

# Creación de dietas mediante AEs

Santiago Pereira

Facultad de Ingeniería, Universidad de la Republica  
Montevideo, Uruguay  
santiagopereira0812@gmail.com

Jorge Miguel Machado

Facultad de Ingeniería, Universidad de la Republica  
Montevideo, Uruguay  
jorge.machado.ottonelli@gmail.com

**Abstract**—In the pursuit of personalized and balanced dietary planning, this study explores the application of Evolutionary Algorithms (EAs) for generating customized meal plans. Addressing the complex task of satisfying daily and weekly nutritional needs while ensuring dietary variety, the proposed EA-based approach optimizes menus for individuals. The algorithm integrates considerations of caloric intake, proteins, carbohydrates, and fats, balancing the nutritional requirements with a strategic selection of foods suitable for specific meal times. The results demonstrate the efficiency of the proposed EA in producing nutritionally balanced and diverse meal plans tailored to individual preferences and dietary restrictions.

**Index Terms**—evolutionary algorithms, personalized diet planning, nutritional optimization, genetic algorithms, dietary restrictions, optimization techniques.

## I. INTRODUCCIÓN

En una época en la que la conciencia de la salud y la nutrición es cada vez más prominente, la planificación de dietas equilibradas y personalizadas se ha convertido en un objetivo fundamental para individuos y profesionales de la salud por igual. Uno de los desafíos más comunes y cruciales en este contexto es la generación de una dieta que satisfaga las necesidades nutricionales y energéticas de una persona, en términos de un objetivo específico de calorías diarias.

Al planificar dietas, la diversidad en la selección de alimentos se convierte en un criterio esencial. No solo se trata de cumplir con las necesidades nutricionales y energéticas individuales, sino también de crear menús variados.

Este problema no solo se relaciona con la salud y el bienestar de las personas, sino que también tiene un impacto directo en la prevención de enfermedades crónicas y en la promoción de un estilo de vida saludable.

## II. DESCRIPCIÓN DEL PROBLEMA

Las personas que desean mantener una vida saludable a menudo se cuestionan como obtener un menú para cada una de las comidas del día (desayuno, almuerzo, merienda y cena), de manera que se acerque a su cantidad de calorías diarias requeridas y que además sea una dieta balanceada.

Dado el número de calorías que cada persona debe de consumir al día, buscar partir de ello hacer menús para cada una de las comidas del día, las cuales estén balanceadas y cumplan con el requisito de que en conjunto se acerquen lo más posible al número de calorías diarias requeridas. Para definir el problema de forma más concreta se necesita la definición de ciertos conceptos:

- *Calorías diarias*: cantidad de calorías a consumir en un día.
- *Proteínas diarias*: cantidad de proteínas a consumir en un día.
- *Carbohidratos diarios*: cantidad de carbohidratos a consumir en un día.
- *Grasas diarias*: cantidad de grasas a consumir en un día.
- *Ranking de comida en el día*: beneficio de comer una comida según el horario.

De igual forma, se considera la variedad en la selección de alimentos en la planificación de dietas personalizadas. Se busca evitar la repetición excesiva de comidas a lo largo de la dieta para proporcionar una experiencia diversa y agradable. Esta variedad no solo mejora la satisfacción, sino que también asegura una ingesta equilibrada de diferentes nutrientes y minimiza el riesgo de deficiencias nutricionales asociadas con la monotonía dietética.

## III. JUSTIFICACIÓN DE UTILIZAR ALGORITMOS EVOLUTIVOS

El uso de AE esta justificado por la naturaleza compleja y multiobjetivo del problema a resolver. El cual implica cumplir con las necesidades diarias y semanales de calorías, así como asegurar un equilibrio adecuado entre proteínas, carbohidratos y grasas.

De forma adicional, el problema presenta restricciones y objetivos múltiples conflictivos, ya que es necesario equilibrar la necesidad de cumplir con los requisitos diarios y semanales, mientras se evita la repetición excesiva de alimentos en el menú. Los operadores evolutivos, como la mutación y la recombinación, permiten introducir variabilidad y diversidad en la población de soluciones, permitiendo encontrar soluciones óptimas y diversas.

Por otra parte, el proceso de planificación de dietas no es estático; las necesidades nutricionales pueden cambiar con el tiempo, y es necesario adaptarse dinámicamente a estos cambios.

## IV. ESTRATEGIA DE RESOLUCIÓN

### A. Método de inicialización

Para la población inicial se toman alimentos del dataset [5] de forma aleatoria, de 100 individuos. Para la asignación aleatoria se utilizó la semilla propia de JMetalPy [3], que cambia en cada ejecución.

