

***Pladiet*: Un sistema de cómputo para el diseño de dietas individualizadas utilizando Algoritmos Genéticos.**

Pedro FLORES, Ma. de Gpe. COTA, Desiderio RAMÍREZ, Ivo José JIMÉNEZ, Juan Antonio RAYGOZA, Lluvia Carolina MORALES.
Departamento de Matemáticas, Universidad de Sonora,
Hermosillo, Sonora. CP 83000 México.

Samuel GALAVIZ, Armida ESPINOZA María Esther OROZCO
Departamento de Ciencias Químico Biológicas, Universidad de Sonora,
Hermosillo, Sonora. CP 83000 México.

RESUMEN

En este trabajo se presentan las bases de funcionamiento del sistema de cómputo *Pladiet* que calcula dietas individualizadas que cumplen con todos los requerimientos nutricionales establecidos por un experto. Para resolver el problema de calcular las dietas, se construye un modelo de Programación Entera no lineal que se resuelve utilizando Algoritmos Genéticos. Aquí se presenta la construcción del modelo, las características específicas del Algoritmo Genético con que se resuelve, y los resultados de las pruebas para ajuste de parámetros correspondientes. *Pladiet* es el resultado de una investigación conjunta entre los Departamentos de Nutrición y Ciencias de la Computación de la Universidad de Sonora en México y actualmente esta en uso en el Modulo de Orientación Nutricional (MON) de dicha Universidad, en donde se atienden anualmente a aproximadamente 400 pacientes con 2000 citas programadas.

Palabras Claves: Algoritmos, Genéticos, Programación, Dietas, Menús, Alimentos.

1. INTRODUCCIÓN

Cada día son más las personas que padecen enfermedades nutricionales debido a la falta de cultura sobre una sana alimentación. Como una muestra de esto se tiene el hecho de que la obesidad y el sobrepeso se están convirtiendo en características de los países desarrollados. En países como Inglaterra y Estados Unidos, más de la mitad de la población adulta padece de obesidad, lo que origina severos problemas de salud como: aumento de colesterol y triglicéridos, hipertensión arterial, problemas cardiovasculares, diabetes y algunos tipos de cáncer. Por otra parte, la alimentación con comida chatarra puede producir anemia con lo cual se generan problemas de: cansancio, sueño constante, dolores de cabeza, etc. que propician bajo rendimiento laboral y escolar [1].

Es por ello que en los últimos años, tanto instituciones públicas como privadas, se han dado a la tarea de diseñar programas de información y educación, y crear clínicas o módulos de atención nutricional para atacar este problema, donde se revisa el estado de salud de los pacientes y se establece una dieta que deberá de llevar a cabo. Posteriormente, el paciente deberá regresar para

evaluar los avances en su estado de salud y recibir una nueva dieta acorde al mismo [3].

La elaboración de la dieta, de acuerdo al estado de salud y costumbres alimenticias del paciente, es uno de los elementos principales que permiten resolver exitosamente los problemas nutricionales de una población. Sin embargo, elaborar una dieta individualizada no es una tarea fácil, pues el experto en nutrición puede tardar varias horas en diseñar un menú (semanal o quincenal) para dicha dieta, debido a que tiene que tomar en cuenta diversas variables para calcularla, tales como: el peso del paciente, edad, índice de masa corporal (IMC), actividad física, hábitos alimenticios, gustos, etc. Todo esto solo para calcular el número de calorías que debe de consumir al día, y a partir de ello hacer menús para cada una de las comidas del día (desayuno, comida, cena) que estén balanceados y cumplan con el requisito de que en conjunto se acerquen lo más posible al número de calorías diarias requeridas [6].

Por tal razón, la mayoría de los especialistas se deciden por la opción de tener sólo dos o tres diseños de menús de dietas diferentes, dependiendo del tipo de paciente, aunque esto no sea lo que mejores resultados pudiera darle a cada persona en particular.

