



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



FIME

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica
y Eléctrica

Producto Integrador de Aprendizaje.

Manual de Usuario de Algoritmos Genéticos

Nombre: Ranfery Josua Peregrina Morales

Matrícula: 1924910

Semestre: Décimo

Materia: Temas Selectos de IA

Docente: Raymundo Said Zamora Pequeño

Grupo: 001

Fecha: 25/05/2024

Periodo: *Enero – Junio 2024*



Índice

Introducción.....	2
Propósito del Software.....	2
Público objetivo del manual.....	2
Visión general de las funcionalidades.....	3
Primeros pasos.....	3
Instalación y Configuración Inicial:.....	3
Cómo Iniciar el Software.....	3
Requisitos Mínimos del Sistema:.....	3
Guía de Uso.....	4
Funcionalidades generales.....	4
Preguntas Frecuentes:.....	6
Glosario de Términos:	7
Bibliografía.....	8

Introducción.

Este proyecto se desarrolla como parte de un trabajo escolar con el objetivo de explorar los principios fundamentales de los algoritmos genéticos, una heurística inspirada en la teoría de la evolución de Charles Darwin[1]. Usada para resolver problemas de optimización. En este caso de combinatoria. En donde se busca evaluar el desempeño de un conjunto de soluciones, y reproducir sus características seleccionando las mejor calificadas. Esperando tener una combinación de características útiles.

Propósito del Software

El propósito de este software es encontrar la mejor solución entre una población de soluciones posibles, aplicando conceptos evolutivos a la búsqueda de la solución óptima.

Cada solución en la población se representa como una cadena de 8 dígitos binarios, que a su vez se divide en dos mitades de 4 dígitos.

Cada mitad se evalúa con una serie de reglas acerca de la interacción entre los dígitos que forman dicha mitad.

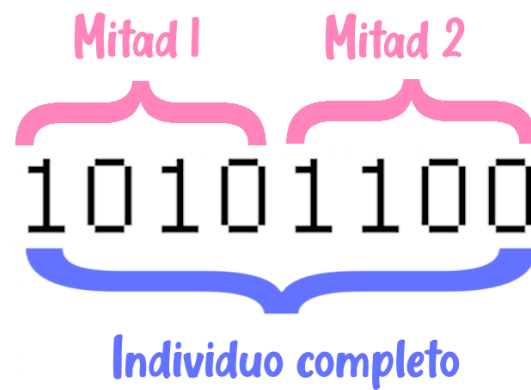
Las reglas de interacción entre estos dígitos son fundamentales para la evaluación de cada solución:

- Si hay un 1 en la cadena, se suma 1 a la puntuación.
- Si hay dos 1s seguidos en la cadena, se suma 1 a la puntuación.
- Si hay más 1s que 0s en la cadena, se suma 1 a la puntuación.
- Si hay un 0 en la cadena, se resta 1 a la puntuación.
- Si hay dos 0s seguidos en la cadena, se resta 1 a la puntuación.
- Si hay más 0s que 1s en la cadena, se resta 1 a la puntuación.

Y el programa está hecho para generar su propio conjunto de soluciones iniciales, y modificarlas (aplicando todo el proceso de un algoritmo genético) repetidas veces para buscar mejorar el conjunto de soluciones.

Público objetivo del manual.

El programa fue hecho para ser evaluado por el profesor Raymundo Said Zamora Pequeño para la clase de *Temas Selectos de IA*. En la *Facultad de Ingeniería Mecánica y Eléctrica* de la **Universidad Autónoma de Nuevo León**. Así que el manual es primeramente dirigido al docente que lo evalúa, pero en caso de ser necesario. También a la/las personas que piensen usar el programa.



Visión general de las Funcionalidades.

El software de Algoritmos Genéticos ofrece una variedad de funcionalidades diseñadas para permitir a los usuarios realizar experimentos, análisis y optimizaciones utilizando algoritmos genéticos. Estas funcionalidades incluyen:

1. **Generación de Población:** El software permite generar una población inicial de soluciones binarias aleatorias para iniciar el proceso de evolución.
2. **Evaluación de la Población:** Se evalúa cada solución de la población utilizando un conjunto predefinido de reglas de interacción entre los dígitos binarios.
3. **Selección de Individuos:** Selecciona individuos de la población actual para la reproducción, basándose en su puntuación y aplicando técnicas de selección.
4. **Cruce de Individuos:** Los individuos seleccionados se cruzan para producir una nueva generación de soluciones, combinando características de los padres.
5. **Mutación de Individuos:** Se aplica una mutación aleatoria a ciertos individuos de la población para introducir variabilidad genética y evitar la convergencia prematura.
6. **Evaluación de Nueva Generación:** Se evalúan los individuos de la nueva generación para determinar su puntuación y ajustar la población en consecuencia.
7. **Finalización del Proceso:** El proceso de evolución continúa durante un número especificado de iteraciones o hasta que se cumpla un criterio de finalización predefinido.

