

Diet Generator Using Genetic Algorithms

Catalán-Salgado Edgar-Armando, Zagal-Flores Roberto, Torres-Fernandez Yuliana,, and Paz-Nieves Alexis

ESCOM-IPN, Av. Juan de Dios Batiz s/n, GAM, Mexico DF

ecatalans@ipn.mx
zagalmmx@gmail.com
yuliana247@hotmail.com
alexis_suzumiya@hotmail.com

Abstract. Overweight and obesity are defined as abnormal or excessive fat accumulation that may be harmful to health and even can even result in death. This problem can be reduced for a person if this follows a proper diet, in which the consumption of kilocalorie, carbohydrates, lipids and proteins per day are restricted. Given a database of foods to find the 5 meals in each day of a week becomes a complex task. In this article a way to generate diets using genetic algorithms is presented, which one considers the restrictions mentioned and also allows establish preferences for certain food groups.

1 Introduction

Overweight and obesity are a chronic disease that can be treatable, that disease surge due to a imbalance energetic between consumed kilocalorie and expended kilocalorie[1].

This overweight generates a diversity of health problems, included but not limited to diabetes, respiratory problems, musculoskeletal problems,cardiovascular diseases, hypertension,thyroid, hypothyroidism, hyperthyroidism, hyperglycemia, cancer and psychological problems.

In Mexico, 72% of women and 66% of men suffers overweight or obesity, thus approximately 16.96 million people. In 2000, using a conservatory stage, the morbid rate for obesity related problems in population over 35 years was 195.71 for each 100000 with a total of 55749 deceases. For the same 2000, using a extended stage, the morbid rate for obesity related problems in population over 35 years was 433.6 for each 100000 with a total of 123511 deceases.

Overweight can be generated in different ways: consumption of more calories than necessary, alimentation deficit, lack of exercise, genetic inheritance, cultural factors, hormonal problems, diseases and medicine secondary effects.

A diet is a way to reduces this problem in a persons, depend on different factors and must be set by a specialist.

One diet is established limiting the quantity of kilocalorie consumed in a day, for example, 1300 kcal, 1500 kcal, etc.

The World Health Organization establish that is best to distribute the kilocalorie consumed in five time aliments per day as is specified in table 1.

Table 1. Percentage of dairy KCal consumption and their corresponding KCal in different diets.

Name	Percentage	1300 KCal	1500 KCal	1800 KCal	2000 KCal
Breakfast	20%	260 KCal	300 KCal	360 KCal	400 KCal
Lunch	10%	130 KCal	150 KCal	180 KCal	220 KCal
Meal	40%	520 KCal	600 KCal	720 KCal	800 KCal
Night tea	10%	130 KCal	150 KCal	180 KCal	220 KCal
Dinner	20%	260 KCal	300 KCal	360 KCal	400 KCal

Furthermore a balanced diet feature is macronutrient intake balance, this macronutrient are carbohydrate, lipids and proteins and their suggested distribution is shown in table 2

Table 2. Corresponding KCal per nutrient in different diets

Name	Percentage	1300 KCal	1500 KCal	1800 KCal	2000 KCal
Carbohydrate	60 %	780 KCal	900 KCal	1080 KCal	1200 KCal
Lipids	25 %	325 KCal	375 KCal	450 KCal	500 KCal
Proteins	15 %	195 KCal	225 KCal	270 KCal	300 KCal

Once we know the nutritional plan contribution of each nutrient, is necessary to know how much grams must be consumed in order to get a balanced diet, with this purpose the following equivalences are considered:

1. One Carbohydrate gram deliver 4 kcal
2. One Lipid gram deliver 9 kcal
3. One Protein gram deliver 4 kcal

Realizing the conversions our table, the distribution in grams terms is shown in table 3:

The problem resides in that given a database of N aliments how to find the appropriate combination to accomplish this distribution for a special needs of one individual. Furthermore how to find an appropriate combination for each aliment time of each day. Also a specific diet with specific aliment should bore the user fast.

The basis of this problem is to find a set of appropriate combination of aliments, the aliment database should contain one hundred or one million of

Table 3. Corresponding grams per nutrient in each diet type

Name	Percentage	1300 KCal	1500 KCal	1800 KCal	2000 KCal
Carbohydrate	60 %	195 g	225 g	270 g	300 g
Lipids	25 %	36.11 g	42.66 g	50 g	55.55 g
Proteins	15 %	48.75g	56.25 g	67.5 g	75 g

aliments, but is necessary to consider most of them when the diet is generated to add versatility.

In this paper we show a way to generate the diet using genetic algorithms, not for the complexity nor the existence of high search space, simple for the ability of find different possible solution in any part of the search space.

Anyway the diet generated must be approved by one specialist, this algorithm does not replace them, is only a tool to help them.