### B. Representación de las soluciones

La representación elegida para las soluciones candidatas es una matriz  $4 \times 7 \times 4$ , donde cada una de las filas corresponde con una de las cuatro comidas diarias (desayuno, almuerzo, merienda, cena) en ese orden y cada columna corresponde a un día de la semana. Cada celda contiene un conjunto de máximo cuatro valores, que representan un alimento, siendo una comida un conjunto de alimentos.

Día	lun.	mar.	miérc.	juev.	vier.	sáb.	dom.
Desayuno	[1,2,3,4]						
Almuerzo							
Merienda							
Cena							

TABLE I

EJEMPLO DE LA REPRESENTACIÓN DE UN INDIVIDUO

### C. Función de fitness

Contamos con dos funciones objetivos, las cuales se buscaran minimizar (Fitness y Variety).

Para evaluar el Fitness de cada individuo se utiliza la siguiente función:

$$Fitness = \|Fit\_col(col1), Fit\_col(col2), \dots, Fit\_col(col7)\|$$

, donde  $Fit\_col$  está definida de la siguiente forma:

$$Fit\_col = \alpha * |kc - c.kcal| + \beta * [|p - c.p| + |hc - c.hc| + |g - c.g|] + \gamma * (1 - pond\_horario(c, c.h))$$

Además, definimos Variety como:

$$Variety = \delta * cant\_rep\_dia(c, d, indiv) + \sigma * cant\_rep\_semana(c, indiv)$$

, donde:

- $\alpha, \beta, \gamma, \delta, \sigma$  = hiperparámetros a definir
- $c$  = comida elegida en un horario
- $kc$  = calorías objetivo.
- $p$  = proteínas objetivo.
- $hc$  = carbohidratos objetivo.
- $g$  = grasas objetivo.
- $h$  = horario en que se come.
- $pond\_horario(c, c.h)$  retorna que tan bien se ajusta una comida a un horario.
- $cant\_rep\_semana(c, indiv)$  retorna la cantidad de veces que aparece 'c' en la semana.
- $cant\_rep\_dia(c, d, indiv)$  retorna la cantidad de veces que aparece 'c' en el día 'd'.

### D. Condición de parada

- Alcanzar la cantidad máxima de generaciones. Esta condición es necesaria para que la ejecución del algoritmo termine.

### E. Operadores evolutivos

- **Mutación:** Seleccione una celda aleatoria de un individuo y cambio su valor, reemplazándolo por otro alimento disponible.

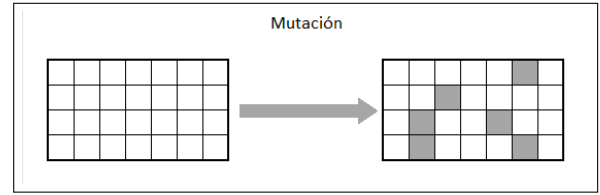


Fig. 1. Mutación de un individuo

- **Cruzamiento:** a continuación se presentan distintos tipos de cruzamiento que serán probados durante las diferentes configuraciones del algoritmo:
  - Para cada columna, seleccionar con probabilidad  $p$  la columna del mismo día del padre 1, y con probabilidad  $(1-p)$  la columna del mismo día del padre 2.

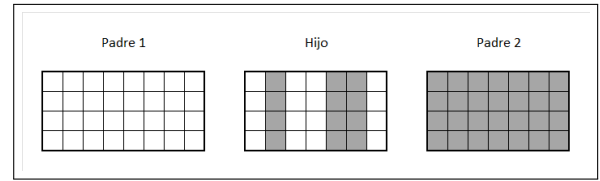


Fig. 2. Cruzamiento por columna

- Para cada fila, seleccionar con probabilidad  $p$  la fila del mismo horario del padre 1, y con probabilidad  $(1-p)$  la fila del mismo horario del padre 2.

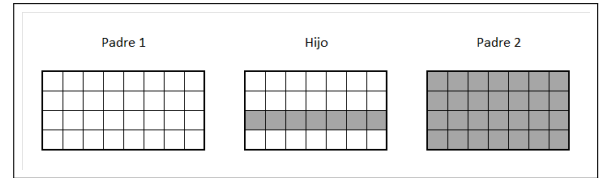


Fig. 3. Cruzamiento por fila

Como algoritmo, dado que es un problema multiobjetivo, se utilizó NSGA-II.

