

# Algoritmos genéticos de permutación

Stalin Muñoz Gutiérrez

Centro de Ciencias de la Complejidad  
Universidad Nacional Autónoma de México (UNAM)

Comenzaremos por platicar un poco acerca del tipo de procesos en los que se inspiran los algoritmos metaheurísticos y abordaremos un tipo de algoritmo muy interesante, el algoritmo genético para problemas de permutación. Veremos como puede aplicarse este algoritmo para resolver el problema del agente viajero.

# Algoritmos metaheurísticos

## Inspirados en la física

- 1 Templado Simulado (Simulated Annealing)
- 2 Gran Explosión Gran Colapso (Big Bang Big Crunch)
- 3 Gravitacional
- 4 Similar a Electromagnetismo
- 5 Optimización de Fuerza Central
- 6 Gotas de Agua Inteligentes
- 7 Dinámica de Formación de Ríos
- 8 Colisión de Partículas

## Algoritmos metaheurísticos

- Templado Simulado (Simulated Annealing)
- Gran Explosión Gran Colapso (Big Bang Big Crunch)
- Gravitacional
- Similar a Electromagnetismo
- Optimización de Fuerza Central
- Gotas de Agua Inteligentes
- Dinámica de Formación de Ríos
- Colisión de Partículas

Existe un gran número de algoritmos metaheurísticos.

Se pueden clasificar de acuerdo a la fuente de inspiración de los mismos.

Aquí tenemos una lista de algunos de los algoritmos basados en procesos físicos.

Entre los elementos de la lista es el algoritmo de Templado Simulado, el cual ya hemos discutido.

# Algoritmos metaheurísticos

## Inspirados en la biología

- 1 Sistema Inmune Artificial
- 2 Algoritmos Genéticos
- 3 Colonia de Hormigas
- 4 Colonia de Abejas
- 5 Algoritmo de Luciérnagas
- 6 Algoritmo de Murciélagos
- 7 Búsqueda del Cuco
- 8 Búsqueda de Cardumen

## Algoritmos metaheurísticos

- Sistema Inmune Artificial
- Algoritmos Genéticos
- Colonia de Hormigas
- Colonia de Abejas
- Algoritmo de Luciérnagas
- Algoritmo de Murciélagos
- Búsqueda del Cuco
- Búsqueda de Cardumen

Aquí tenemos otra lista de algoritmos metaheurísticos inspirados en la biología y en diferentes organismos.

Un gran número de estas metáforas se basan en abstraer la forma en que poblaciones de organismos resuelven colectivamente un problema.

Son de particular importancia los algoritmos genéticos, los cuales se inspiran en la teoría de evolución de Darwin para encontrar soluciones a problemas.

## Algoritmos metaheurísticos

### Otras fuentes de inspiración

- 1 Algoritmos Culturales
- 2 Optimización de Partículas
- 3 Sociedad Anarquista
- 4 Competitivo Imperialista
- 5 Búsqueda de Armonía
- 6 Rueda Chirriante
- 7 Optimización social cognitiva
- 8 Explosión de minas

### Algoritmos metaheurísticos

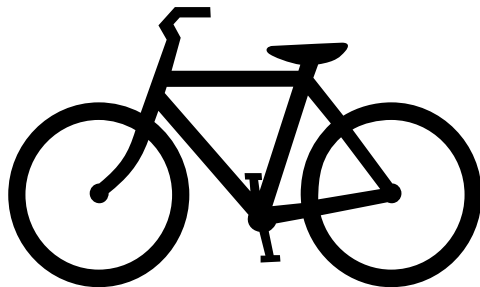
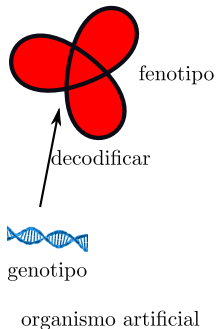
- Algoritmos Culturales
- Optimización de Partículas
- Sociedad Anarquista
- Competitivo Imperialista
- Búsqueda de Armonía
- Rueda Chirriante
- Optimización social cognitiva
- Explosión de minas

No hay límite a las posibles fuentes de inspiración.