2 Materials and Methods

A genetic algorithm emulates the biological evolutionary process in intelligent search, operates through a cycle of the following stages[2, 3].

1. Creation of a population of strings, representing possible solutions.
2. Evaluation of each string
3. Selection of best strings
4. Genetic manipulation to create a new population of strings

We consider as an aliment as any food stored in the aliment database, for which one we have the measure of kilocalorie, carbohydrate, protein and lipid for each 100gr.

An aliment time is one of the five possible defined by the World Health Organization, thus is: Breakfast, lunch, meal, night tea and dinner

Given a aliment database, the challenge is to find a the set of aliments in each aliment time, we solve this using a genetic algorithm as specified below.

2.1 Chromosome Representation

Being a an aliment time, a chromosome \mathbf{C} of n aliments is represented in the following way:

$$\mathbf{C} = [a_1, \dots, a_n]$$

2.2 Fitness Evaluation

Chromosome evaluation is made calculating the total of kilocalorie and comparing with the expected value according the diet. In fact, this cant be complete exact, is almost impossible to achieve by a collection of aliments an exactly sum

of 1300 kcal for example. So we allow to establish a percent p , expressed in the range $[0,1]$, of accepted variation over the reference values for diet: reference kilocalorie value denoted by r_1 , carbohydrate reference value denoted by r_2 , protein reference value denoted by r_3 , and lipid reference value denoted by r_4

First calculate for \mathbf{C} the total Kilocalorie k using (1), total carbohydrate h using (2), total protein p (3), total lipid d (4)

$$k = \sum_{i=1}^n k(a_i) \quad (1) \quad h = \sum_{i=1}^n h(a_i) \quad (2) \quad t = \sum_{i=1}^n t(a_i) \quad (3) \quad d = \sum_{i=1}^n d(a_i) \quad (4)$$

Also a proximity bonus is added to the evaluation, so if its closer to the reference value a bonus is added.

First we evaluate the total Kilocalories obtained using (1) in (5) in order to get the evaluation with respect to kilocalories.

$$f_1 = \begin{cases} 2 & \text{if } r_1 * (1 - p/2) \leq k \leq r_1 * (1 + p/2) \\ 0 & \text{if } r_1 * (1 - p) \leq k < r_1 * (1 - p/2) \\ & \text{or } r_1 * (1 + p/2) < k \leq r_1 * (1 + p) \\ -2 & \text{if } k < r_1 * (1 - p) \text{ or } k > r_1 * (1 + p) \end{cases} \quad (5)$$

Later we evaluate the total carbohydrate obtained using (2) in (6) in order to get the evaluation with respect to carbohydrates.

$$f_2 = \begin{cases} 2 & \text{if } r_2 * (1 - p/2) \leq h \leq r_2 * (1 + p/2) \\ 0 & \text{if } r_2 * (1 - p) \leq h < r_2 * (1 - p/2) \\ & \text{or } r_2 * (1 + p/2) < h \leq r_2 * (1 + p) \\ -2 & \text{if } h < r_2 * (1 - p) \text{ or } h > r_2 * (1 + p) \end{cases} \quad (6)$$

Similarly we evaluate the total protein obtained using (3) in (7) in order to get the evaluation with respect to proteins.

$$f_3 = \begin{cases} 2 & \text{if } r_3 * (1 - p/2) \leq t \leq r_3 * (1 + p/2) \\ 0 & \text{if } r_3 * (1 - p) \leq t < r_3 * (1 - p/2) \\ & \text{or } r_3 * (1 + p/2) < t \leq r_3 * (1 + p) \\ -2 & \text{if } t < r_3 * (1 - p) \text{ or } t > r_3 * (1 + p) \end{cases} \quad (7)$$

Finally we evaluate the total Lipids obtained using (4) in (8) in order to get the evaluation with respect to lipids.

$$f_4 = \begin{cases} 2 & \text{if } r_4 * (1 - p/2) \leq d \leq r_4 * (1 + p/2) \\ 0 & \text{if } r_4 * (1 - p) \leq d < r_4 * (1 - p/2) \\ & \text{or } r_4 * (1 + p/2) < d \leq r_4 * (1 + p) \\ -2 & \text{if } d < r_4 * (1 - p) \text{ or } d > r_4 * (1 + p) \end{cases} \quad (8)$$

To get the final fitness value we add four previous values using (9)

$$f = f_1 + f_2 + f_3 + f_4 \quad (9)$$

2.3 Selection

According with the evaluation we order the population from best fitness to lower fitness, we select half of the population that corresponds to the best evaluated chromosomes reproduces it using the crossover function.

2.4 Crossover

being C^1 and C^2 two chromosomes of the same dimension as follows

$$C^1 = [a_1, \dots, a_n]$$

$$C^2 = [b_1, \dots, b_n]$$

A single point crossover between this two chromosomes is used in order to generate another two chromosomes, being p the cross point position we have

$$C = [a_1, \dots, a_p, \dots, b_{p+1}, b_n]$$

$$C = [b_1, \dots, b_p, \dots, a_{p+1}, a_n]$$

2.5 Mutation

Mutation only select one descendent and one position randomly and change its value.