El problema de calcular una dieta individualizada que cumpla con todos los requerimientos nutricionales establecidos por un experto y minimizando, por ejemplo, el costo de la dieta, se modela como un problema de Programación Lineal Entera que, por ser NP-Completo [2], no se puede resolver exactamente en un tiempo razonable. En este trabajo se resuelve el problema de calcular dietas para un periodo de tiempo dado, con un cierto número de calorías diarias en promedio y donde los alimentos sean lo mas diversos posibles. El modelar esta situación nos lleva a plantear un problema de programación entera no-lineal.

El sistema de cómputo *Pladiet* satisface la necesidad, de las personas dedicadas a la asesoría nutricional, de tener una herramienta que facilite y agilice la atención personalizada de numerosos pacientes con necesidad de mejorar su calidad alimenticia. *Pladiet* puede ser usado en hospitales, escuelas, comedores, consultorios médicos, etc. En general, gracias a su diseño y documentación, incluso puede ser utilizado en hogares y por cualquier persona que tenga nociones básicas de salud nutricional.

En la siguiente sección se describe nuestro problema y en la sección tres se presenta el modelo de programación entera no-lineal que construimos para resolverlo. En la sección cuatro se

resumen los principales aspectos de los Algoritmos Genéticos (AG) y la implementación que hicimos para nuestros problemas. Posteriormente en esta sección se presentan los resultados de ajustes de parámetros y por último daremos las conclusiones de este trabajo.

2. DESCRIPCIÓN DEL PROBLEMA

El procedimiento que se sigue para prestar la asesoría nutricional consta de las siguientes etapas:

En la primera etapa se evalúa al paciente con el fin de conocer sus hábitos alimenticios así como su actividad física, horas de sueño e historial médico.

En la segunda etapa se valora el estado físico y se obtienen varios indicadores importantes como: índice de masa corporal, índice cintura-cadera, etc.

Posteriormente, se realiza una encuesta alimenticia tomando en cuenta lo que el paciente desea y no desea comer. Esta es la parte de la consulta que más tiempo consume.

Por último, se le calcula un programa alimenticio donde cada mes se establece una dieta que permita llegar a los resultados esperados.

Para esto último hay que tomar en cuenta que, mientras más diversidad tenga la dieta, y se parezca más a las comidas a las que está acostumbrado el paciente, mayor será la posibilidad de éxito. Sin embargo, el esfuerzo de calcular una dieta individual para cada paciente y hacerlo manualmente (como tradicionalmente se hace) es muy grande. A continuación veremos la construcción del modelo que se utilizó para este problema.

3. CONSTRUCCION DEL MODELO

Se cuenta con un diccionario de alimentos como el de la Figura 1. Cada registro corresponde al nombre del alimento, su clave y la cantidad de alimento equivalente a una porción. Los campos siguientes indican una serie de contenidos nutricionales que cada porción de alimento aporta. En este sistema se tiene la información de los nutrientes: energía, fibra, calcio, proteína, grasas (varios tipos), colesterol, hierro, vitaminas: A, C, E, Folato, Sodio y Zinc.

Alimento	Clave	Grs.	E	Fibra
Cerveza Regular	1	360.00	150.00	1.80
Licor Gin., Rum., Vodka., Whiskey	4	42.00	105.00	0.00
Vino Blanco	8	102.00	80.00	0.00
Agua Mineral	9	355.00	0.00	0.00
Soda de cola (coca o pepsi) regular	10	369.00	160.00	0.00
Soda de dieta	11	355.00	0.00	0.00
Soda Lima-Limón	14	372.00	155.00	0.00
Soda de Naranja	15	372.00	180.00	0.00

Figura 1

Le llamaremos menú a una agrupación de alimentos y los menús se clasifican en: desayunos, comida, cenas e intermedios. Los intermedios son menús que se consumen entre el desayuno y la comida y entre la comida y la cena. Una dieta para un día

esta formada por un desayuno, una comida, una cena y dos intermedios. A continuación en la figura 2 se muestra una dieta para una persona que requiere ingerir 1500 calorías diarias y que solo toma en cuenta desayunos, comidas y cenas. Por cada día aparecen menús formados por diversos alimentos agrupados en desayunos, comidas y cenas. El sistema actualmente trabaja con más de 3000 menús.