### F. Restricciones

Se restringe en el problema la cantidad de alimentos que componen una comida a un máximo de cuatro. Adicionalmente, se tiene únicamente en cuenta que un día se distribuye en cuatro comidas.

Si bien es posible modificar los valores de configuración del algoritmo para aceptar un rango mayor de alimentos y comidas diarias, esto implica la creación de un nuevo dataset que pueda soportar una mayor cantidad de ponderaciones horarias.

## V. PROPUESTA DE EVALUACIÓN EXPERIMENTAL

### A. Generación de instancias

Se diseñaron dos instancias para realizar calibraciones, las cuales están inspiradas en dos personas de diferentes características, y con diferentes objetivos. De esta manera buscamos una dieta diaria en torno a las siguientes características:

Característica	Instancia 1	Instancia 2
Calorías	2000.0 kc	1704.0 g
Proteínas	153.0 g	128.0 g
Carbohidratos	205.0 g	170.0 g
Grasas	68.0 g	57.0 g

TABLE II  
REQUERIMIENTOS DE CADA INSTANCIA

De forma adicional, se fijan los hiperparámetros definidos de la siguiente forma:

- $\alpha = 1.0$
- $\beta = 1.0$
- $\gamma = 0.75$
- $\delta = 100.0$
- $\sigma = 10.0$

En base a estos objetivos, se configuran 18 algoritmos, que varían en los parámetros y probabilidades comentados en secciones anteriores. Estas configuraciones son:

AE	Tipo Cruzamiento	Prob. Cruzamiento	Prob. Mutación
1	Por Columna	0.3	0.1
2	Por Columna	0.3	0.3
3	Por Columna	0.3	0.5
4	Por Columna	0.5	0.1
5	Por Columna	0.5	0.3
6	Por Columna	0.5	0.5
7	Por Columna	0.8	0.1
8	Por Columna	0.8	0.3
9	Por Columna	0.8	0.5
10	Por Fila	0.3	0.1
11	Por Fila	0.3	0.3
12	Por Fila	0.3	0.5
13	Por Fila	0.5	0.1
14	Por Fila	0.5	0.3
15	Por Fila	0.5	0.5
16	Por Fila	0.8	0.1
17	Por Fila	0.8	0.3
18	Por Fila	0.8	0.5

TABLE III  
CONFIGURACIÓN DE LOS PARÁMETROS DEL ALGORITMO

### B. Test de normalidad sobre las muestras

En todos los casos se aplicaron los tests de Anderson-Darling y Shapiro-Wilk para comprobar la normalidad de los datos obtenidos y así poder compararlos. Esto se realizó con la biblioteca de Python, SciPy [8], con los paquetes stats.anderson y stats.shapiro, los cuales proveen el test directamente y retornan si la distribución es normal o no.

Se aplicaron los tests a los resultados de las 30 ejecuciones para cada una de las 18 configuraciones definidas para ambas instancias, obteniendo los siguientes resultados:

Se observa, que mediante las pruebas utilizando Anderson-Darling se obtuvo que todas las AEs de ambas instancias presentan una distribución normal; por otra parte, mediante Shapiro-Wilk se obtuvo que para 5 AEs (en total entre ambas instancias) no se seguía la misma distribución, lo cual se considera despreciable y se asume que los valores de Fitness siguen una distribución normal. Por lo tanto se procede a comparar los algoritmos con el fin de determinar el mejor.

AE	T Cruz.	P. Cruz.	P. Mut.	Anderson Darling	Shapiro Wilk
1	Columna	0.3	0.1	True	True
2	Columna	0.3	0.3	True	True
3	Columna	0.3	0.5	True	True
4	Columna	0.5	0.1	True	False
5	Columna	0.5	0.3	True	True
6	Columna	0.5	0.5	True	True
7	Columna	0.8	0.1	True	True
8	Columna	0.8	0.3	True	True
9	Columna	0.8	0.5	True	True
10	Fila	0.3	0.1	True	True
11	Fila	0.3	0.3	True	False
12	Fila	0.3	0.5	True	True
13	Fila	0.5	0.1	True	True
14	Fila	0.5	0.3	True	True
15	Fila	0.5	0.5	True	True
16	Fila	0.8	0.1	True	True
17	Fila	0.8	0.3	True	False
18	Fila	0.8	0.5	True	True

TABLE IV  
TEST DE NORMALIDAD PARA LA INSTANCIA 1.