Otros algoritmos se basan en temas culturales, políticos, artísticos, industriales, incluso en aforismos como "The squeaky wheel gets the grease" (La rueda chirriante obtiene la atención).



# Algoritmo Genético



Problema a resolver (Entorno)

## Algoritmo Genético



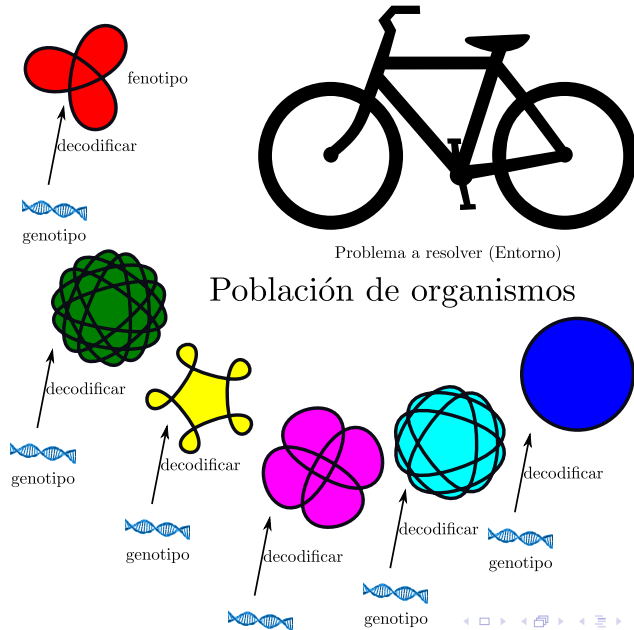
Los algoritmos genéticos fueron propuestos por John Holland en los años 60s, y fueron popularizados por David Goldberg en los años 80s.

Los algoritmos genéticos se basan en metáforas de la evolución de organismos.

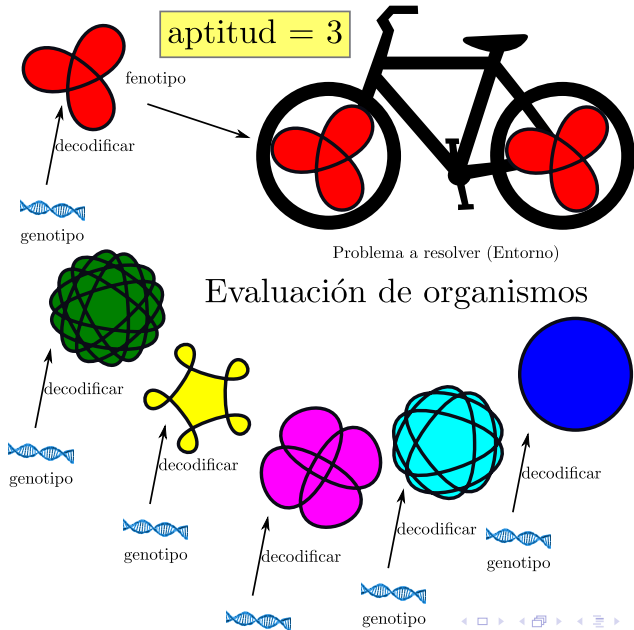
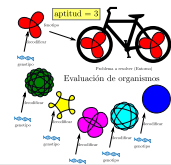
La información necesaria para describir la construcción de un organismo artificial esta codificada en su genotipo.

El genotipo se expresa o decodifica en un fenotipo el cual representa una solución candidata al problema a resolver.

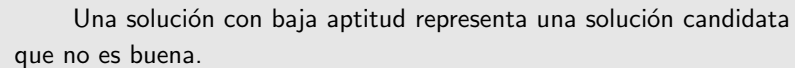
Los problemas son vistos como un entorno o medio ambiente que recompensa a las organismos más aptos. Siendo los fenotipos de los organismos artificiales las soluciones candidatas al problema.