LUNES	MARTES	MIÉRCOLES
1 taza de yogurth natural 1 taza de fruta picada 1 cucharada de granola 1 reb de pan integral	1 rebanada de jamon pavo 1 reb de pan integral 1/2 platano 1 taza de leche descremada	1 taza de cereal bran flakes 1 taza de leche descremada 1 fruta
1/2 pechuga de pollo con vegetales 1 taza de arroz integral 1 vaso de agua de fruta natural ensalada verde	1 plato de crema de calabaza 1/2 pechuga de pollo a la parrilla 1 taza de arroz integral Ensalada de hojas verdes 1 vaso de agua de frutas natural	3 brochetas de pollo/verduras parrilla 1/2 taza arroz integral 1 vaso de agua de jamaica
1/2 Taza de yogurth 1 mollete 1/2 bolillo s/migajón 1 rebanada de queso cocido 1 cucharada de frijol salsa bandera	1 taza de leche descremada 1 taza de cereal integral 1 fruta mediana	1 licuado de yogurt con fruta licuado con hielo 1 pan integral con requeson

Figura 2

Antes de calcular la dieta se le pregunta al paciente que alimentos no le gusta comer y los menús que contienen los alimentos no deseados, son excluidos de tal forma que no se toman en cuenta a la hora de calcular la dieta.

El modelo general que se construye para este problema se basa en los siguientes supuestos:

- 1) Se tienen una serie de menús que pertenece a varios grupos, por ejemplo desayunos, comidas, cenas, etc. Cabe agregar que el modelo permite considerar que un grupo está constituido por varios subgrupos. Por ejemplo una comida pudiera ser formada por: un guisado, una sopa y una ensalada
- 2) En cada día se elige siempre el mismo número de menús de cada grupo, por ejemplo un desayuno, una comida, una cena y dos intermedios.
- 3) Para este trabajo solo se solicita que la dieta proporcione un cierto número de calorías diarias en promedio. Este modelo se puede extender a incluir otros nutrientes.
- 4) El objetivo es construir una programación de menús que sea lo más diversa posible. Esto con el fin de evitar la rutina en la alimentación.

A continuación definiremos las siguientes variables:

n	= número de menús
T	= período de días
c	= calorías diarias promedio
x_i	= número de veces que se programa el menú i en los T días.
a_i	= número de calorías del menú i .
k	= número de grupos en la dieta.
G_j	= conjunto de índices de menús que están en el grupo de alimentos j .
Z	= medida de la diversidad de la dieta.

Con esta notación si

$$G_1 = \{1, \dots, r\}$$

$$G_2 = \{r+1, \dots, s\}$$

:
:
 $G_k = \{t+1, \dots, n\}$

y

$$f(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$$

nuestro problema se plantearía como:

$$\max Z = \sum_{i=1}^n f(x_i) \quad (1)$$

sujeto a

$$a_1 x_1 + \dots + a_n x_n = c * T$$

$$x_1 + \dots + x_r = T$$

$$x_{r+1} + \dots + x_s = T$$

:
:

$$x_{t+1} + \dots + x_n = T$$

donde la función a maximizar representa que la dieta sea lo más diverso posible. La primera restricción significa que el promedio diario de calorías sea c y las restricciones siguientes representan que todos los días se programa un solo alimento de cada grupo. Si se desea programar más alimentos de un grupo cada día, la modificación al modelo es trivial.

4. ALGORITMOS GENETICOS

Los AG se utilizan principalmente en la solución de problemas de búsqueda y optimización y están basados en los métodos de evolución biológica: selección natural, reproducción y mutación. Fueron descubiertos y formalizados teóricamente por John Holland en 1975 y desde sus inicios han sido usados en problemas reales tales como: optimización y control de tuberías de gas, diseño de redes de comunicación, diseño de armaduras metálicas, etc. [4]

Para resolver un problema particular, utilizando AG, es necesario, como se señala en [5], contar con:

- 1) Una representación genética del problema (el individuo). Esta representación deberá ser codificada con cadenas de longitud finita sobre un conjunto finito de símbolo.