Primeros pasos

Instalación y Configuración Inicial:

Para ejecutar el software de Algoritmos Genéticos, asegúrate de tener R instalado en tu sistema. Puedes descargar e instalar R desde el sitio web oficial de R (<https://www.r-project.org/>). Una vez que R esté instalado, simplemente abre el archivo de script "PIA.r" en tu entorno de desarrollo preferido para comenzar. Yo usé "RStudio" y no lo probé en ningún otro. Así que recomiendo RStudio.

Cómo Iniciar el Software:

Para iniciar el software, abre el archivo de script "PIA.r" utilizando RStudio, Jupyter Notebook u otro entorno de desarrollo compatible con R. Una vez abierto el script, puedes ejecutarlo línea por línea o en su totalidad, según tu preferencia. Teniendo en cuenta que el código primero carga librerías, luego inicializa 2 variables iniciales, y después define cada función. Y hay funciones que dependen de otras. Por lo que puedes ejecutar la declaración de las funciones por separado, pero se recomienda ejecutar todas juntas. (Es decir, todo el programa)

Requisitos Mínimos del Sistema:

R: Asegúrate de tener R instalado en tu sistema. El software de Algoritmos Genéticos está escrito en R y requiere un entorno de ejecución de R para funcionar correctamente.

Entorno de Desarrollo: Se recomienda el uso de un entorno de desarrollo integrado (IDE) como RStudio para una experiencia de programación más cómoda, pero no es obligatorio. Otros IDEs o editores de texto con soporte para R también son compatibles.

Recursos del Sistema: Los requisitos de recursos del sistema dependerán del tamaño de los conjuntos de datos y la complejidad de los cálculos realizados por el script. En general, se recomienda disponer de suficiente memoria RAM y espacio en disco para manejar los datos y ejecutar los cálculos sin problemas. El programa al ejecutarse toma del entorno un mínimo de 133MBs de memoria RAM.

Guía de Uso

El software de Algoritmos Genéticos implementado en R es una herramienta de línea de comandos que permite ejecutar algoritmos genéticos para optimizar soluciones basadas en conjuntos de datos binarios.

Instrucciones:

1. Abre el software de RStudio o cualquier otro entorno de desarrollo compatible con R en tu sistema. (Pero este manual está hecho para RStudio)
2. Carga el archivo de script "PIA.r" que contiene el código del programa de Algoritmos Genéticos.
3. Ejecuta el script haciendo clic en el botón de ejecución o utilizando el atajo de teclado correspondiente. Para el caso de RStudio, es la teclada de Enter una vez pegado el código o abierto el archivo en la ventana de Terminal.
4. Durante la ejecución del script, se imprimirán mensajes en la consola que indican el progreso del algoritmo genético, incluyendo la población inicial, la puntuación después de cada iteración y cualquier mensaje de error o advertencia que pueda surgir.
5. Una vez que el script haya completado su ejecución, se imprimirá la población final junto con cualquier otra información relevante, como la puntuación máxima alcanzada y la iteración en la que se alcanzó.

Flujo de Trabajo Básico: El flujo de trabajo básico para usar el software de Algoritmos Genéticos consiste en cargar el script en un entorno de desarrollo de R, ejecutar el script y revisar los resultados en la consola de RStudio o en la salida de la línea de comandos. El usuario no necesita interactuar con ninguna interfaz gráfica o realizar acciones adicionales más allá de ejecutar el script.

Funcionalidades generales.

Dado que el programa es un programa de consola y su única función es ser ejecutado, detallaré la funcionalidad de cada una de las líneas que contiene el programa. Comenzando por:

Generación de la Población Inicial:

Esta función se encarga de crear la población inicial de individuos para el algoritmo genético. Para ello, toma en cuenta la cantidad deseada de individuos y la longitud que se espera para cada uno. Además, implementa las validaciones necesarias para asegurar que los individuos generados cumplan con los requisitos establecidos. Por ejemplo, si se especifica una longitud menor a 8 para los individuos, el programa ajustará automáticamente la longitud a 8. Asimismo, se emiten advertencias en caso de que la población sea demasiado pequeña para el funcionamiento óptimo del algoritmo.