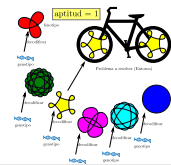
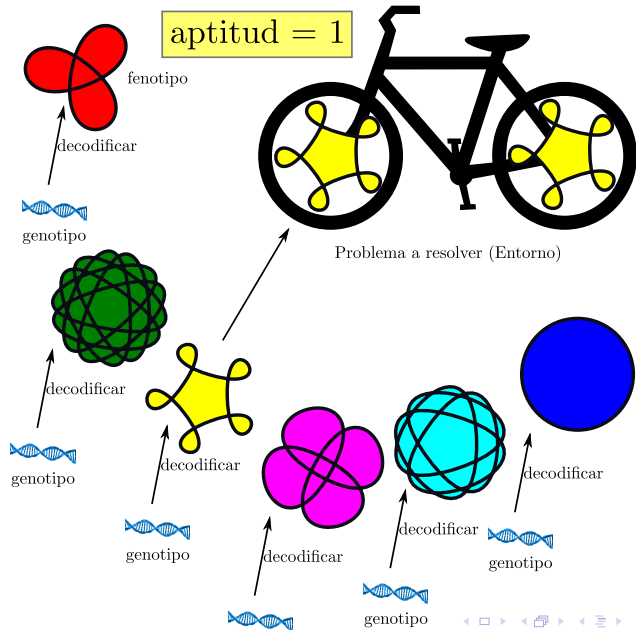


La evolución artificial opera sobre una población de organismos.

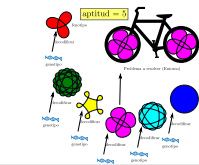
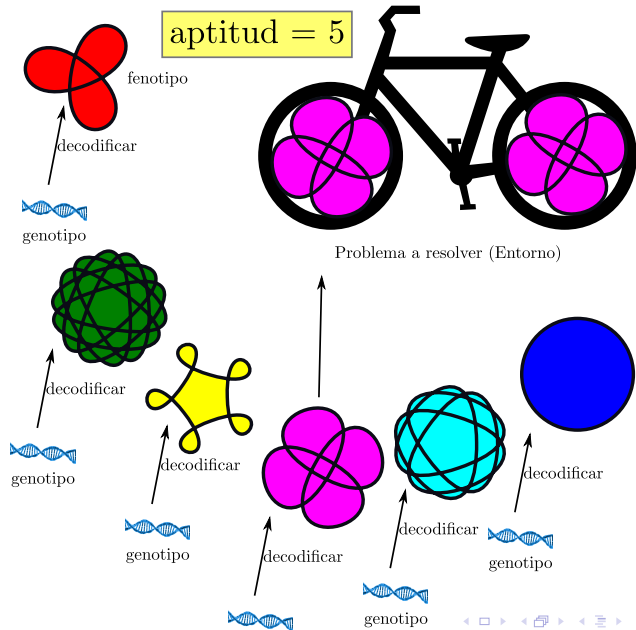


Cada individuo es evaluado en el entorno obteniendo una aptitud.  
La aptitud es más alta para individuos mejor adaptados al entorno.

