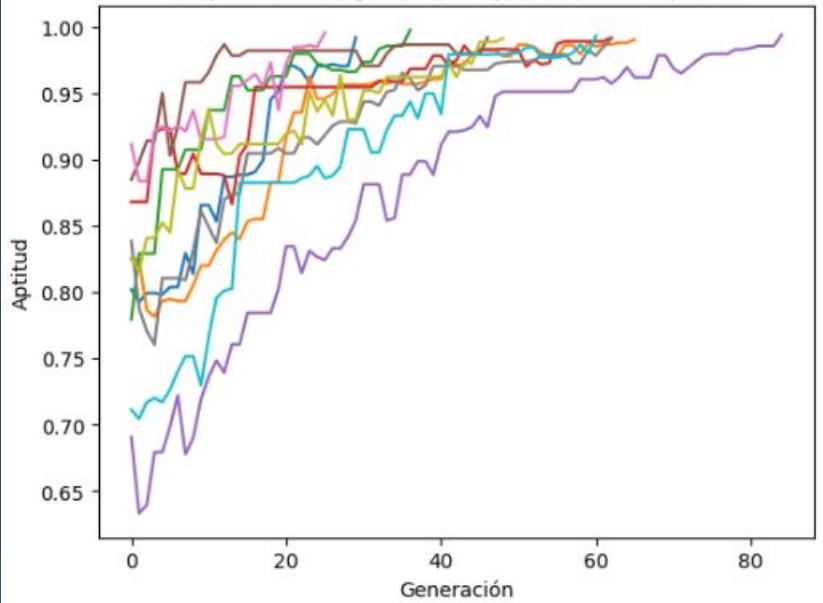
En orden de convergencia a la solución utilizando new over actual:

1. **Un punto**
2. **Angular**
3. **Doble punto**
4. **Uniforme**

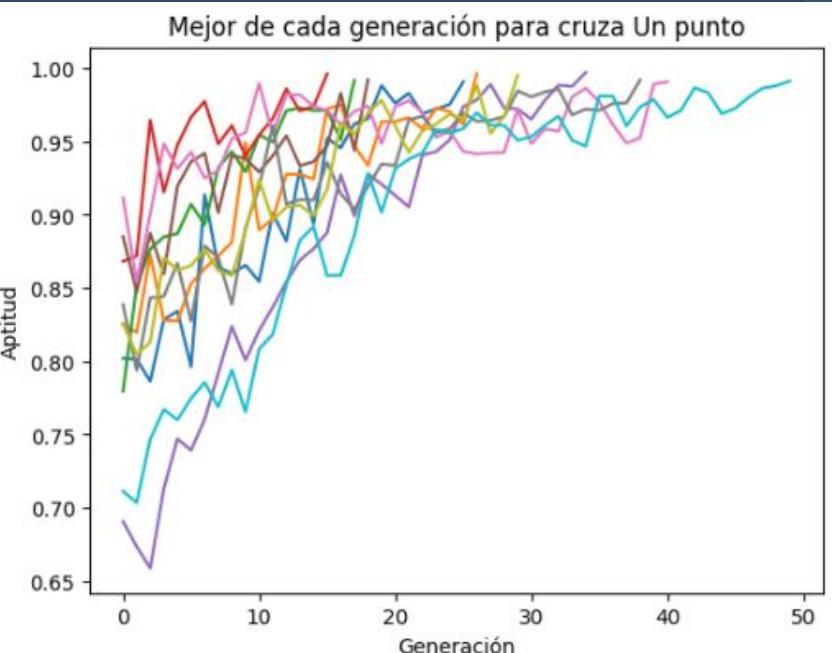
Como se prioriza a los hijos, demasiada crusa puede llevar a nuevas configuraciones no favorables de genes



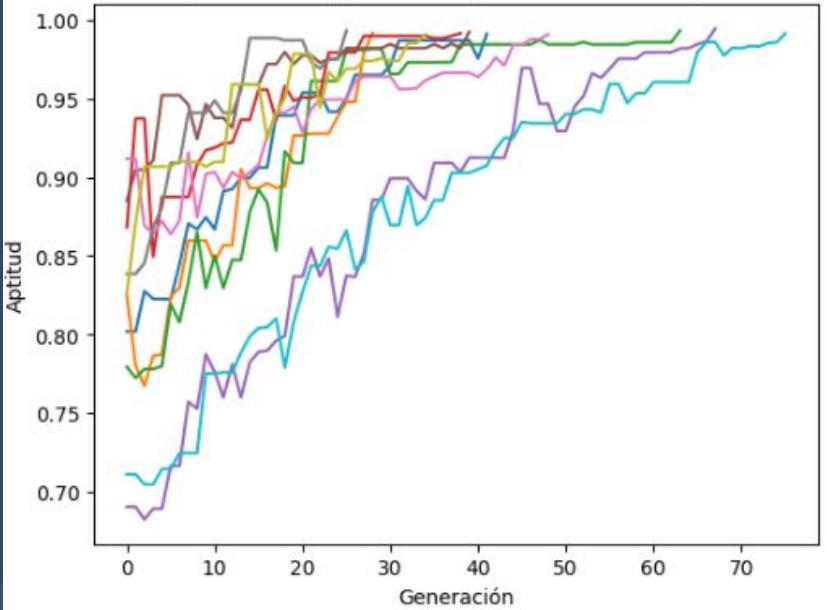
Mejor de cada generación para cruza Un punto



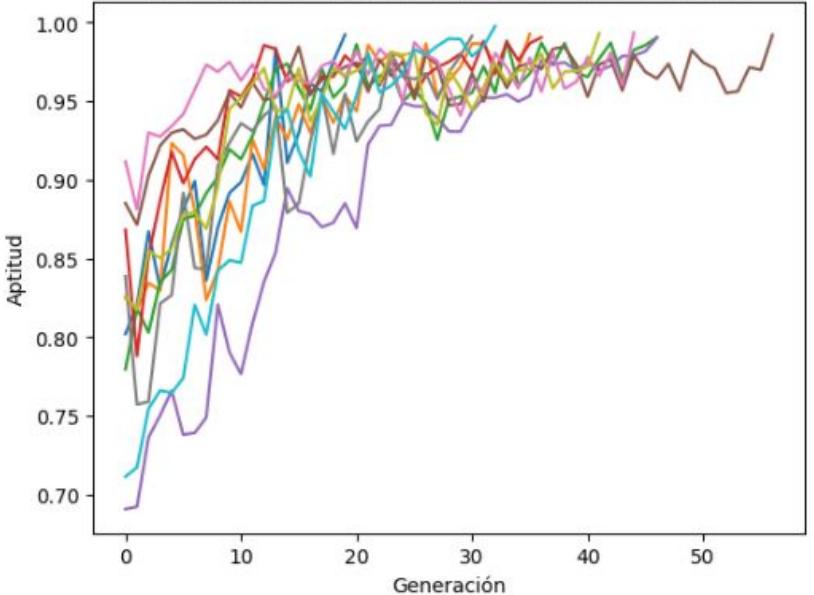
Mejor de cada generación para cruza Un punto



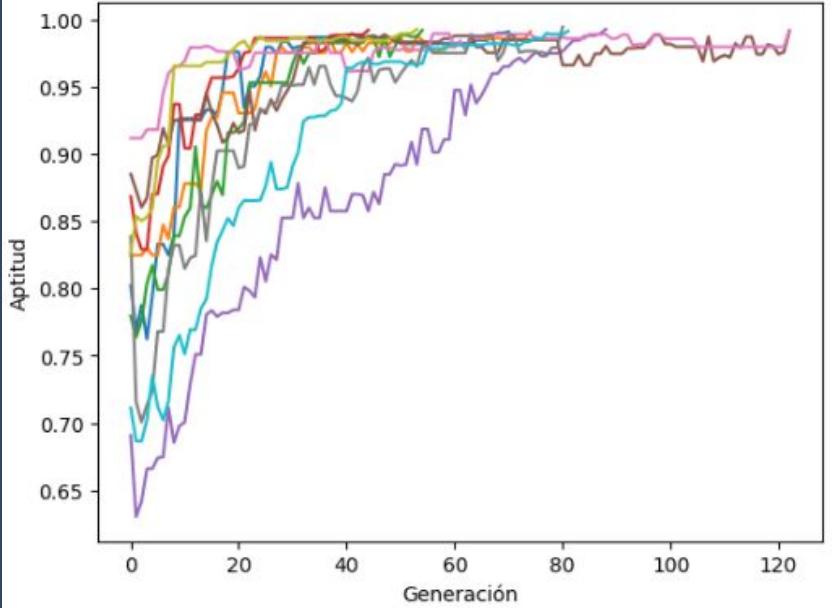
Mejor de cada generación para cruce Doble punto



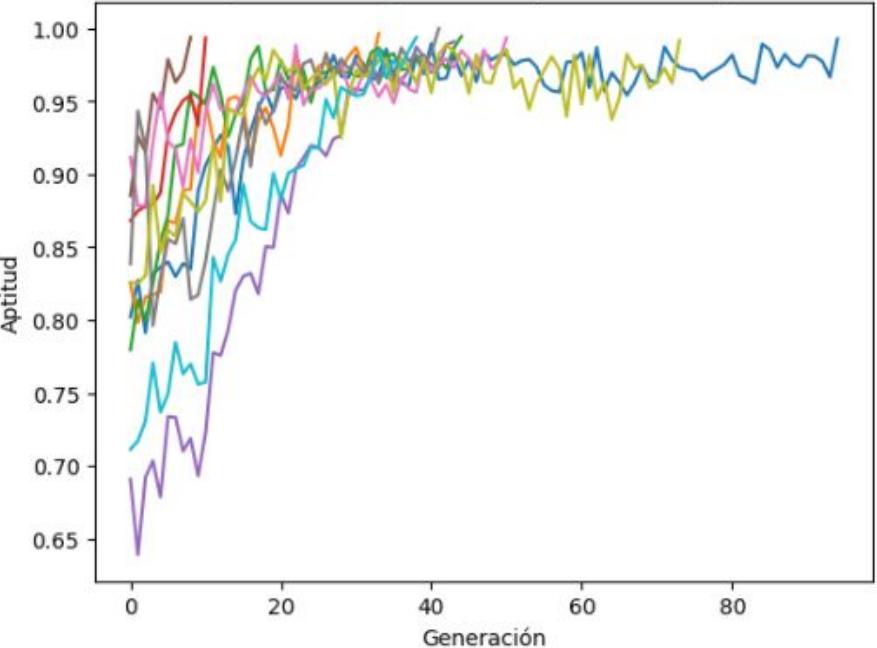
Mejor de cada generación para cruce Doble punto