Cálculo de la Calificación de la Población:

La calificación de cada población se determina evaluando a cada uno de sus individuos. Para ello, se aplican una serie de reglas basadas en las interacciones binarias dentro de cada individuo. Estas reglas consideran la presencia de unos y ceros, así como la secuencia de unos o ceros en la cadena. A continuación, se detallan las reglas utilizadas para calcular la calificación de cada individuo:

- Si hay un 1 en la cadena: se suma 1 a la calificación.
- Si hay 2 unos seguidos en la cadena: se suma 1 a la calificación.
- Si hay más unos que ceros en la cadena: se suma 1 a la calificación.
- Si hay un 0 en la cadena: se resta 1 a la calificación.
- Si hay 2 ceros seguidos en la cadena: se resta 1 a la calificación.
- Si hay más ceros que unos en la cadena: se resta 1 a la calificación.

Este proceso se aplica a todos los individuos de la población, y la calificación total de la población se calcula sumando las calificaciones individuales.

Selección y Cruce de Individuos:

Después de calcular las calificaciones de la población, se seleccionan los individuos mejor calificados para el cruce. En este proceso, se toman las dos mitades de los individuos seleccionados y se combinan para formar nuevos individuos en una nueva población.

El proceso de selección se basa en elegir los individuos con las calificaciones más altas. Estos individuos se denominan "mejores individuos". La cantidad de mejores individuos seleccionados suele ser una fracción de la población total y se determina según un criterio preestablecido.

Una vez seleccionados los mejores individuos, se procede a realizar el cruce. Este proceso implica tomar las dos mitades de cada uno de los mejores individuos y combinarlas para formar nuevos individuos en una nueva población. Por ejemplo, la primera mitad de un individuo se combina con la segunda mitad de otro individuo para crear un nuevo individuo en la población.

Este proceso de selección y cruce se repite hasta generar una nueva población con el mismo tamaño que la población original. De esta manera, se busca mantener la diversidad genética y encontrar soluciones potencialmente mejores en cada iteración.

Mutación de la Población:

Después de la fase de cruce, se aplica un proceso de mutación a toda la población resultante. Este proceso consiste en cambiar un bit en cada individuo de la población, de acuerdo con su posición. Es decir, al primer individuo se le cambia el primer bit, al segundo individuo se le cambia el segundo bit, y así sucesivamente.

Si la población tiene más de 8 individuos, Por ejemplo, que se tenga una población de 9 individuos, al tener que cambiar el "noveno bit", se regresa al primero y se continúa alternando hasta completar la mutación de todos los individuos.

La mutación es un componente importante del algoritmo genético, ya que introduce variabilidad en la población y ayuda a evitar la convergencia prematura hacia una solución subóptima o atascarnos en óptimos locales.

Repetición del Proceso:

Una vez que se ha aplicado la mutación a la población y se han calificado los individuos resultantes, se guarda esa calificación como la calificación de la iteración actual. A continuación, se repite todo el proceso utilizando esta población mutada como la población inicial de la siguiente generación.

Este proceso se repite hasta que se alcance el límite máximo de iteraciones declaradas, o hasta que no haya una mejora en la calificación después de cierta cantidad de intentos consecutivos, según el límite establecido.

La repetición del proceso garantiza que se explore exhaustivamente el espacio de soluciones y se busque continuamente mejorar la calidad de la solución encontrada.

Preguntas Frecuentes:

Nadie me preguntó nada, nadie ha leído este manual ni usado este programa además de mí. Pero anoté una serie de preguntas que podrían surgir comúnmente.

1. ¿Qué hacer si la población inicial no se genera correctamente debido a restricciones de longitud?

- *Respuesta:* Si la población inicial no se genera correctamente debido a restricciones de longitud, se recomienda revisar los parámetros de entrada, asegurándose de que el tamaño de la población y la longitud de cada individuo sean coherentes con las restricciones establecidas en el programa. Además, se puede ajustar el tamaño de la población y la longitud de los individuos según sea necesario para evitar este problema. Aunque el programa tiene un algoritmo de validación para generar datos iniciales coherentes.