AE	T Cruz.	P. Cruz.	P. Mut.	Anderson-Darling	Shapiro-Wilk
1	Columna	0.3	0.1	True	True
2	Columna	0.3	0.3	True	True
3	Columna	0.3	0.5	True	True
4	Columna	0.5	0.1	True	True
5	Columna	0.5	0.3	True	True
6	Columna	0.5	0.5	True	True
7	Columna	0.8	0.1	True	True
8	Columna	0.8	0.3	True	True
9	Columna	0.8	0.5	True	True
10	Fila	0.3	0.1	True	True
11	Fila	0.3	0.3	True	True
12	Fila	0.3	0.5	True	True
13	Fila	0.5	0.1	True	False
14	Fila	0.5	0.3	True	True
15	Fila	0.5	0.5	True	False
16	Fila	0.8	0.1	True	True
17	Fila	0.8	0.3	True	True
18	Fila	0.8	0.5	True	True

TABLE V  
TEST DE NORMALIDAD PARA LA INSTANCIA 2.

### C. Elección de la mejor configuración

Una vez determinado que los valores de los mejores Fitness siguen una distribución normal, se puede realizar un test estadístico.

Para esto se utiliza un test de hipótesis de una vía (1- way) ANOVA (Analysis of Variance), para descartar o confirmar si los valores de cada algoritmo provienen de una media similar o no. Para la ejecución del test se utiliza la biblioteca SciPy [8] de Python.

Para ello se plantan las siguientes hipótesis:

- Hipótesis nula: Los valores provienen de la misma media.
- Hipótesis alternativa: Los valores no provienen de la misma media.

Se utiliza  $\alpha = 5 \%$ , buscando así tener un 95% de confianza para asegurar que la hipótesis nula no sea rechazada. Por tanto, si el  $p$  valor para el test de ANOVA es menor que el nivel de significancia (0.05) entonces rechazamos la hipótesis nula, asegurando que las medias de los grupos difieren. Se

realizan todas las combinaciones entre los distintos algoritmos obteniendo los siguientes  $p$ -valores:

AE	1	2	3	4	5
1	1	1.07E-52	2.88E-68	0.0783	1.79E-51
2	1.07E-52	1	4.65E-11	7.07E-50	0.978
3	2.88E-68	4.65E-11	1	5.75E-68	5.78E-11
4	0.0783	7.07E-50	5.75E-68	1	7.18E-49
5	1.79E-51	0.978	5.78E-11	7.18E-49	1
6	5.36E-70	5.39E-11	0.872	1.27E-69	6.75E-11
7	0.466	5.23E-55	4.06E-73	0.29	6.53E-54
8	1.18E-45	0.572	1.9E-8	9.7E-44	0.591
9	1.72E	3.9E-12	0.908	5.43E-73	5.37E-12
10	0.448	2.86E-56	1.39E-73	0.25	5.66E-55
11	5.73E-54	0.921	4.91E-11	3.89E-51	0.944
12	4.96E-69	8.61E-13	0.48	5.57E-69	1.05E-12
13	0.291	1.06E-50	6.35E-67	0.44	1.66E-49
14	1.74E-43	0.176	7.99E-14	9.53E-41	0.17
15	1.13E-69	2.8E-10	0.656	4.28E-69	3.7E-10
16	0.579	8.7E-56	3.44E-73	0.198	1.43E-54
17	1.13E-51	0.706	2.83E-12	7.42E-49	0.687
18	2.94E-68	9.11E-12	0.753	4.24E-68	1.11E-11

TABLE VI

TEST DE ANOVA Y  $p$ -VALORES OBTENIDOS (AES = [1, 5]).

AE	6	7	8	9
1	5.36E-70	0.466	1.18E-45	1.72E-73
2	5.39E-11	5.23E-55	0.572	3.9E-12
3	0.872	4.06E-73	1.9E-8	0.908
4	1.27E-69	0.29	9.7E-44	5.43E-73
5	6.75E-11	6.53E-54	0.591	5.37E-12
6	1	6.66E-75	2.3E-8	0.777
7	6.66E-75	1	3.34E-48	2.01E-78
8	2.3E-8	3.34E-48	1	3.3E-9
9	0.777	2.01E-78	3.3E-9	1
10	2.12E-75	0.977	6.91E-49	3.75E-79
11	5.77E-11	2.34E-56	0.63	4.11E-12
12	0.38	4.38E-74	6.3E-10	0.539
13	1.34E-68	0.762	5.82E-44	4.2E-72
14	7.74E-14	2.35E-45	0.0756	4.64E-15
15	0.771	2.21E-74	9.6E-8	0.563
16	5.4E-75	0.838	1.29E-48	1.23E-78
17	3.16E-12	5.45E-54	0.36	1.98E-13
18	0.631	3.27E-73	4.5E-9	0.833

TABLE VII

TEST DE ANOVA Y  $p$ -VALORES OBTENIDOS (AES = [6, 9]).