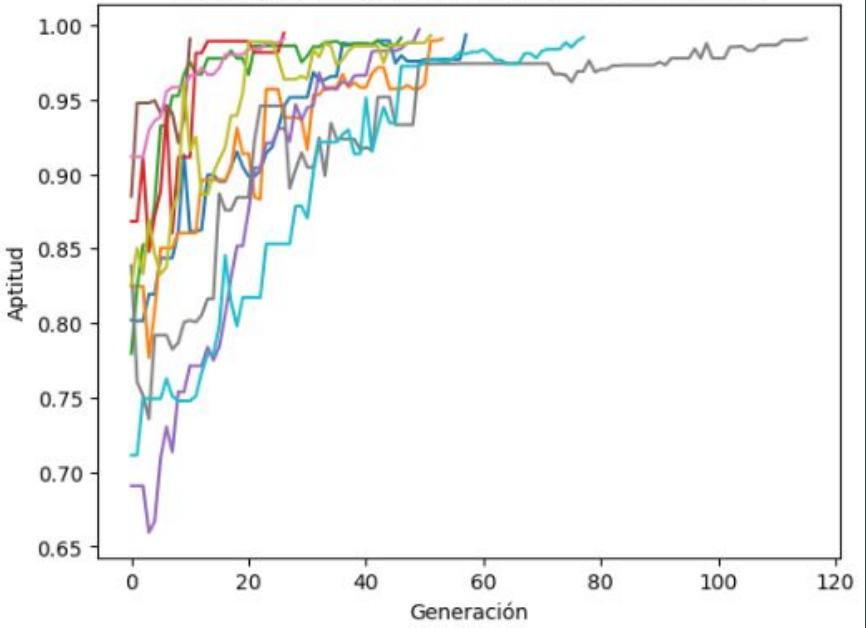
Mejor de cada generación para cruce Angular



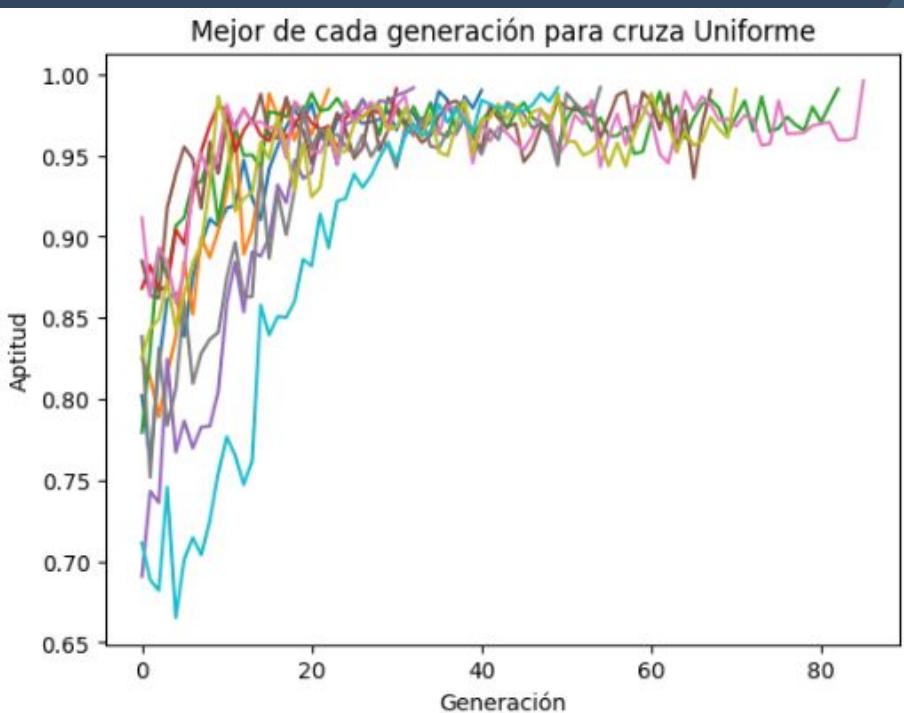
Mejor de cada generación para cruce Angular



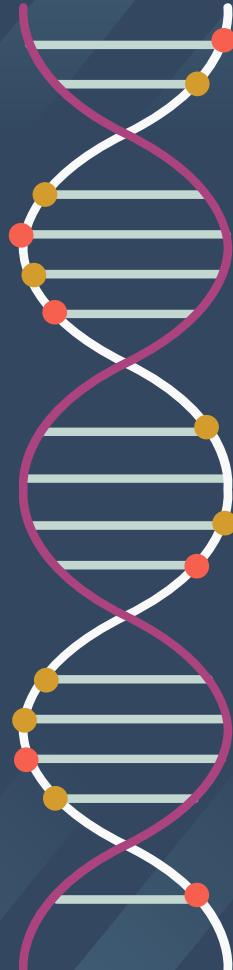
Mejor de cada generación para crusa Uniforme



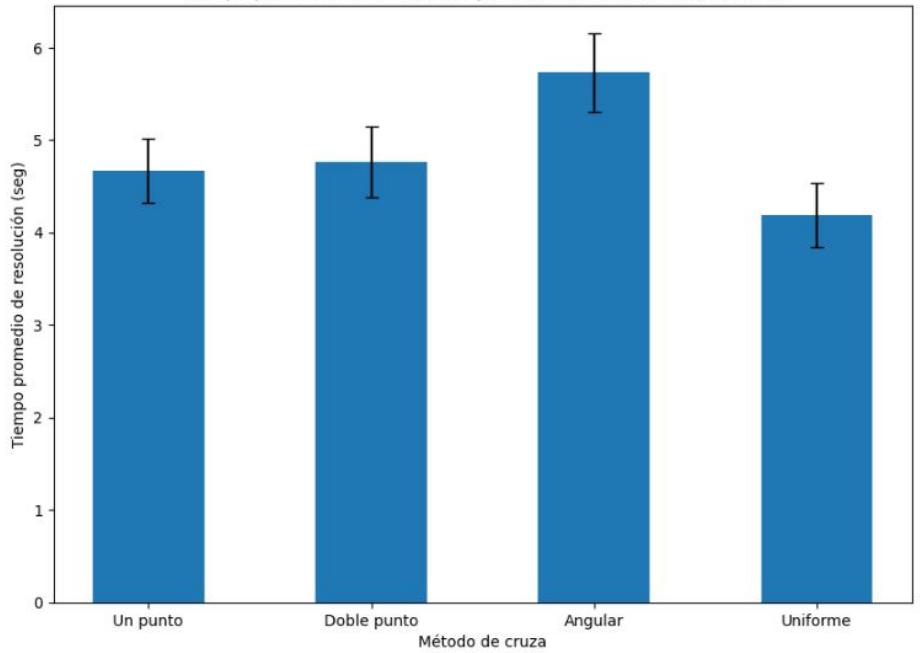
Mejor de cada generación para crusa Uniforme



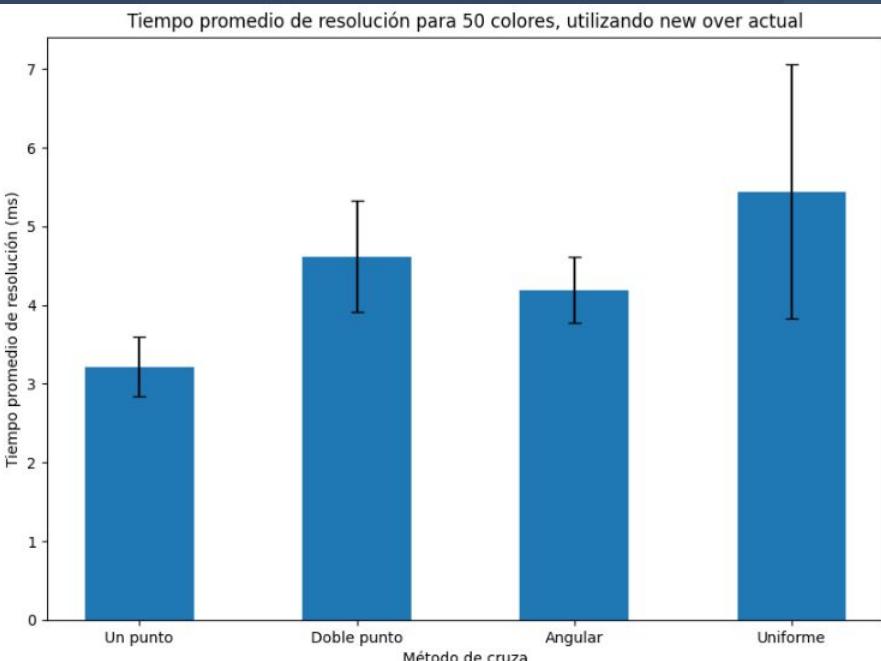
# Tiempos de resolución promedio



Tiempo promedio de resolución para 50 colores, utilizando use all



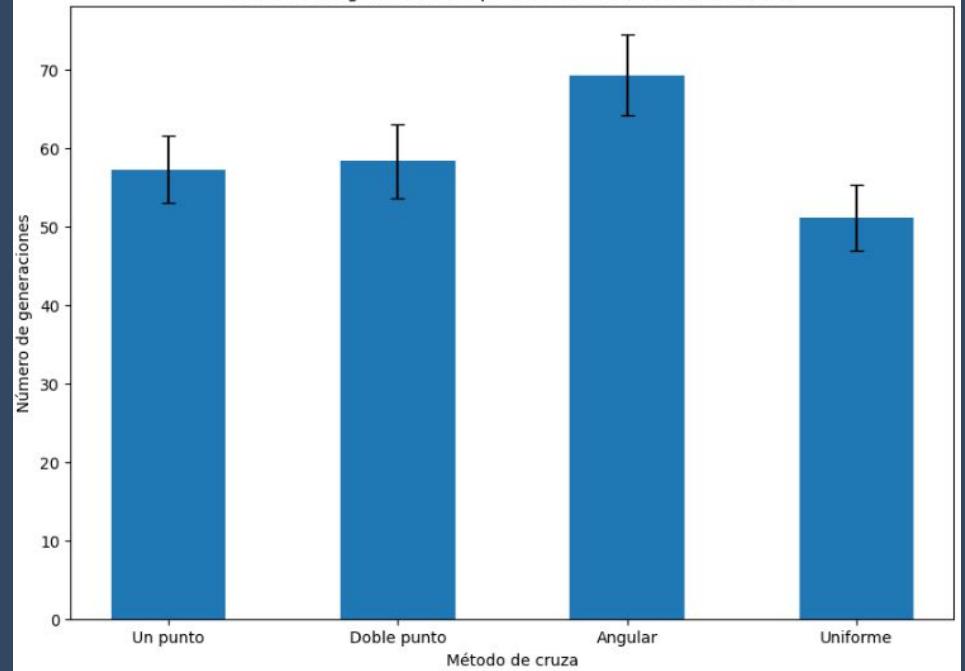
Tiempo promedio de resolución para 50 colores, utilizando new over actual