2. ¿Qué hacer si la población inicial es demasiado pequeña para generar una diversidad genética adecuada?

- *Respuesta:* Si la población inicial es demasiado pequeña para generar una diversidad genética adecuada, se recomienda aumentar el tamaño de la población para permitir una exploración más exhaustiva del espacio de soluciones. Aumentar el tamaño de la población ayudará a mejorar las posibilidades de encontrar una

solución óptima. Aunque el programa tiene un algoritmo de validación para expandir los datos cuando sea estrictamente necesario.

3. ¿Cómo interpretar la salida del programa durante la ejecución?

- *Respuesta:* Durante la ejecución del programa, la salida mostrará información sobre cada iteración, incluida la población actual, la puntuación inicial, las mutaciones realizadas y la puntuación después de la mutación. Además, se proporcionará una gráfica que muestra la evolución de la puntuación en cada iteración, lo que permite visualizar el progreso del algoritmo genético.

Contacto para Soporte Técnico:

Para obtener soporte técnico adicional o resolver problemas específicos relacionados con la ejecución del programa, no dude en contactar al desarrollador a través del correo electrónico: ranfery_morales@hotmail.com

Glosario de Términos:

1. **Algoritmo Genético (AG):** Un algoritmo de optimización inspirado en el proceso de selección natural. Utiliza conceptos evolutivos como la selección natural, la herencia genética y la mutación para resolver problemas de optimización.
2. **Población:** Un conjunto de soluciones candidatas o individuos que representan posibles soluciones al problema de optimización.
3. **Individuo:** Una solución candidata dentro de la población, representada por un conjunto de genes o cromosomas que codifican una posible solución al problema.
4. **Cromosoma:** Una estructura de datos que representa un individuo en el algoritmo genético. Consiste en una cadena de genes que codifican características específicas de la solución.
5. **Gen:** Un componente de un cromosoma que codifica una característica específica de la solución. En un contexto binario, un gen puede ser un bit que representa una variable del problema.
6. **Función de Aptitud (Fitness):** Una función que evalúa la calidad de una solución candidata en función de su capacidad para resolver el problema de optimización. Cuanto mayor sea la aptitud, mejor será la solución.
7. **Selección:** El proceso de elegir individuos de la población actual para reproducirse y crear la siguiente generación de soluciones.
8. **Cruce (Crossover):** El proceso de recombinar la información genética de dos individuos seleccionados para crear nuevos individuos.
9. **Mutación:** El proceso de introducir cambios aleatorios en los genes de los individuos para aumentar la diversidad genética y explorar nuevas soluciones.

10. **Iteración:** Una sola ejecución del ciclo de vida de un algoritmo genético, que incluye la selección, el cruce, la mutación y la evaluación de la población.
11. **Convergencia:** El estado en el que la población converge hacia una solución óptima o subóptima después de varias iteraciones del algoritmo genético.
12. **Límite de Iteraciones:** El número máximo de iteraciones permitidas antes de que el algoritmo genético finalice su ejecución.
13. **Límite de Iteraciones sin Mejora:** El número máximo de iteraciones consecutivas sin mejora en la función de aptitud antes de que el algoritmo genético finalice su ejecución.
14. **Optimización:** El proceso de encontrar la mejor solución posible para un problema específico, generalmente maximizando o minimizando una función objetivo.
15. **Espacio de Búsqueda:** El conjunto de todas las posibles soluciones al problema de optimización.
16. **Solución Óptima:** La mejor solución posible dentro del espacio de búsqueda, que maximiza o minimiza la función objetivo.
17. **Solución Subóptima:** Una solución que no es la mejor posible, pero que aún cumple con los criterios de la función objetivo.
18. **Graficación:** La representación visual de datos, como la evolución de la función de aptitud a lo largo de las iteraciones del algoritmo genético, para comprender mejor el comportamiento del algoritmo y el progreso hacia una solución óptima.

Bibliografía:

- [1].- Darwin, C. (1859). *El origen de las especies por la selección natural*.
- [2].- Cadenas binarias - Referencia de la API Web | MDN. (s. f.). MDN Web Docs.
<https://developer.mozilla.org/es/docs/Web/API/btoa>
- [3].- Pérez, V. C. S. y. P. J. (2018, 3 diciembre). *Curso de Introducción a R*.
<https://www.uv.es/vcoll/primeros-pasos.html>