El algoritmo con mejor Fitness total es el número 16 el cual, según los test de hipótesis realizados, difiere su media a todos los algoritmos exceptuando de algunas configuraciones. De todas formas, se entiende que el algoritmo 16 fue superior en general a los demás algoritmos propuestos y proviene de una muestra con media distinta al resto. Por esto el algoritmo seleccionado es el número 16 con la configuración:

- Tipo cruzamiento: Por Fila
- Probabilidad de Cruzamiento: 0.8
- Probabilidad de Mutación: 0.1

AE	10	11	12	13	14	15
1	4.5E-01	5.7E-54	5.0E-69	2.9E-01	1.7E-43	1.1E-69
2	2.9E-56	9.2E-01	8.6E-13	1.1E-50	1.8E-01	2.8E-10
3	1.4E-73	4.9E-11	4.8E-01	6.4E-67	8.0E-14	6.6E-01
4	2.5E-01	3.9E-51	5.6E-69	4.4E-01	9.5E-41	4.3E-69
5	5.7E-55	9.4E-01	1.1E-12	1.7E-49	1.7E-01	3.7E-10
6	2.1E-75	5.8E-11	3.8E-01	1.3E-68	7.7E-14	7.7E-01
7	9.8E-01	2.3E-56	4.4E-74	7.6E-01	2.4E-45	2.2E-74
8	6.9E-49	6.3E-01	6.3E-10	5.8E-44	7.6E-02	9.6E-08
9	3.8E-79	4.1E-12	5.4E-01	4.2E-72	4.6E-15	5.6E-01
10	1	1.3E-57	1.8E-74	7.2E-01	2.5E-46	4.8E-75
11	1.3E-57	1	8.5E-13	6.0E-52	1.4E-01	3.1E-10
12	1.8E-74	8.5E-13	1	9.2E-68	1.4E-15	2.5E-01
13	7.2E-01	6.0E-52	9.2E-68	1	1.5E-41	2.9E-68
14	2.5E-46	1.4E-01	1.4E-15	1.5E-41	1	4.7E-13
15	4.8E-75	3.1E-10	2.5E-01	2.9E-68	4.7E-13	1
16	8.5E-01	3.9E-57	4.4E-74	6.0E-01	5.2E-46	1.4E-74
17	4.0E-55	6.3E-01	4.2E-14	4.2E-14	1.3E-49	1.9E-11
18	1.3E-73	9.4E-12	7.0E-01	6.0E-67	1.5E-14	4.5E-01

TABLE VIII

TEST DE ANOVA Y  $p$ -VALORES OBTENIDOS (AES = [10, 15]).

AE	16	17	18
1	5.8E-01	1.1E-51	2.9E-68
2	8.7E-56	7.1E-01	9.1E-12
3	3.4E-73	2.8E-12	7.5E-01
4	2.0E-01	7.4E-49	4.2E-68
5	1.4E-54	6.9E-01	1.1E-11
6	5.4E-75	3.2E-12	6.3E-01
7	8.4E-01	5.5E-54	3.3E-73
8	1.3E-48	3.6E-01	4.5E-09
9	1.2E-78	2.0E-13	8.3E-01
10	8.5E-01	4.0E-55	1.3E-73
11	3.9E-57	6.3E-01	9.4E-12
12	4.4E-74	4.2E-14	7.0E-01
13	6.0E-01	1.3E-49	6.0E-67
14	5.2E-46	3.1E-01	1.5E-14
15	1.4E-74	1.9E-11	4.5E-01
16	1	1.0E-54	3.1E-73
17	1.0E-54	1	5.0E-13
18	3.1E-73	5.0E-13	1

TABLE IX

TEST DE ANOVA Y  $p$ -VALORES OBTENIDOS (AES = [16, 18]).

#### D. Calidad de soluciones

Se evalúan las dietas generadas en términos de su cumplimiento con los requisitos nutricionales. Se analiza la precisión con la que las dietas satisfacen las necesidades diarias y semanales de calorías, proteínas, carbohidratos y grasas. Además, se considera la diversidad y la variedad de alimentos en las soluciones obtenidas. Las dietas que presenten una amplia gama de alimentos y una distribución equitativa de nutrientes se consideran de alta calidad.