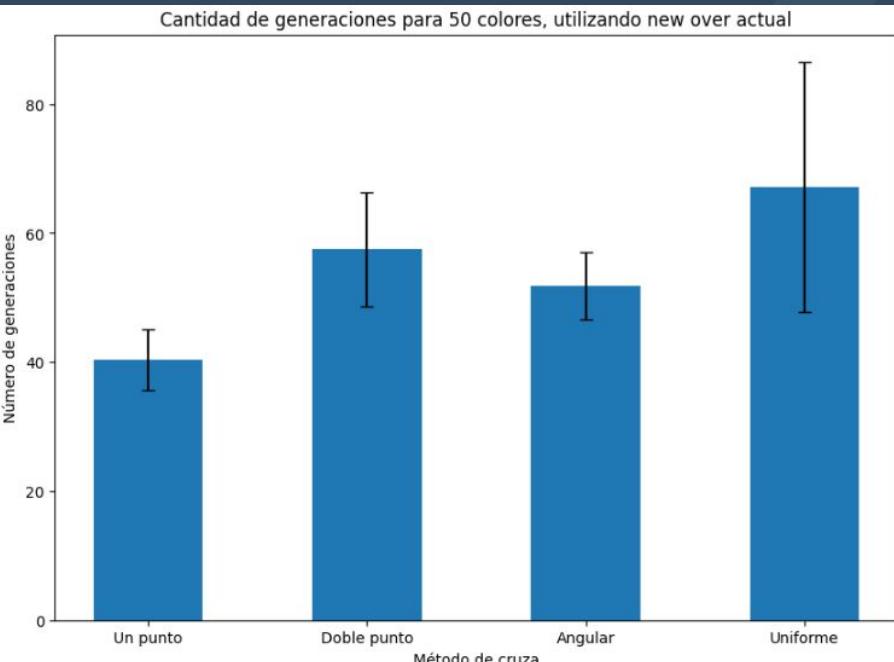
# Cantidad de generaciones



Cantidad de generaciones para 50 colores, utilizando use all



Cantidad de generaciones para 50 colores, utilizando new over actual



# Variando métodos de mutación

Parámetros fijos:  
Selección por Torneos Probabilísticos  
Cruza Uniforme  
Porcentaje de mutación = 0,5



# • Métodos de mutación

En orden de convergencia a la solución utilizando use all:

1. **Completa**
2. **Multigen Limitado**
3. **Un gen**
4. **Multigen Uniforme**

No hay cambios significativos en cuanto al método de mutación elegido, así que podría desempatarse por coste computacional o espacial



# • Métodos de mutación

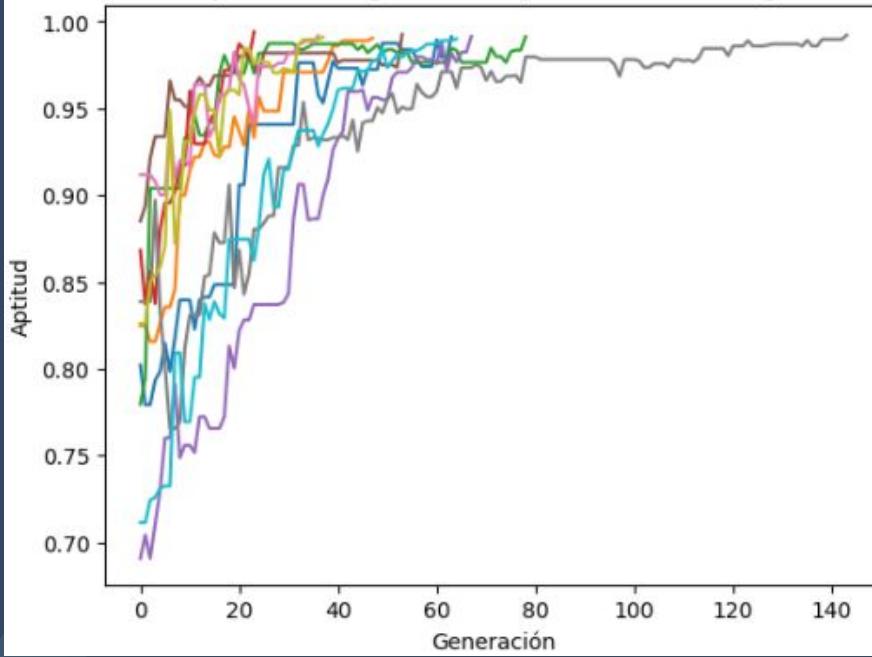
En orden de convergencia a la solución utilizando **new over actual**:

1. **Un gen**
2. **Multigen Uniforme**
3. **Completa**
4. **Multigen Limitado**

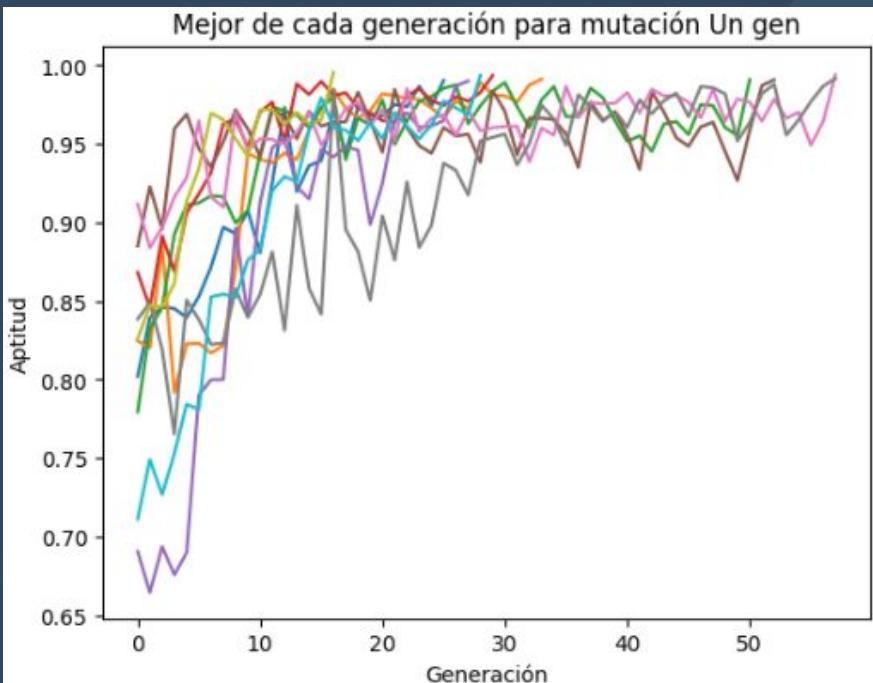
Al igual que con use all, los valores obtenidos siguen siendo similares, pero se recomienda Un gen o Multigen Uniforme que arrojan mejores resultados



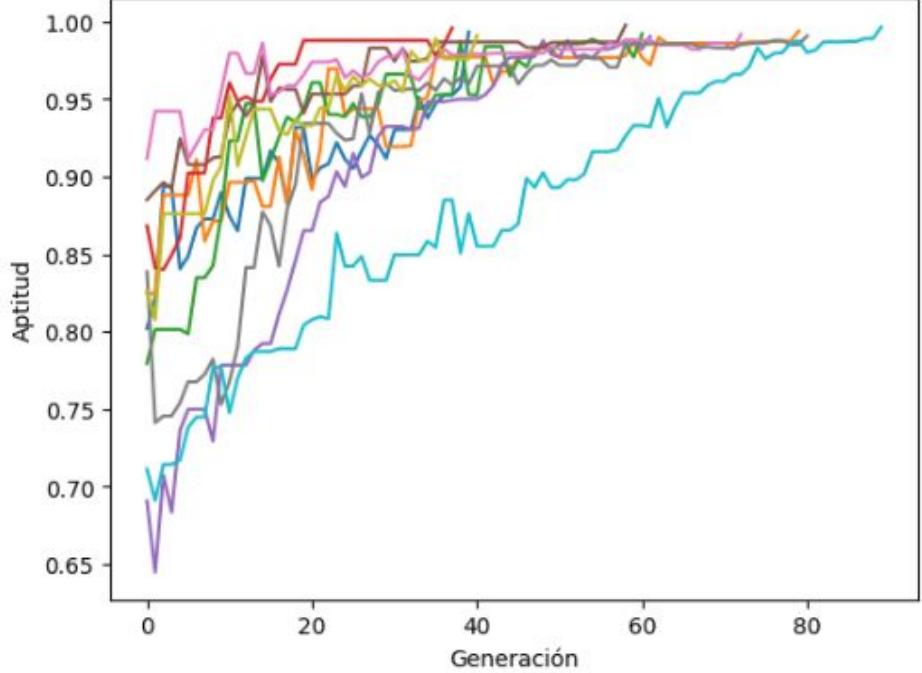
Mejor de cada generación para mutación Un gen