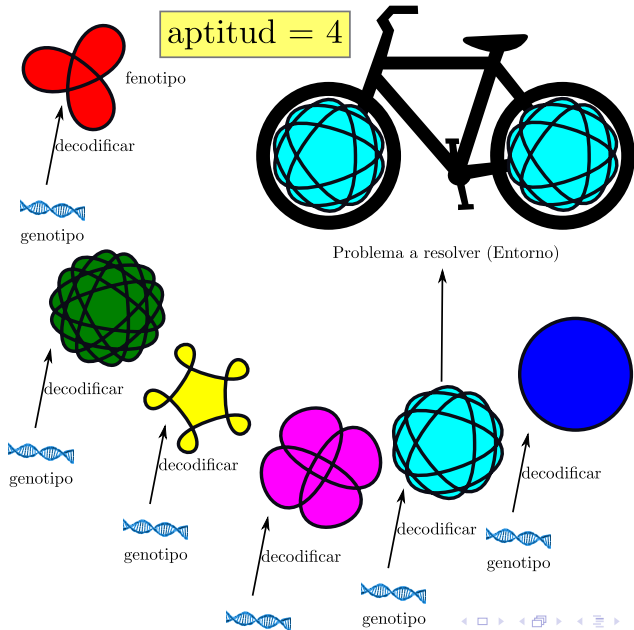
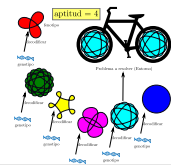




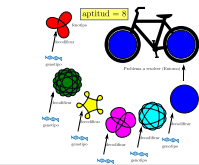
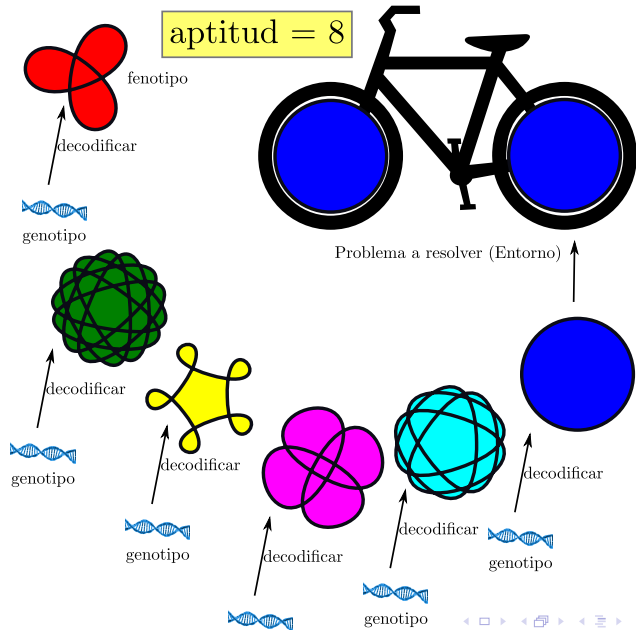
En el algoritmo, los organismos menos aptos tendrán menor probabilidad de reproducirse.



La aptitud asignada a un organismo artificial debe ser informativa para el algoritmo de selección.



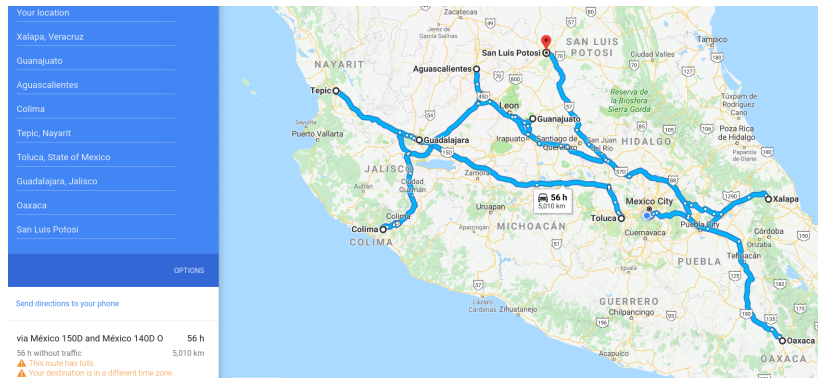
La selección artificial seleccionará a los individuos más aptos.



Sin embargo, en el algoritmo genético canónico, no hay garantía de que el mejor organismo de la población será seleccionado, ya que la selección artificial depende de una simulación estocástica.



# Algoritmos genéticos para resolver el problema del agente viajero



└ Algoritmos genéticos para resolver el problema del agente viajero



Utilizaremos el ejemplo del problema del agente viajero para ilustrar el funcionamiento de un algoritmo genético para problemas de permutación.