#### E. Comparación con otros algoritmos

Se lleva a cabo una comparación entre los resultados obtenidos por el algoritmo propuesto y los generados por un algoritmo ávido (Greedy) implementado. El objetivo principal es evaluar el rendimiento y la eficacia del enfoque basado en Algoritmos Evolutivos (AE) en comparación con una estrategia ávida.

El algoritmo ávido implementado se caracteriza por tomar decisiones locales óptimas en cada etapa del proceso sin

considerar la totalidad del problema. En este caso, se evaluará la calidad nutricional de las soluciones, la diversidad en la selección de alimentos y la eficiencia computacional. Se compararán los mejores resultados de Fitness obtenidos en las diferentes instancias ejecutadas.

Como se observa en las figuras, las AEs cuya configuración presenta una muy baja probabilidad de mutación tienden a tener medias de Fitness más bajas.

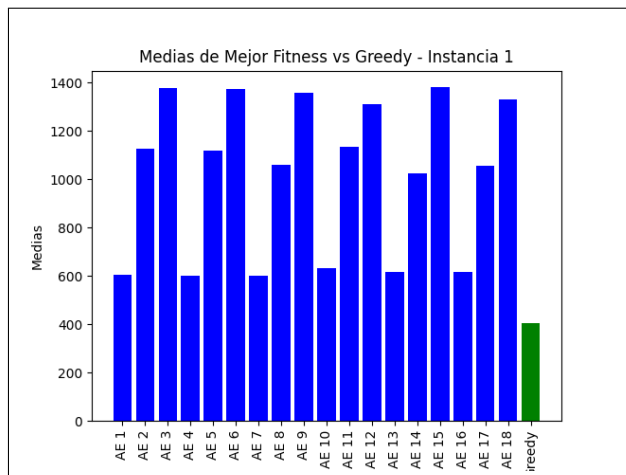


Fig. 4. Medias de mejores Fitness de la primer instancia en comparación con el algoritmo Greedy

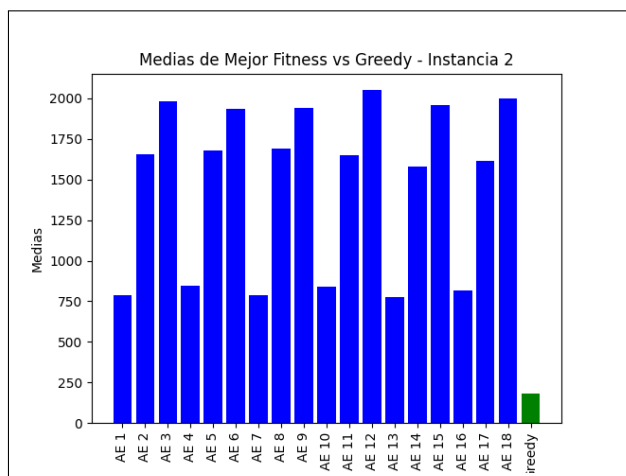


Fig. 5. Medias de mejores Fitness de la segunda instancia en comparación con el algoritmo Greedy

Esta tendencia también se ve reflejada en los mínimos de los mejores Fitness para ambas instancias.

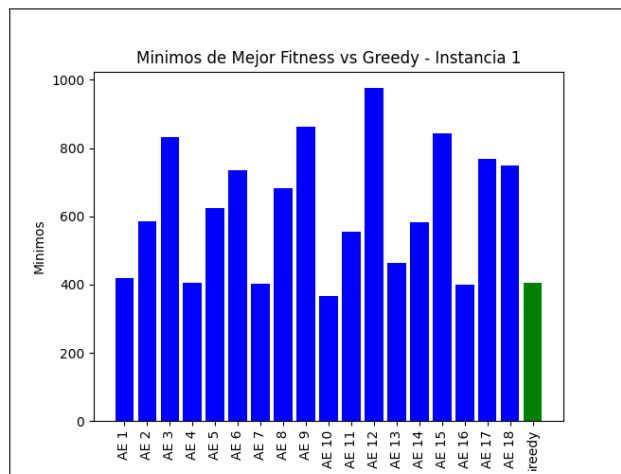


Fig. 6. Mínimos de mejores Fitness de la primer instancia en comparación con el algoritmo Greedy