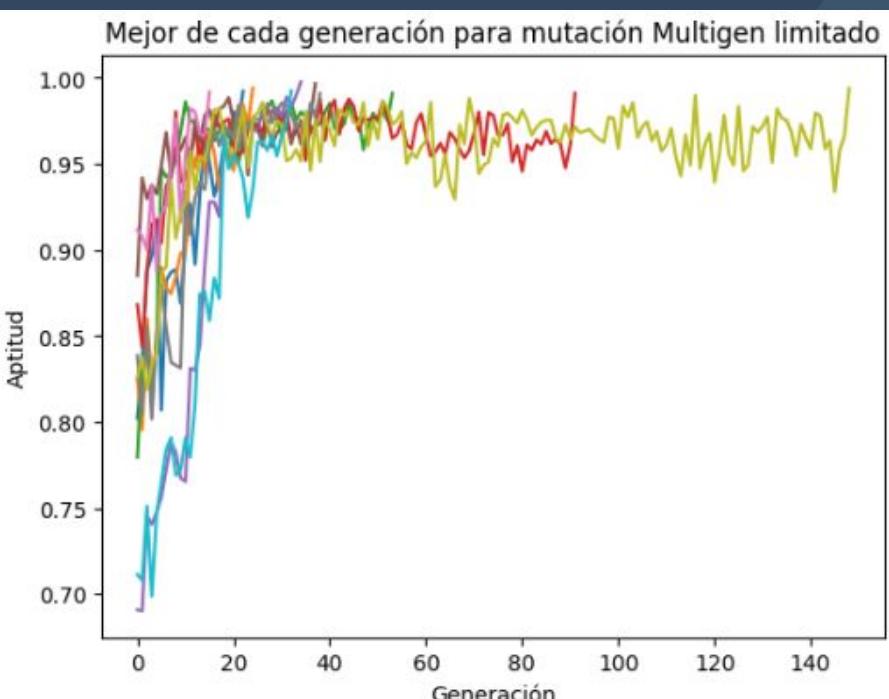
Mejor de cada generación para mutación Un gen



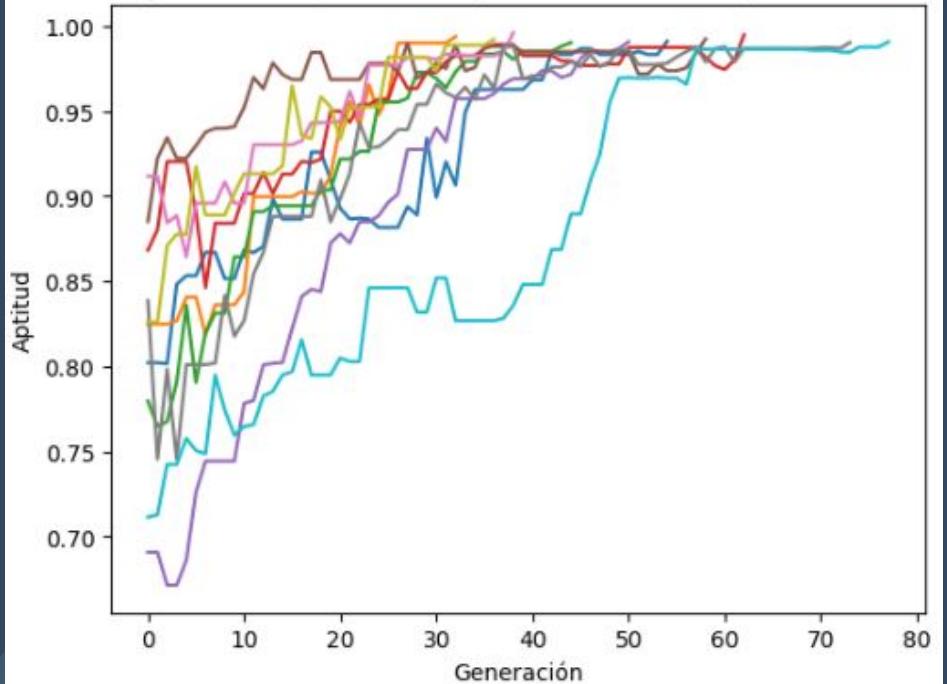
Mejor de cada generación para mutación Multigen limitado



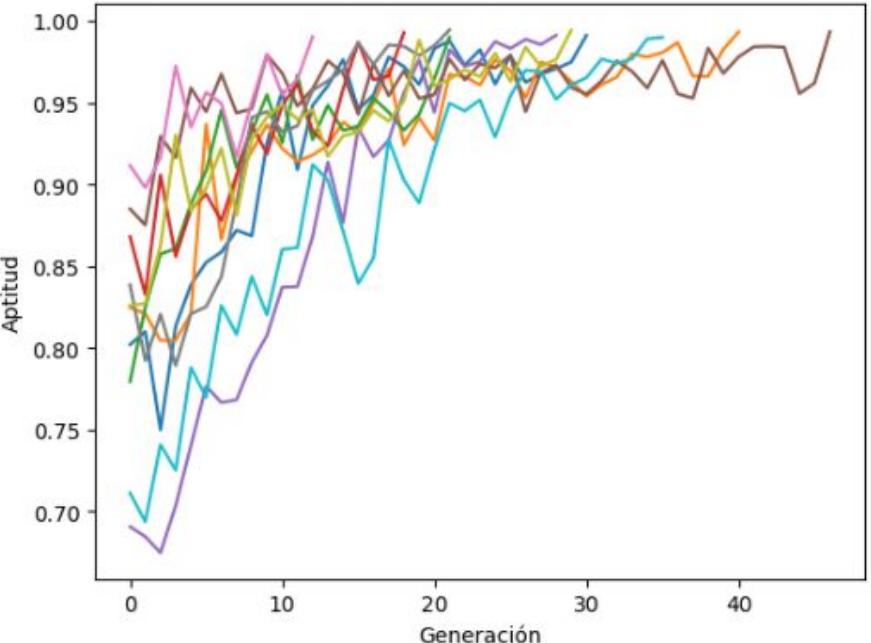
Mejor de cada generación para mutación Multigen limitado



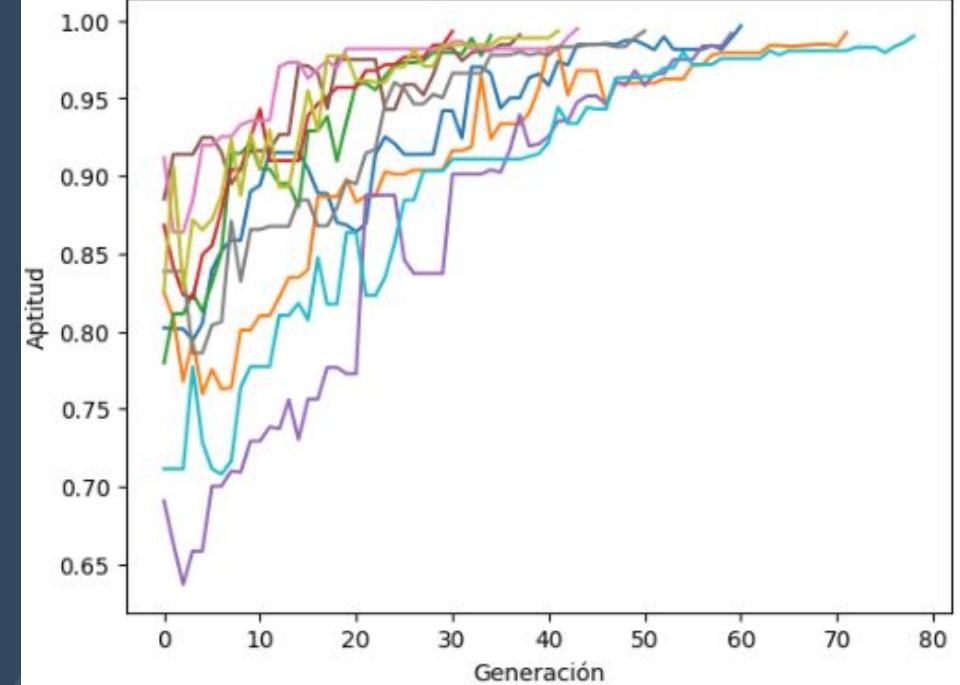
Mejor de cada generación para mutación Multigen uniforme



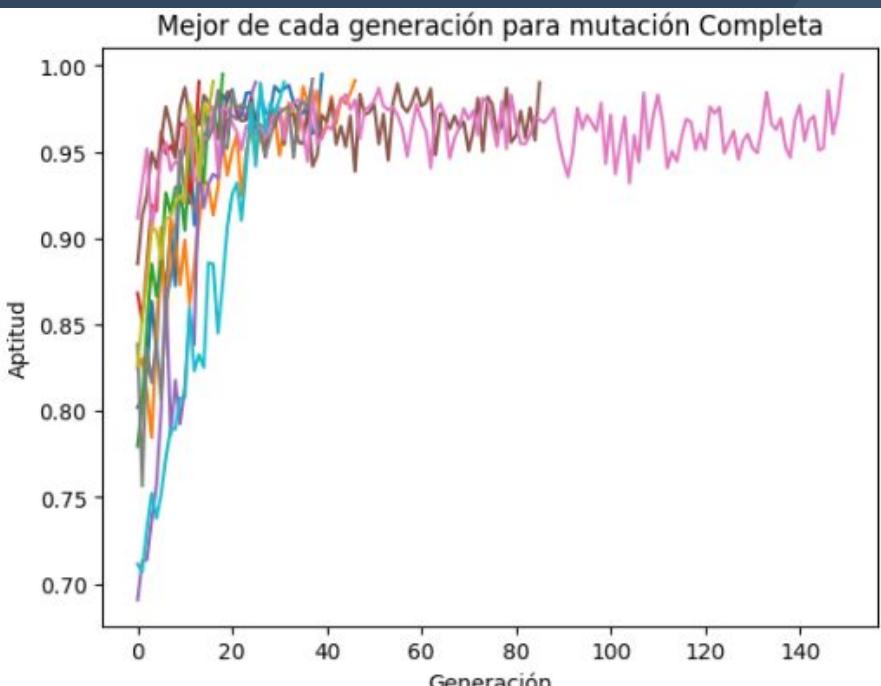
Mejor de cada generación para mutación Multigen uniforme