3 Results and Discussion

We prove our algorithm with different parameters to see the convergence time, the graphics result is shown.

Test 1

Create a breakfast, with the following parameters:

Population size:15

Generations: 100

Number of aliments: 2

Crossover position: 1

Mutation frequency: each 5 generations

We shown the fitness evolution in fig.1 In this case the fitness value arrives to local maxima around generation 15 and stay there until 55 generation arrives, thus probably due to a mutation.

Test 2.

we create a meal with the following parameters:

Population size:15

Generations: 100

Number of aliments: 3

Crossover position: 1

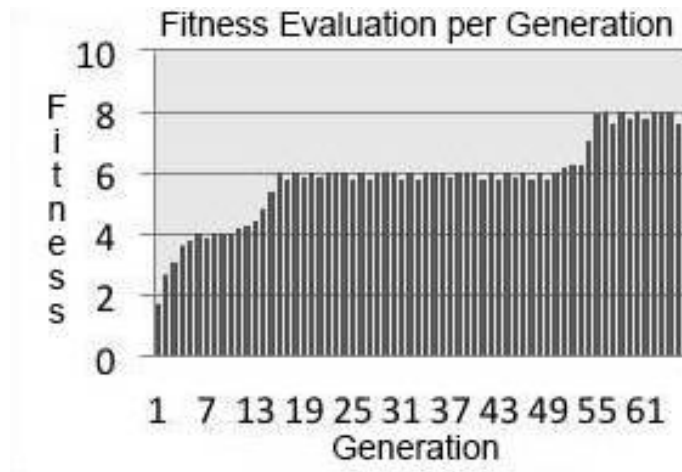


Fig. 1. Fitness value evolution for a breakfast of two aliments

Mutation frequency: each 5 generations

We shown the fitness evolution in fig.2. In this case the fitness value quickly arrives to local maxima value of six, but in generation twenty one arises to eight, thus probably due to a mutation.

4 Conclusions

In this paper a way to create a diet using genetic algorithms is used. As we shown in the section results we can generate each aliment time separately this give us the ability to satisfy the standard or to create our own percent to satisfy special needs.

The fact of generate seaparately each aliment time give a versatility to the diet generator, we can create days with two heavy aliments in meal time and another with four soft aliments.

We take advantage in the fact that the genetic algorithm dont ever converge to the same result, in this way we can generate different aliment configuration for the same parameters easily

This works can be improved developing the following:

1. Add to the aliment database the ability to generate diets for diabetic people
2. Expand the database to include prepared aliments and their corresponding ingredients
3. Expand the algorithm to consider the physic activities that the individual realizes in a day

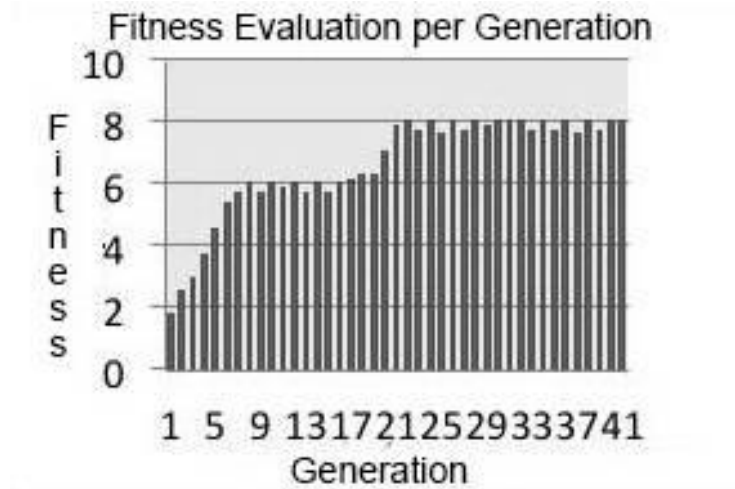


Fig. 2. Fitness value evolution for a meal of three aliments

4. Expand the algorithm to consider prohibited aliments in order to generate diets for allergic people.
5. A way to select or exclude an alimentary group would be desirable, or even a system of preferences would improve this system.

Acknowledgments

The authors are grateful to the secretary of research and postgraduate of National Polytechnic Institute for the economic support given to the multidisciplinary project SIP-20130307

References

1. <http://www.who.int/mediacentre/factsheets/fs311/es/>. Tech. rep., WHO, 2014.
2. KONAR, A. *Artificial Intelligence and Soft Computing: Behaviorial and cognitive Modeling of the human Brain*. CRC Press, 1999.
3. MITCHELL, T. M. *Machine Learning*. McGraw-Hill, 1997.