- 2) Alguna manera de crear la población inicial, la cual puede ser aleatoria o seleccionada por otros métodos
- 3) Una función de desempeño que revise que tan adaptado se encuentra un individuo al medio y los separe, por un procedimiento llamado selección, según su desempeño.
- 4) Operadores genéticos que alteran la estructura de los hijos; normalmente se consideran el cruzamiento y la mutación.

Los valores de varios parámetros que son requeridos por los AG, como son: el tamaño de la población (*pop_size*), probabilidad de cruzamiento (*pc*), y probabilidad de mutación (*pm*).

5. SOLUCIÓN DEL PROBLEMA CON AG

Para programar una dieta se cuenta con una base de datos donde los alimentos están numerados y asignados a algún grupo (desayunos, comidas, cenas e intermedios).

Representación de los individuos.

A continuación supondremos que queremos establecer una dieta para T días, y que en cada día se eligen k menús de los grupos establecidos.

Para resolver el problema, se necesita tener la representación de una posible solución, la cual es llamada individuo. En este problema, el individuo es representado por un vector de números enteros de tamaño $k*T$. Estos vectores podemos considerar que están formados por secciones, y cada sección de k componentes representa los menús que se programan en un día y aparecen T de estas secciones, una por día. Por ejemplo si tenemos 3 grupos de alimentos (digamos: desayunos, comidas y cenas) y queremos programar una dieta para 5 días se tendría que un individuo es un vector de la forma:

$$(3,5,7|8,2,13|9,5,6|21,12,23|7,10,11)$$

que significa que en estos cinco días se programarían los desayunos: 3,8,9,21 y 7; las comidas:5,2,5,12,10 y las cenas 7,13,6,23 y 11. Las secciones aparecen separadas por el símbolo |, y cada sección representa lo que se consume en un día.

Generando la Población Inicial

Para generar la población inicial (o matriz de padres), se generan los renglones (individuos) de acuerdo al desarrollo que se presenta a continuación.

Al número de renglones de la matriz de padres le llamaremos *pop-size*.

Para generar un individuo (o renglón) se va eligiendo al azar un número de menú de cada grupo de alimentos hasta que se termina la programación de un día y se continua así hasta que se termina la programación de los T días en cuestión.

Este procedimiento se continúa hasta que se genera los *pop-size* renglones de la matriz de padres

Función de Desempeño y Mecanismo de Selección

La selección es el procedimiento a partir del cual se construye la matriz de hijos tomando como origen la matriz de padres y representando el fenómeno que ocurre en la naturaleza de la supervivencia del más apto. La forma de medir la aptitud es por medio de una función de desempeño.

En nuestro problema la función de desempeño se construye por etapas tomando primero

$$E = | a_1x_1 + \dots + a_nx_n - c \cdot T |$$

Donde E es una medida del error de no cumplir con las calorías requeridas y si consideramos la función Z definida en (1).

Se tiene que Z es la medida de la diversidad de la dieta y si consideramos

$$k \cdot T - Z$$

este término es una medida del error en la diversidad de la dieta. Con esto la función de desempeño D a minimizar sería

$$D = E + \lambda \cdot (k \cdot T - Z)$$

Donde λ es un parámetro a ajustar del modelo. Para resolver este problema se utiliza una versión de AG, la cual se encarga de minimizar esta función de desempeño.

El método de selección que aquí se usa es el de torneo binario, que es un método que consiste en seleccionar aleatoriamente dos individuos y dejar que compitan, es decir los valores de su función de desempeño son comparados, y el individuo con el mejor valor pasa a la siguiente generación. Esto se repite hasta que la matriz con los hijos este llena.

Operadores Genéticos.

Los operadores genéticos que utilizamos son el cruzamiento y la mutación.