Mejor de cada generación para mutación Completa



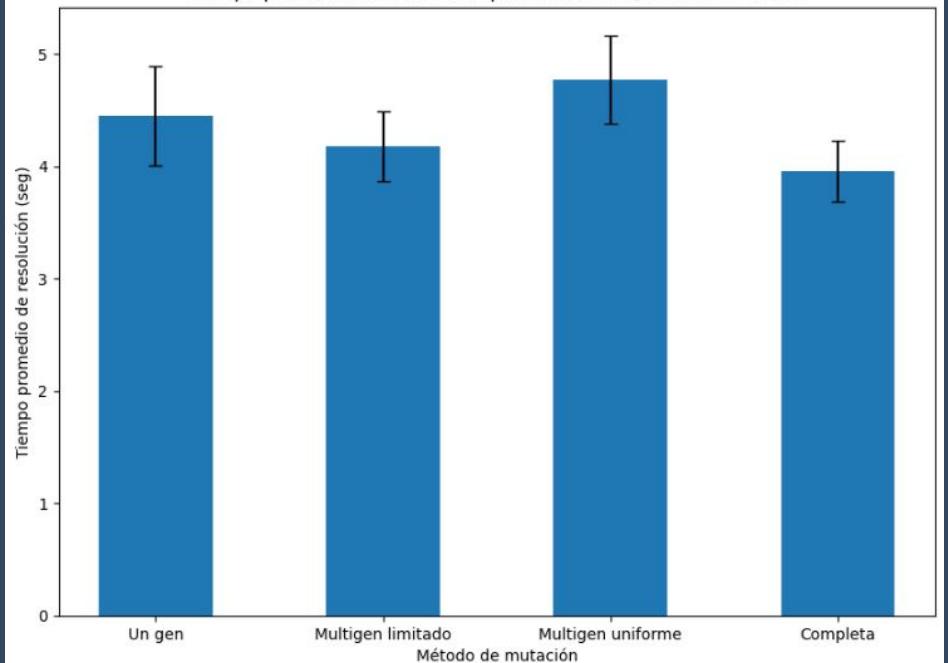
Mejor de cada generación para mutación Completa



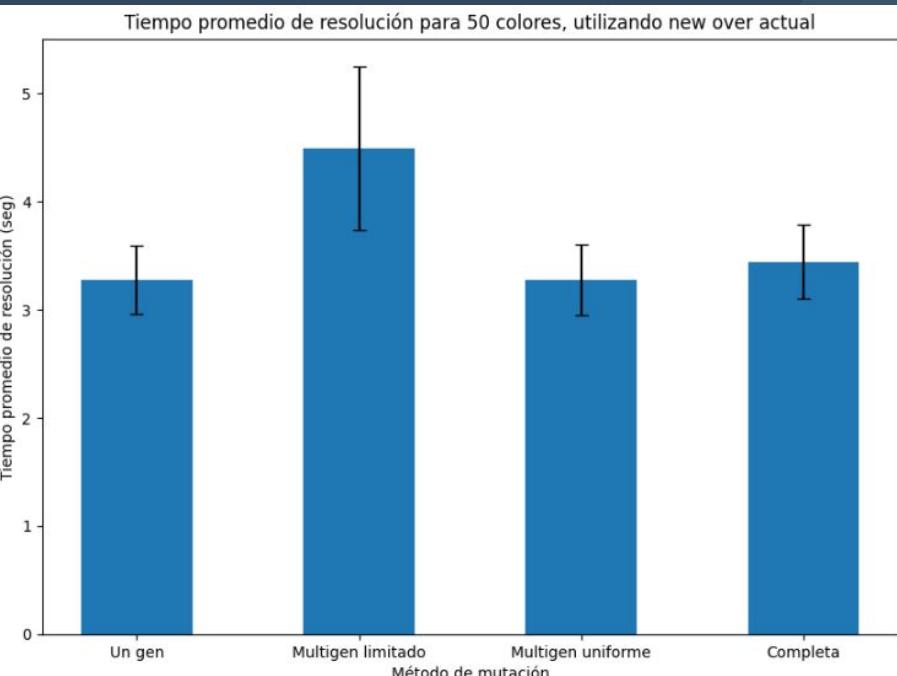
## Tiempos promedio de resolución



Tiempo promedio de resolución para 50 colores, utilizando use all



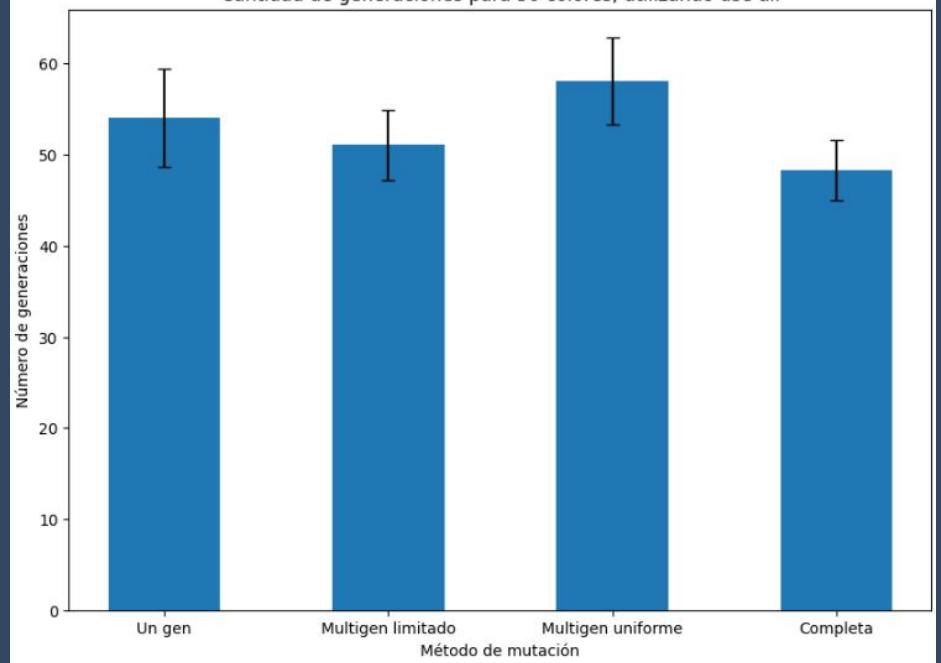
Tiempo promedio de resolución para 50 colores, utilizando new over actual



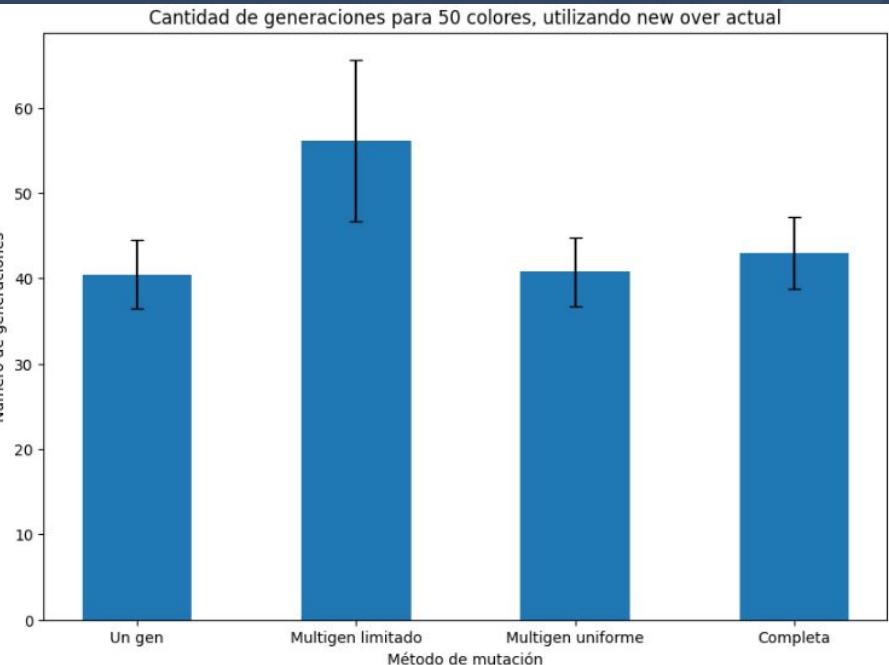
# Cantidad de generaciones



Cantidad de generaciones para 50 colores, utilizando use all



Cantidad de generaciones para 50 colores, utilizando new over actual



# Variando porcentajes de mutación (Pm)

Parámetros fijos:  
Selección por Torneos Probabilísticos  
Cruza Uniforme  
Mutación Uniforme



# • Porcentaje de mutación

En orden de convergencia a la solución utilizando use all:

1. 0,6
2. 0,4
3. 0,2
4. 0,8
5. 1

Los promedios de las generaciones para cada porcentaje forma una especie de parábola, con su punto más bajo en los porcentajes 0,6 y 0,4