## Genotipo

Genotipo en un algoritmo genético canónico:

$[1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1]$

Genotipo para un algoritmo genético de permutación:

$[5 \ 3 \ 2 \ 10 \ 9 \ 7 \ 8 \ 1 \ 4 \ 6]$

## Genotipo

Genotipo en un algoritmo genético canónico:  
 $[1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1]$

Genotipo para un algoritmo genético de permutación:  
 $[5 \ 3 \ 2 \ 10 \ 9 \ 7 \ 8 \ 1 \ 4 \ 6]$

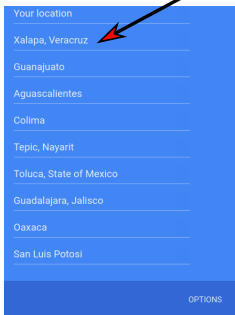
Los organismos artificiales cuentan con un genotipo.

En el algoritmo genético canónico se trata de una cadena binaria.

En un algoritmo genético de permutación la cadena esta formada por una permutación de un segmento inicial de números enteros.

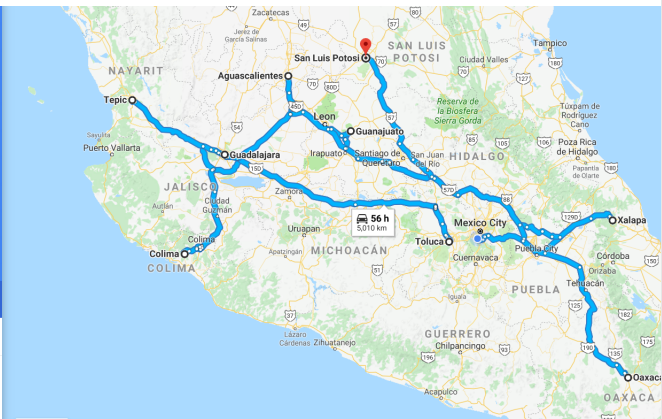
## Fenotipos

[5 3 2 10 9 7 8 1 4 6]



Send directions to your phone

via México 150D and México 140D O 56 h  
 56 h without traffic 5,010 km  
 ⚠ This route has tolls.  
 ⚠ Your destination is in a different time zone.



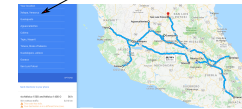
## Algoritmos genéticos de permutación

2019-08-29

└ Fenotipos

Fenotipos

[5 3 2 10 9 7 8 1 4 6]



Análogamente a un organismo biológico la información genética contiene un código que se expresa en un fenotipo. En un organismo biológico el fenotipo son los atributos físicos del organismo, su anatomía, fisiología, etc.

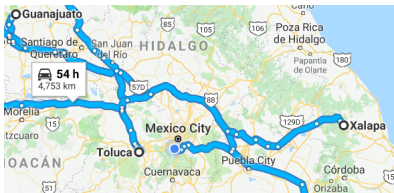
En un algoritmo genético el fenotipo es una solución candidata del problema a resolver.

En el caso del agente viajero, cada número entero representa una ciudad a visitar.

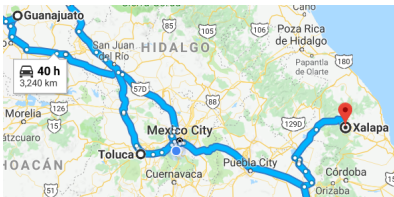
La secuencia de numeros es la ruta a tomar.

## Aptitud

solución A



solución B



$$\text{aptitud}(A) < \text{aptitud}(B)$$

## Algoritmos genéticos de permutación

2019-08-29

└ Aptitud

Aptitud

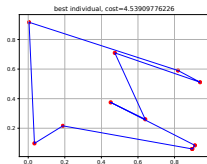


La aptitud de una ruta puede definirse en función de la distancia total de la ruta o del tiempo estimado.