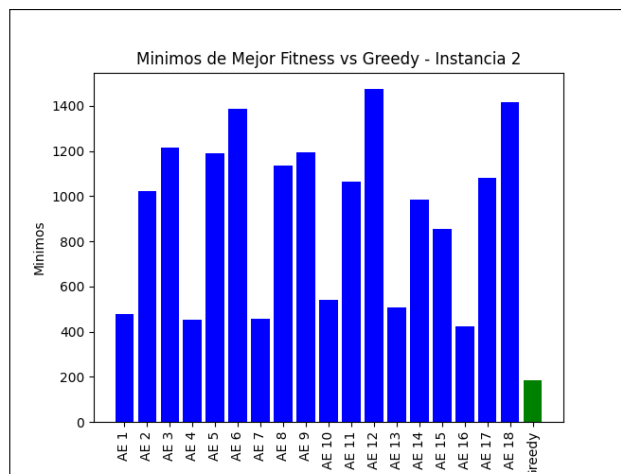


Fig. 7. Mínimos de mejores Fitness de la segunda instancia en comparación con el algoritmo Greedy

De forma adicional, se evalúa la eficiencia del algoritmo propuesto en términos de tiempo de ejecución.

Se observa que no hay tanta varianza en los tiempos de ejecución de las AEs a pesar de la configuración o instancia elegida. Sin embargo, el tiempo de ejecución del algoritmo ávido suele aproximarse a la mitad de las medias de ejecución de las AEs.

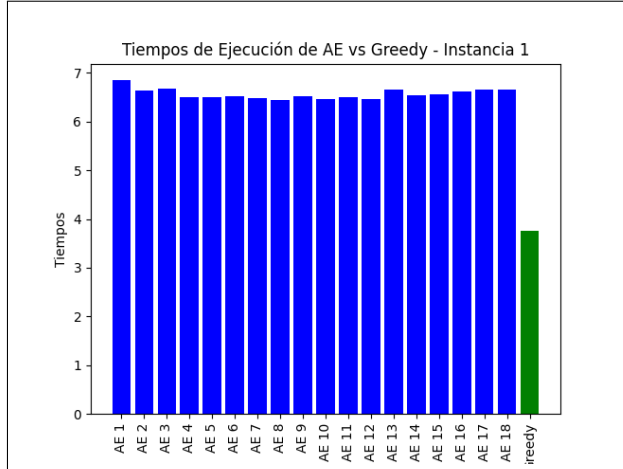


Fig. 8. Tiempos de ejecución de los AEs de la primer instancia en comparación con el algoritmo Greedy

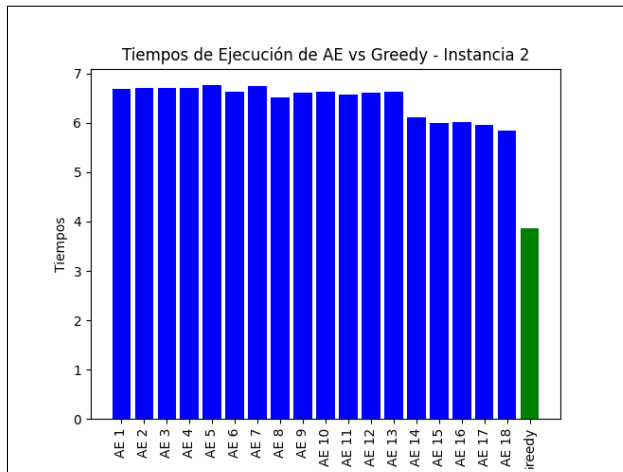


Fig. 9. Tiempos de ejecución de los AEs de la segunda instancia en comparación con el algoritmo Greedy

## VI. EVALUACIÓN EXPERIMENTAL

Para la evaluación experimental se trabajó con la configuración del algoritmo 16 (AE16) y se comparo los resultados con los de la ejecución del algoritmo Greedy implementado. Las características del PC utilizado para estas pruebas son las siguientes:

- CPU: Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz
- RAM: 16.0 GB @ 2667 MHz
- SO: Windows 11 Pro 22H2

A continuación se muestran los resultados de la ejecución del AE16 y el algoritmo Greedy donde se evalúa cada instancia en cada algoritmo.