Entonces, podemos decir que mucha o muy poca mutación no son convenientes



# • Porcentaje de mutación

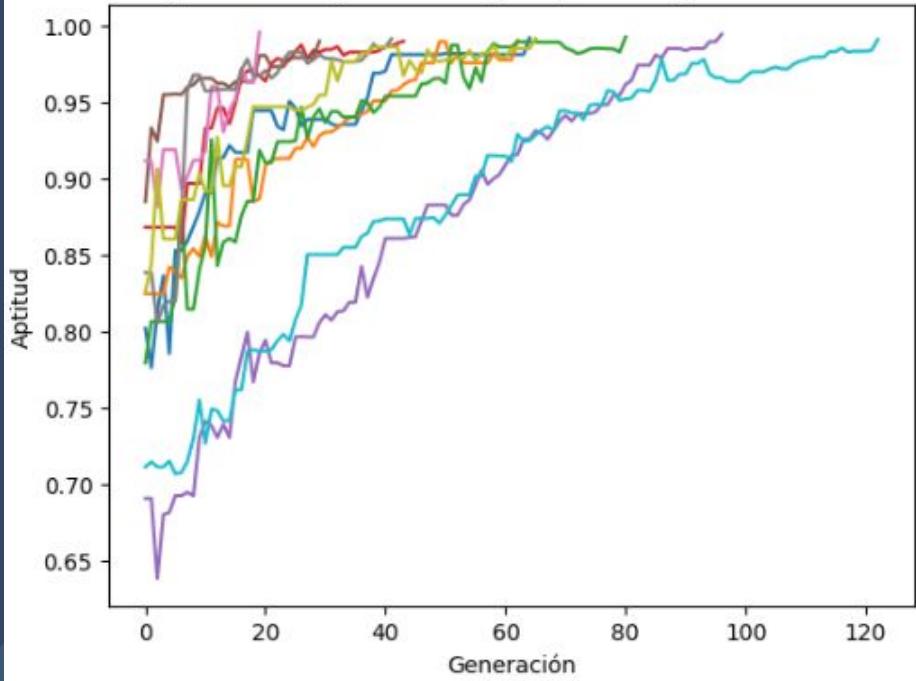
En orden de convergencia a la solución utilizando new over actual:

1. 0,2
2. 0,4
3. 0,6
4. 0,8
5. 1

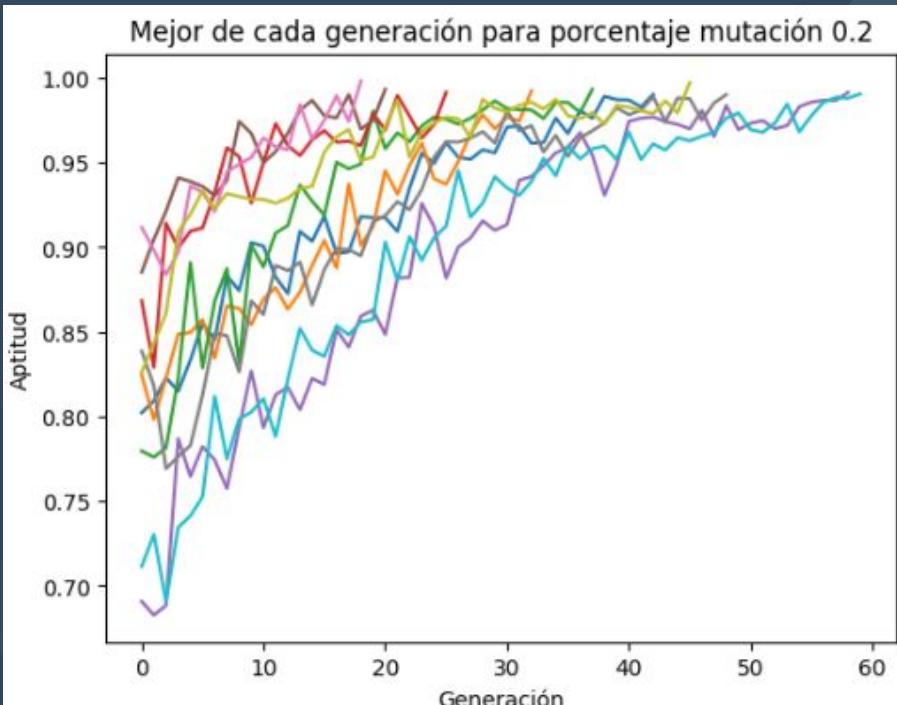
A mayor porcentaje de mutación se pierden más las características de los padres, entonces se tarda más generaciones en estos casos



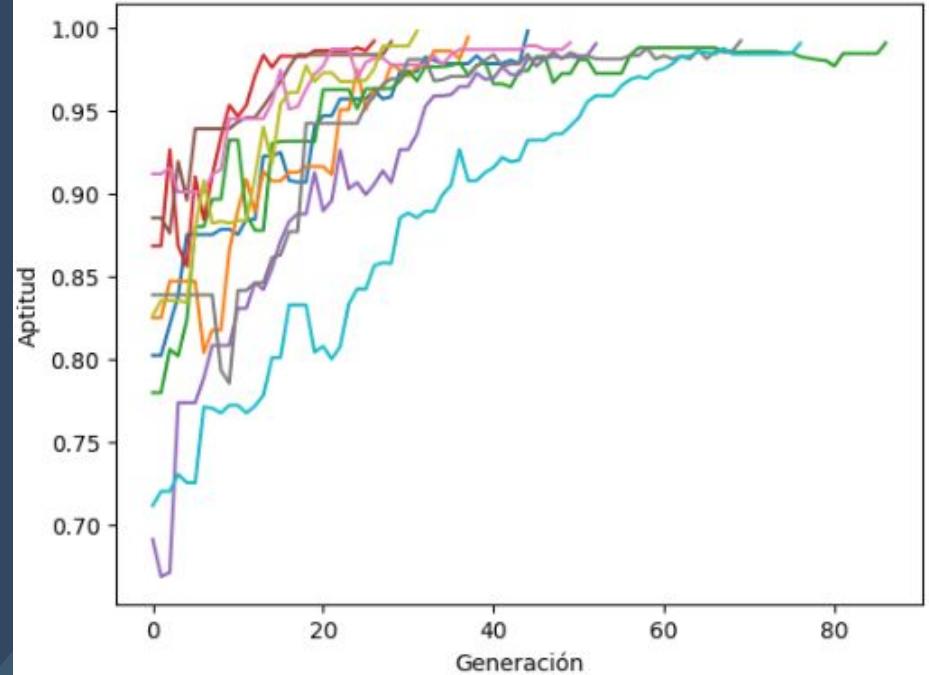
Mejor de cada generación para porcentaje mutación 0.2



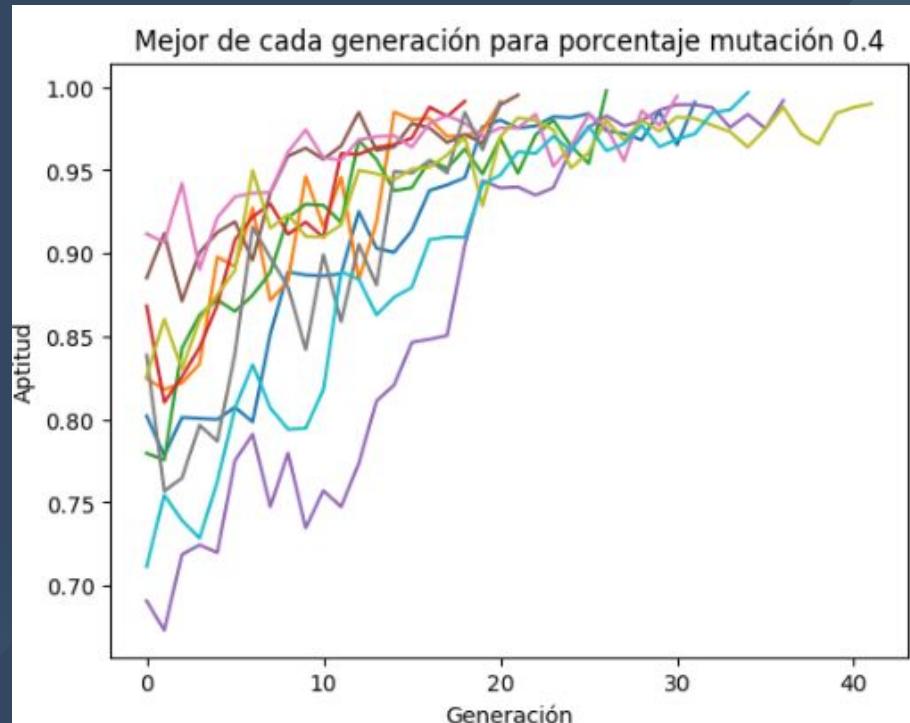
Mejor de cada generación para porcentaje mutación 0.2



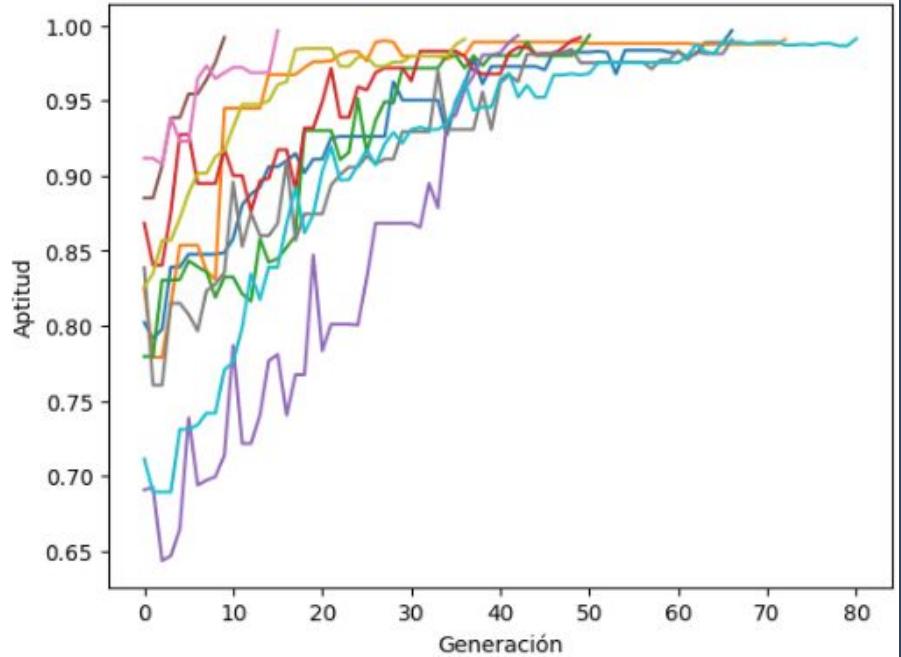
Mejor de cada generación para porcentaje mutación 0.4