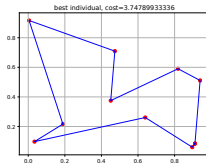
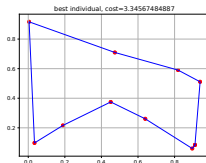
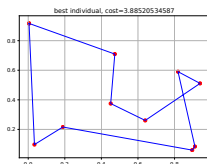
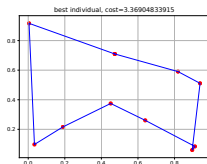
En este ejemplo la aptitud de la solución candidata A es menor que la aptitud de la solución candidata B, dado que el tiempo estimado de A es 54 horas mientras que el tiempo estimado de B es 40 horas.

Dependiendo del algoritmo de selección podemos requerir o no de una función de aptitud explícita. De ser así, una opción puede ser el inverso de la distancia o el tiempo.

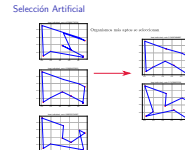
## Selección Artificial



Organismos más aptos se seleccionan



## Selección Artificial

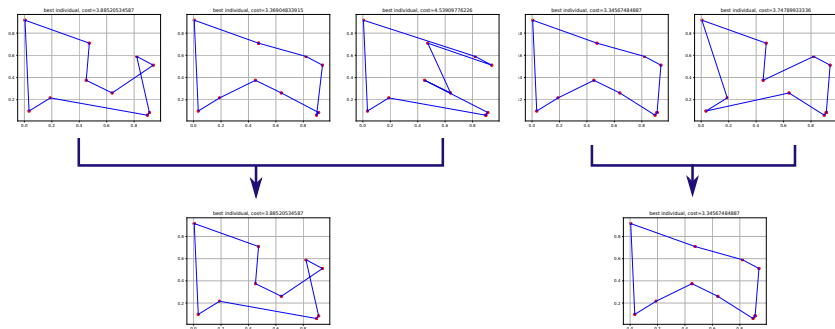


El algoritmo de selección discrimina las mejores soluciones candidatas.

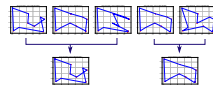
Estas soluciones serán sujetas a reproducción para generar nuevas soluciones candidatas.



## Selección tipo torneo



## Selección tipo torneo



Por ello existen otras formas de seleccionar organismos. Una muy efectiva es la selección tipo torneo.

Se define un tamaño de torneo  $k$ . Por ejemplo torneo binario o entre dos soluciones.

El torneo se forma tomando una muestra aleatoria de  $k$  soluciones de la población.

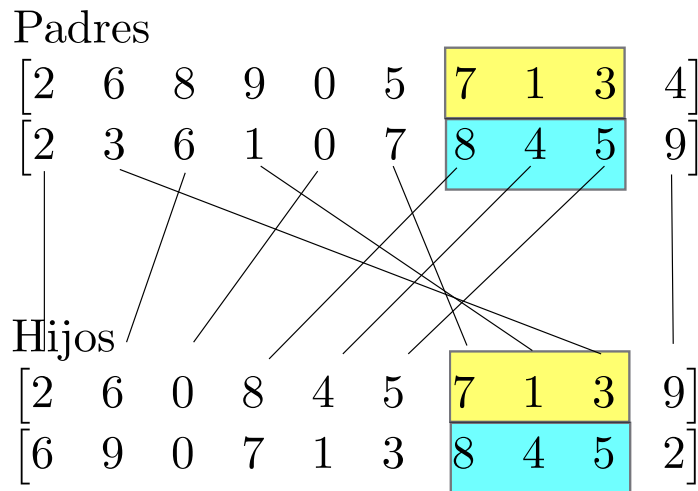
Se define como ganador del torneo el organismo con la mayor aptitud de la muestra.

Este ejemplo ilustra dos torneos binarios.