En cuanto al cruzamiento, se utilizó el cruzamiento simple. Para este cruzamiento se van tomando parejas de individuos y, con probabilidad pc , se decide si se cruzan ó no. Si se decide no cruzar dos individuos, estos pasan directamente a formar parte del conjunto de hijos. Si se decide cruzar se escoge una componente aleatoriamente del vector de individuos y se lleva a acabo el intercambio de componentes de los vectores desde esa componente en adelante.

La mutación consiste en elegir una componente al azar de un individuo, ver a que numero de grupo corresponde (Es decir, si es desayuno, comida, cena, etc.) y se cambia por otro menú del mismo grupo. La probabilidad de mutar una componente de un individuo es pm . Para mayores detalles consultar [8].

Parámetros utilizados por los Algoritmos genéticos.

Los parámetros son los siguientes: tamaño de población, iteraciones, probabilidad de cruzamiento y probabilidad de mutación. Además, aparece un parámetro λ que permite la diversidad de la dieta. Estos parámetros deben ser encontrados para cada problema en particular. Por esta razón se establece un periodo de tiempo de 28 días para el calculo de dietas. Esto explica por que son probados diferentes tamaños de población, probabilidades de cruzamiento y mutación. Además una vez encontrados los parámetros para un cierto periodo de tiempo

nada asegura que se sean buenos parámetros para otro periodo de tiempo distinto.

Ajustes de parámetros

Para el ajuste de parámetros, cada prueba se repitió 50 veces y se tomo el promedio de los 50 resultados obtenidos.

El ajuste de parámetros se hizo en tres etapas. En la primera etapa se encontraron los parámetros adecuados para obtener dietas de 1500 calorías en promedio durante 28 días sin considerar diversidad.

En una segunda etapa, con los parámetros ya obtenidos se calculo el valor de λ para obtener la mayor diversidad posible y por ultimo se reviso que tan robusto era el algoritmo con los parámetros aquí encontrados para dietas con distintos requerimientos en calorías, pero siempre durante 28 días.

Durante la primera etapa se buscaron los mejores parámetros para este algoritmo. Los parámetros que se probaron fueron los siguientes: tamaño de población (pop_size), para lo cual se hicieron pruebas con valores de 50, 80, 100, 150, 200, 250 (manteniendo $pc=0.50$ y $pm=0.10$ fijos) y se realizaron pruebas con iteraciones entre 1000 y 2000 con incrementos de 100 en 100 y los mejores resultados que se obtuvieron fueron con un $pop_size=150$ e iteraciones=2000. En la tabla 1 se muestran los resultados correspondientes a 2000 iteraciones.

Población	Error en Calorías
50	0.24422
80	0.30070
100	0.25430
150	0.22281
200	0.25563
250	0.26531

Tabla 1

Para ajustar la probabilidad de cruzamiento (pc) esta fue variada entre 0.30 y 0.90, con incrementos de 0.05 (manteniendo fijos $pop_size=150$ e iteraciones=2000), obteniendo mejores resultados con $pc=0.50$, como se observa en la tabla 2.

Probabilidad de Cruzamiento	Error en Calorías
0.30	0.26266
0.35	0.26711
0.4	0.28062
0.45	0.30812
0.50	0.21656
0.55	0.34430
0.6	0.31836
0.65	0.23016
0.7	0.25211
0.75	0.28977
0.8	0.24641
0.85	0.29555
0.9	0.29750

Tabla 2

Posteriormente, para ajustar la probabilidad de mutación (pm) esta fue originalmente fijada a 0.05 y se hicieron pruebas con incrementos de 0.05 hasta llegar al valor de 0.50 (manteniendo fijos $pop_size=150$, $pc=0.50$ e iteraciones=2000), el valor que produjo mejores resultados fue $pm=0.35$, como se observa en la tabla 3.