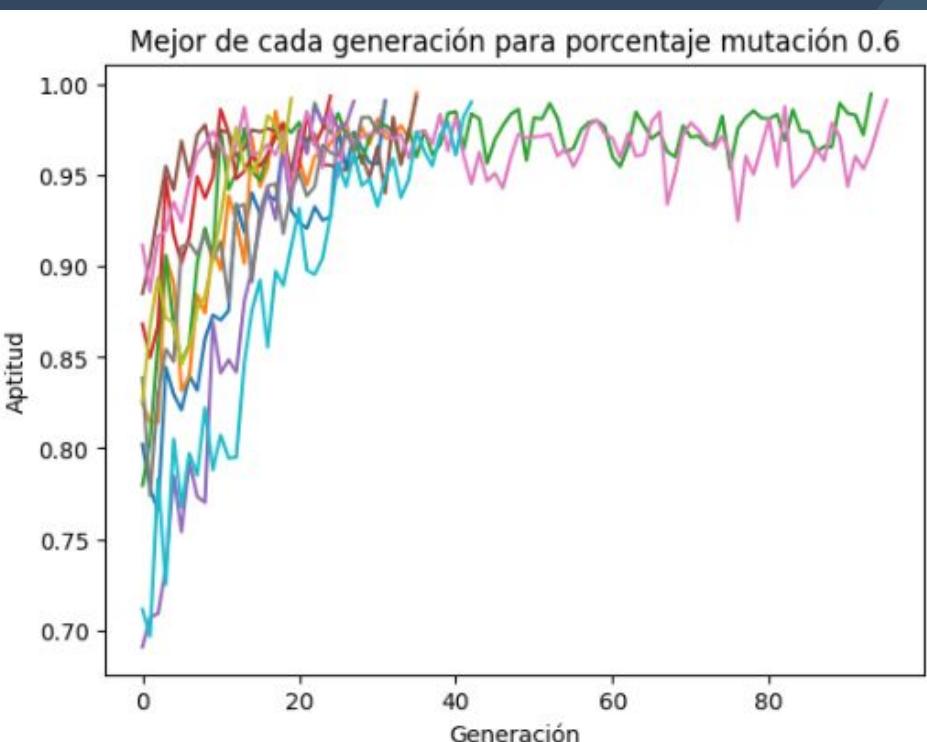
Mejor de cada generación para porcentaje mutación 0.4



Mejor de cada generación para porcentaje mutación 0.6

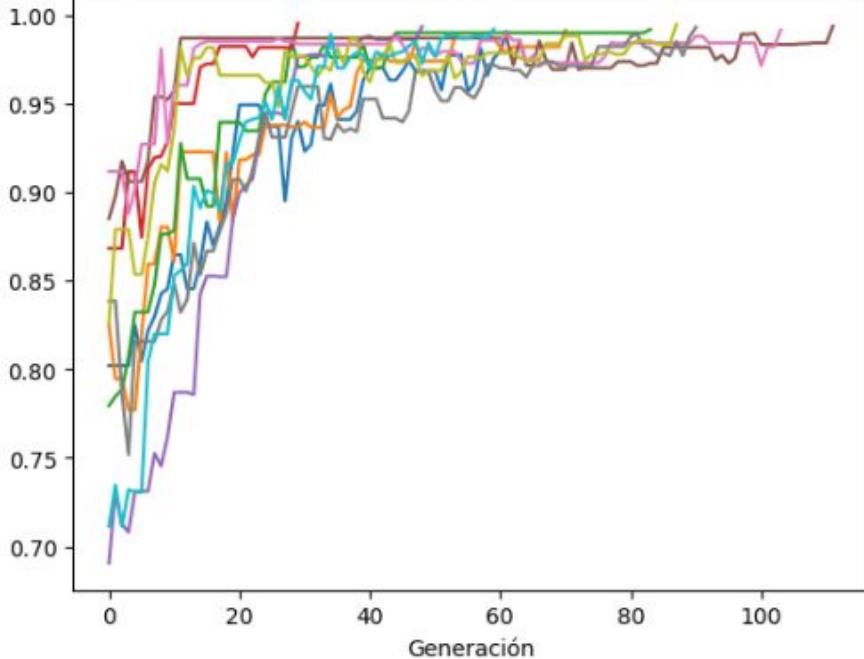


Mejor de cada generación para porcentaje mutación 0.6



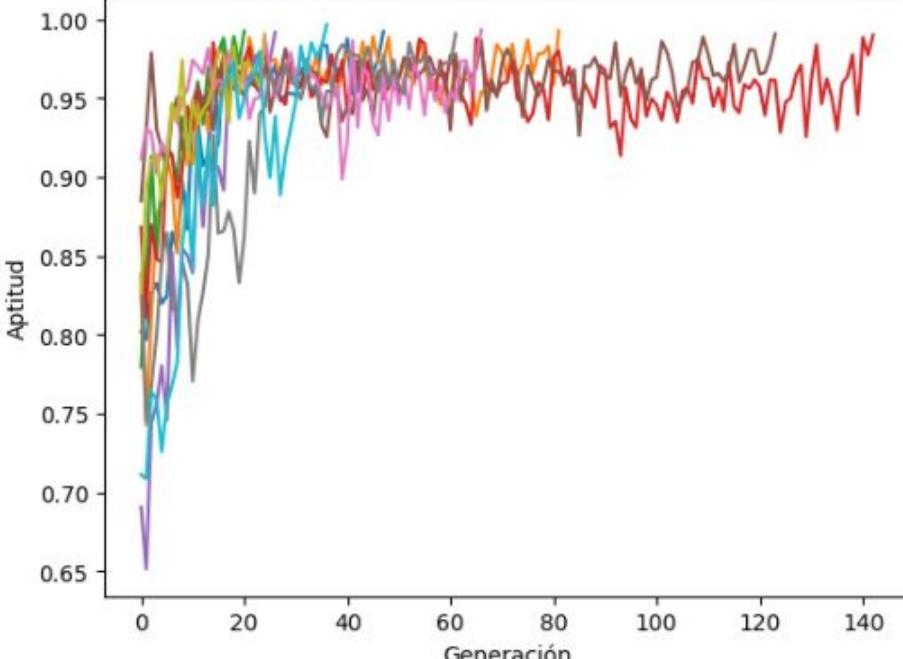
Mejor de cada generación para porcentaje mutación 0.8