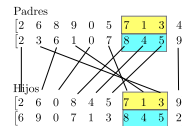
Las soluciones candidatas seleccionadas son las rutas más cortas en cada caso.

Al ser una muestra aleatoria no hay garantía de que el mejor individuo se seleccione en un torneo.

## Reproducción



## Reproducción



Las soluciones candidatas seleccionadas se reproducen para generar nuevas soluciones.

Hay muchas formas de hacer la recombinación de los genotipos.

Aquí ilustramos una muy simple. La cruce requiere dos padres y genera dos hijos.

Seleccionamos una región aleatoria de la permutación. Esta región es copiada a cada uno de los hijos. Después, considerando que el cromosoma es circular, colocamos los números que vienen de posiciones subsecuentes del otro padre, siempre que no se repitan números.

En el ejemplo el primer hijo tiene la secuencia 7,1,3 del primer padre. Después tomamos la secuencia del segundo padre a partir de esa posición. Copiamos el número 9, considerando el cromosoma circular le sigue el número 2, luego el 3, pero este se repite, por lo que no lo copiamos, luego copiamos el 6, sigue el 1, pero se repite por lo que no lo copiamos, copiamos el cero, sigue el 7 pero se repite, copiamos 8, 4 y 5.



## Mutación

### Mutación de intercambio

$$\begin{bmatrix} 2 & 6 & 8 & 9 & 0 & 5 & 7 & 1 & 3 & 4 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 7 & 8 & 9 & 0 & 5 & 6 & 1 & 3 & 4 \end{bmatrix}$$

### └ Mutación

Mutación de intercambio

$$\begin{bmatrix} 2 & 6 & 8 & 9 & 0 & 5 & 7 & 1 & 3 & 4 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 7 & 8 & 9 & 0 & 5 & 6 & 1 & 3 & 4 \end{bmatrix}$$

La mutación es otro operador genético del algoritmo.

Un ejemplo de mutación es el intercambio aleatorio de ciudades.

En el ejemplo tras la mutación se han intercambiado las posiciones de las ciudades 6 y 7.

# Evolución

En la evolución aplicamos estas operaciones en el orden siguiente:

- 1 Generar población inicial
- 2 Repetir cierto número de generaciones
  - 1 Asignar aptitud de la población actual
  - 2 Seleccionar mejores soluciones
  - 3 Cruzar las soluciones generando nueva población
  - 4 Mutar la nueva población
  - 5 La población actual es la nueva población

## Evolución

En la evolución aplicamos estas operaciones en el orden siguiente:

- 1 Generar población inicial
- 2 Repetir cierto número de generaciones
  - 1 Asignar aptitud de la población actual
  - 2 Seleccionar mejores soluciones
  - 3 Cruzar las soluciones generando nueva población
  - 4 Mutar la nueva población
  - 5 La población actual es la nueva población

Aquí un algoritmo genético muy simple.

Generamos la población inicial con cromosomas aleatorios.

Repetimos los pasos siguientes cierto número de generaciones.

Decodificamos y evaluamos a los organismos en el problema para obtener aptitud.

Seleccionamos a los mejores.

Reproducimos a los organismos seleccionados.

Mutamos a los organismos.

Hacemos la población actual igual a la nueva población.

## Comentario final.



Curso: Cómputo evolutivo,  
Especialización: Introducción a la inteligencia artificial de la UNAM.

## Algoritmos genéticos de permutación

2019-08-29

└ Comentario final.

Comentario final.



Curso: Cómputo evolutivo,  
Especialización: Introducción a la inteligencia artificial de la UNAM.

Los algoritmos genéticos son metaheurísticos muy poderosos.  
Aquí solo presentamos una versión muy simple.  
Hay muchas variaciones de estos algoritmos.  
Si quieres saber más respecto de estos algoritmos te recomendamos tomar el curso:  
Cómputo evolutivo,  
dentro de la especialización Introducción a la inteligencia artificial de la UNAM.