Probabilidad de Mutación	Error en Calorías
0.01	0.21477
0.05	0.29672
0.10	0.27781
0.15	0.24586
0.2	0.26789
0.25	0.26547
0.3	0.26687
0.35	0.20781
0.4	0.27078
0.45	0.30047
0.5	0.23625

Tabla 3

En resumen, los parámetros con los que se obtuvieron los mejores resultados son: $pc = 0.50$, $pm = 0.35$, $pop_size = 150$ e $iteraciones=2000$.

En la segunda etapa se ajusto el parámetro λ buscando un valor que diera la mayor diversidad posible y manteniendo el error en calorías E con valor 0. Para esto el valor de λ se varió desde 100 hasta 300 con incrementos de 10 en 10. El mejor valor encontrado para este parámetro fue de $\lambda = 290$ (Ver tabla 4).

Variabilidad en λ	Error en Calorías	Error en Variabilidad $\lambda*(k*T - Z)$	Error Total
100	0.75891	312.00	312.76
110	4.02117	275.00	279.02
120	1.05594	247.20	248.26
130	4.26203	197.60	201.86
140	3.36328	148.40	151.76
150	2.17086	33.00	35.17
160	0.50461	9.60	10.10
170	2.53055	13.60	16.13
180	0.64883	0.00	0.65
190	0.45406	0.00	0.45
200	0.43758	0.00	0.44
210	0.45375	0.00	0.45
220	0.48297	0.00	0.48
230	0.41797	0.00	0.42
240	0.43734	0.00	0.44
250	0.39055	0.00	0.39
260	0.38719	0.00	0.39
270	0.37695	0.00	0.38
280	0.37156	0.00	0.37
290	0.35008	0.00	0.35
300	0.40766	0.00	0.41

Tabla 4

Por ultimo se hicieron pruebas para ver que tan robusto era el algoritmo con los parámetros aquí señalados cuando se pedían dietas de distinto numero de calorías. Las pruebas realizadas nos permiten concluir que con el conjunto de parámetros aquí encontrados se obtenían buenas dietas para cualquier requerimiento de calorías. Una muestra de los resultados obtenidos se presenta en la tabla 5.

Calorías en la Dieta	Error en Calorías	Error en Variabilidad $+\lambda*(k*T - Z)$	Error Total
1500	0.35008	0.00	0.35
2000	0.74445	0.00	0.74
2500	0.33437	0.00	0.33

Tabla 5

6. CONCLUSIONES

La principal conclusión de este trabajo es que el algoritmo aquí propuesto resuelve, en forma robusta, el problema de diseñar dietas para 28 días que cumplan con diversos requerimientos respecto a calorías.

Además, las dietas se calculan en segundos por lo cual se tiene mayor capacidad de atender a los pacientes y según la opinión de los mismos, son mas satisfactorias que las elaboradas anteriormente en forma manual.

Por ultimo cabe señalar que *Pladiet* abre la posibilidad, a futuro, de diseñar dietas sin necesidad de estar en presencia de especialistas, incluso por medio de Internet y tomando en cuenta otros nutrientes adicionales.

7. AGRADECIMIENTOS

Se agradece al profesor Francisco Salazar por sus valiosas observaciones a este artículo.

8. REFERENCIAS

- [1] Esquivel R.I . **Nutrición y Salud** .Martínez Correa, S.M., Martínez Correa, J.L. Editores. Manual Moderno. México. D.F. 1998.
- [2] Garey M. & Johnson D.: **Computer and Intractability. A Guide to the theory of NP-Completeness**. W.H. Freeman Company. 1979.
- [3] Girolami de H. **Fundamentos de Valoración Nutricional y Composición Corporal**. Editorial Ateneo. 2003
- [4] Goldberg D.E.: **Genetic Algorithms in Search, Optimization and Machine Learning**. Addison-Wesley Publishing Company, Inc. 1989.
- [5] Michalewicz, Z.: **Genetic Algorithms + Data Structures = Evolution Programs**. New York, USA: Springer Verlag. Second, extended edition. 1994.
- [6] Pérez A.B. **Nutriólogía Médica**. Editorial Médica Panamericana. Fundación Mexicana para la Salud. México. D.F. 2001.