Aptitud

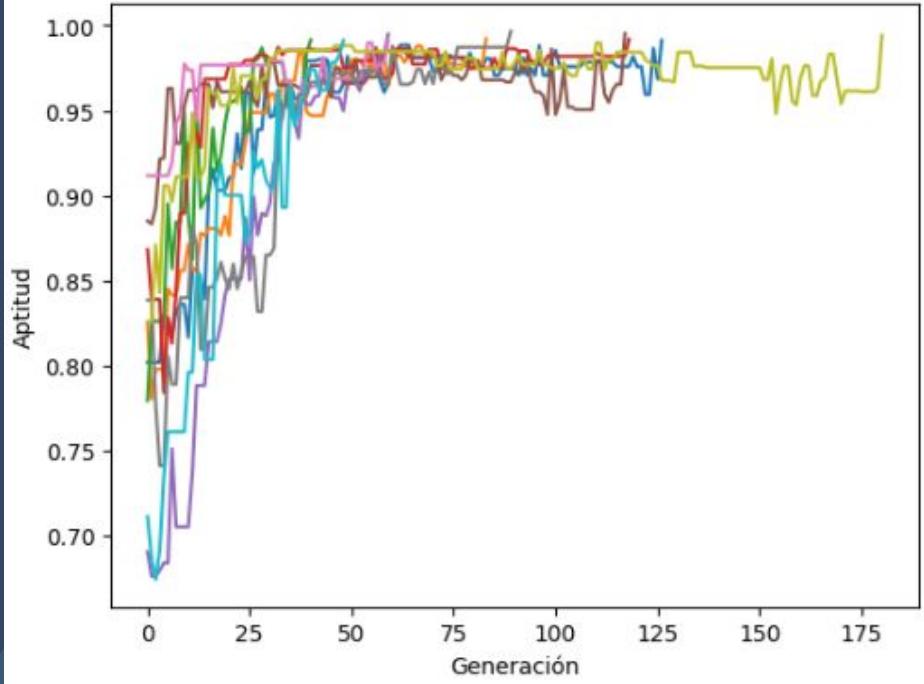


Mejor de cada generación para porcentaje mutación 0.8

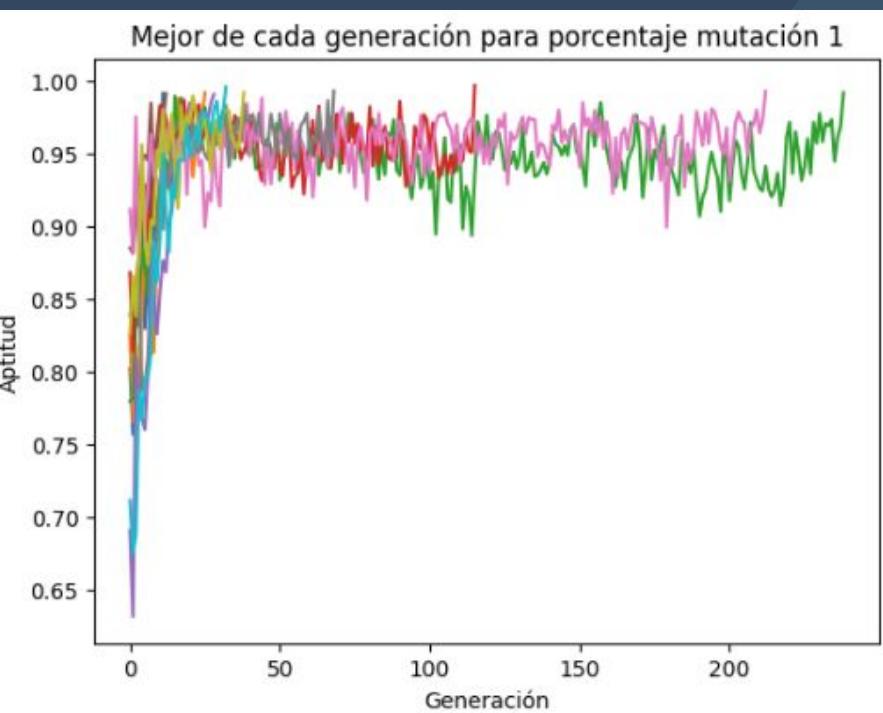
Aptitud



Mejor de cada generación para porcentaje mutación 1



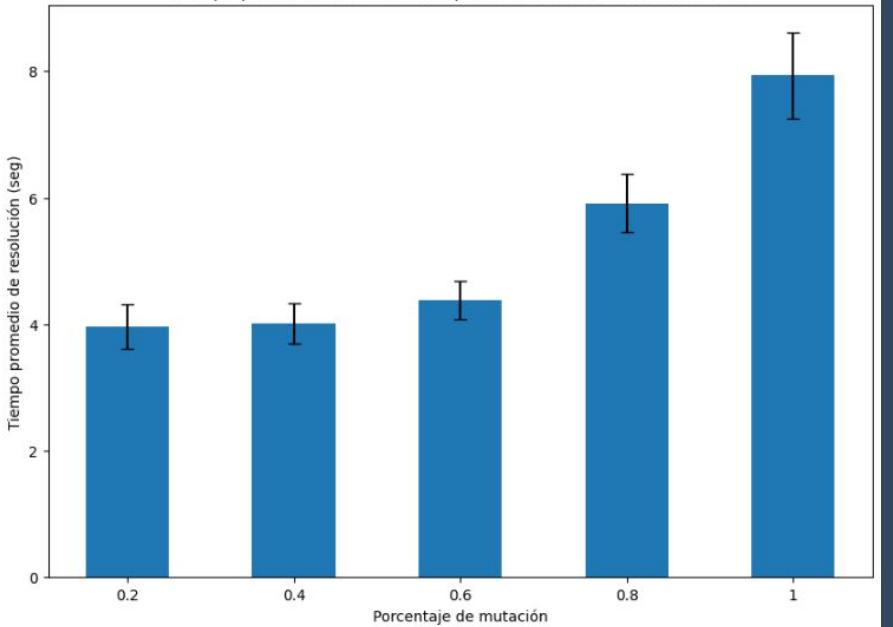
Mejor de cada generación para porcentaje mutación 1



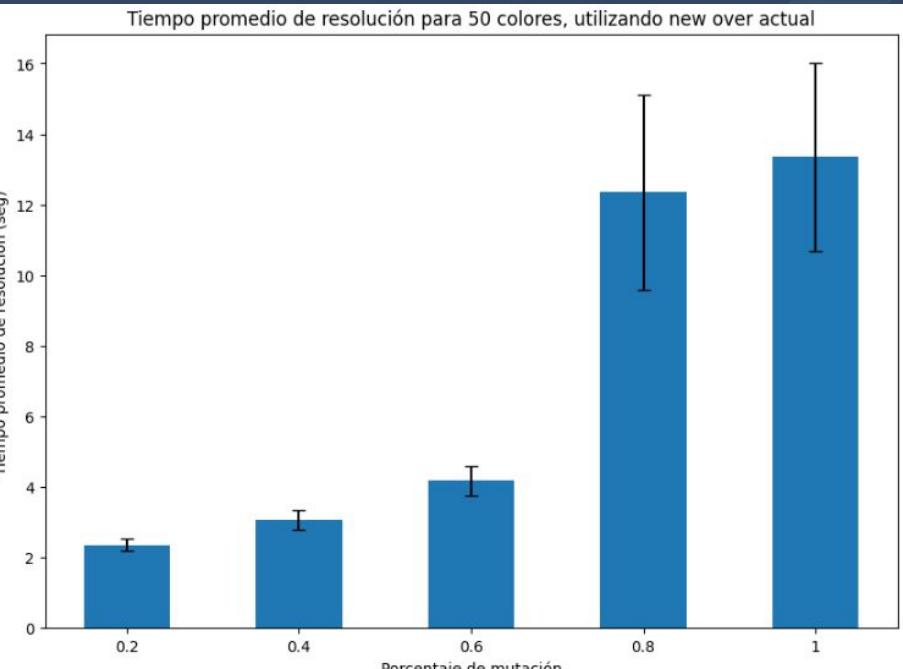
## Tiempos promedio de resolución



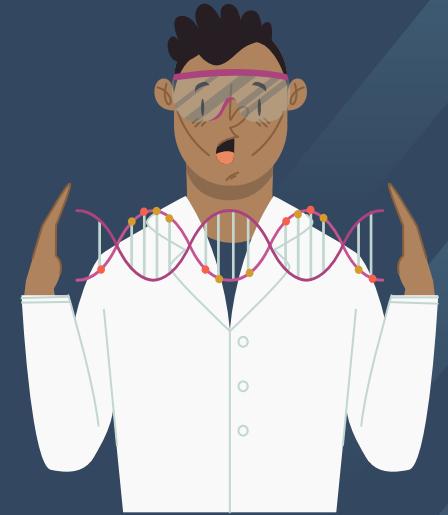
Tiempo promedio de resolución para 50 colores, utilizando use all



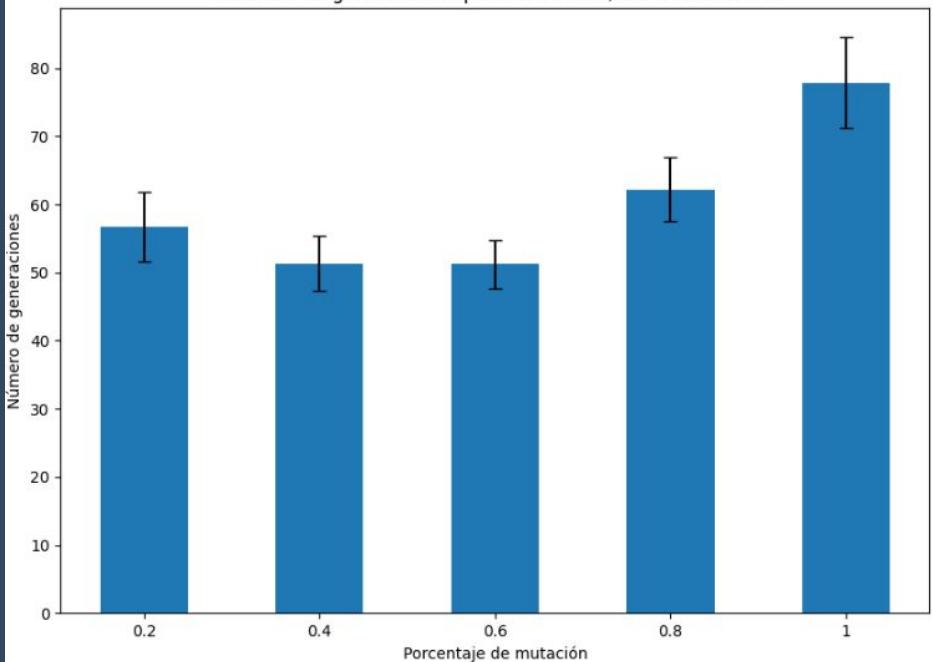
Tiempo promedio de resolución para 50 colores, utilizando new over actual



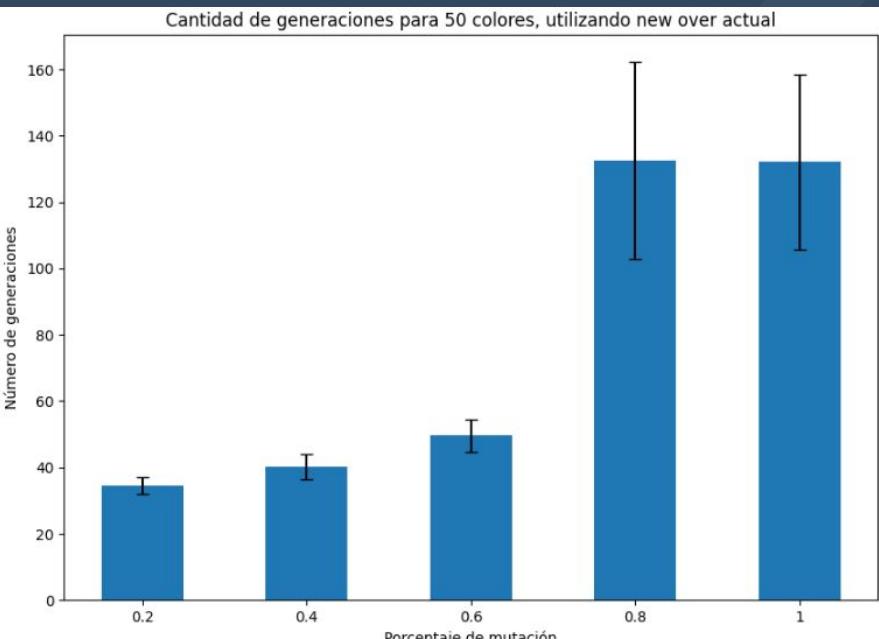
# Cantidad de generaciones



Cantidad de generaciones para 50 colores, utilizando use all



Cantidad de generaciones para 50 colores, utilizando new over actual

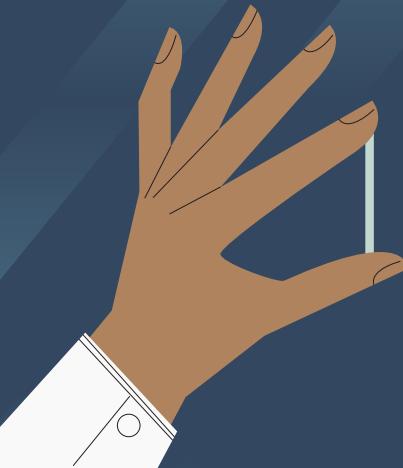




## • Conclusión general

Si tuviéramos que utilizar diariamente esta implementación de algoritmos genéticos, utilizaríamos la siguiente combinación de hiperparámetros del sistema:

1. Selección por torneos probabilísticos o ruleta
2. Cruza de un punto o doble punto
3. Mutacion multigen uniforme o completa
4. Porcentaje de mutación  $P_m=0,5$
5. Use all o new over actual, ya que ambos son performantes para las variables previamente mencionadas



# Muchas Gracias