AE	Tiempo Media(s)	STD Tiempo
16	6.62	0.36
Greedy	3.76	0

TABLE X

RESUMEN DE TIEMPO DE EJECUCIÓN - INSTANCIA 1

AE	Fit_min	Fit_mean	Fit_max
16	398.45	616.96	771.50
Greedy	405.25	405.25	405.25

TABLE XI

RESUMEN DE FITNESS - INSTANCIA 1

AE	Var_min	Var_mean	Var_max
16	0.0	0.0	120.0
Greedy	0.0	0.0	0.0

TABLE XII

RESUMEN DE VARIABILIDAD - INSTANCIA 1

AE	Tiempo Media (s)	STD Tiempo
16	6.01	0.13
Greedy	3.85	0

TABLE XIII

RESUMEN DE TIEMPO DE EJECUCIÓN - INSTANCIA 2

AE	Fit_min	Fit_mean	Fit_max
16	423.33	818.23	1305.34
Greedy	183.06	183.06	183.06

TABLE XIV

RESUMEN DE FITNESS - INSTANCIA 2

AE	Var_min	Var_mean	Var_max
16	0.0	10	140
Greedy	0.0	0.0	0.0

TABLE XV

RESUMEN DE VARIABILIDAD - INSTANCIA 2

## VII. CONCLUSIONES

El algoritmo Greedy diseñado realiza una buena aproximación a la solución deseada y a los objetivos definidos. Sin embargo, la obtención de una dieta mediante este método no parece ser una buena solución, ya que implica que cada ejecución arrojará los mismos resultados, teniendo la misma dieta todas las semanas. Por otro lado, el algoritmo evolutivo desarrollado, se ajusta muy bien a los objetivos nutricionales definidos y a la obtención de un menú equilibrado y bien distribuido. Además, cada ejecución de este proporciona como resultado en un tiempo despreciable y da como resultado una dieta diferente para cada semana aportando diversidad.

Analizando en un aspecto mas general, se observa que los mejores resultados obtenidos en los algoritmos presentados se dan cuando la probabilidad de mutación es baja, alrededor de 0.1.

## VIII. TRABAJO A FUTURO

Como trabajos a futuro se plantea:

- Explorar nuevas configuraciones de los parámetros, sobre todo en valores de probabilidad de mutación.
- 
- 

Búsqueda exhaustiva de parámetros: Uno de los puntos más importantes a trabajar es realizar un análisis sobre los parámetros de la mutación SGD. La misma trae consigo 3 nuevos parámetros que pueden ser configurables para obtener un desempeño mucho mayor al logrado. Expansión en una red mayor: Se podría realizar el mismo procedimiento pero en una red mucho más grande como puede ser una VGG16 (con 16 capas convolucionales y 135 millones de pesos).

## REFERENCES

- [1] "Material Teórico – Curso Algoritmos Evolutivos". EVA FING. <https://eva.fing.edu.uy/course/view.php?id=1049> (accessed: Oct. 21, 2023).
- [2] Babalola, A.E., Ojokoh, B.A., Odili, J.B.: Diet optimization techniques: A review, Computer Engineering and Computer Science (ICMCECS), pp. 1-5, IEEE (2020).
- [3] "jMetalPy". GitHub. <https://github.com/jMetal/jMetalPy> (accessed: Oct. 21, 2023).
- [4] "Calorías de los Alimentos - Información Nutricional". Medicina Información. <https://medicinainformacion.com/wp-content/uploads/2018/04/tabla-de-alimentos.pdf> (accessed: Oct. 21, 2023).
- [5] "Food and Nutrient Database for Dietary Studies". Agricultural Research Service. U.S. Department of Agriculture. <https://www.ars.usda.gov/northeast-area/beltsville-md-bhnrc/beltsville-human-nutrition-research-center/food-surveys-research-group/docs/fndds-download-databases/> (accessed: Dec. 12, 2023).
- [6] "Calculadora de Calorías y Macros para Calcular tu Metabolismo" TRX España. <https://trxspain.es/calculadora-de-calorias-y-macros/> (accessed: Dec. 12, 2023).
- [7] "Recomendaciones de Ingesta Energía y Nutrientes para la población uruguaya". Ministerio de Salud Pública. [https://www.gub.uy/ministerio-salud-publica/sites/ministerio-salud-publica/files/documentos/publicaciones/RECOMENDACIONES\\_INGESTAS\\_ENERGIA\\_NUTRIENTES\\_POBLACION\\_URUGUAYA.pdf](https://www.gub.uy/ministerio-salud-publica/sites/ministerio-salud-publica/files/documentos/publicaciones/RECOMENDACIONES_INGESTAS_ENERGIA_NUTRIENTES_POBLACION_URUGUAYA.pdf) (accessed: Dec. 12, 2023).
- [8] "Scipy." <https://www.scipy.org/> (accessed: Dec. 11, 2023)