

A Comparative Study on Bayesian Optimization Algorithm for Nutrition Problem

Serpil Gumustekin*, Talat Senel, Mehmet Ali Cengiz

Department of Statistics, Faculty of Arts and Science, Ondokuzmayis University, Samsun, Turkey

*Corresponding author: serpil.gumustekin@omu.edu.tr

Received October 14, 2014; Revised November 13, 2014; Accepted November 20, 2014

Abstract In this paper, the Bayesian Optimization Algorithm (BOA), which is one of the multivariate EDA algorithms with graphical model, was investigated. Then BOA was applied to the problem of nutrition for breakfast. The results obtained from BOA were compared to Genetic Algorithm and Linear Programming. At the end of the comparisons, for the problem of a recommended diet for breakfast, BOA approach gives more effective results than the other mentioned methods in terms of time and the cost.

Keywords: *genetic algorithm, optimization, Bayesian network, nutrition, breakfast, nutrients*

Cite This Article: Serpil Gumustekin, Talat Senel, and Mehmet Ali Cengiz, "A Comparative Study on Bayesian Optimization Algorithm for Nutrition Problem." *Journal of Food and Nutrition Research*, vol. 2, no. 12 (2014): 952-958. doi: 10.12691/jfnr-2-12-15.

1. Introduction

Nutrition is the foundation of health at every period of life. It is sometimes done consciously or sometimes unconsciously. Scientists have been working on adequate and balanced diets for many years. Nutrition is the supply of food required by organisms and cells to stay alive. In science and human medicine, nutrition is the science or practice of consuming and utilizing foods. Various nutrients are essential to build up and maintain healthy cell tissues, glands and organs in the human body. Without specific nutrients the body is not able to perform many of its functions, be they metabolic, mental, physical or chemical. The nutrients are available only through food which is one of the most essential factors in maintaining and preserving health. The scientific research has revealed that there is a requirement for more than 40 kinds of nutrients for human life. Also, they determined that each of these nutrients must be received daily.

Mathematical programming techniques are widely used in the analysis of nutritional problems. They have been widely studied in recent years, and an extensive summary of the approaches taken can be found in (Sukhatme, 1961); (Edwardson, 1974); (Anderson and Earle, 1983); (Alpaslan, 1996); (Kaldırım and Kose, 2006); (Lv, 2009); (Sahingöz and Sanlier, 2011).

An optimization problem is the challenge to find the optimal or near optimal solution from a specified set of feasible solutions using some measure for evaluating each individual solution. The task is to find the solution of the highest quality. The Nutrition Problem was chosen because of its historical significance, instructional value and ease of understanding. It can be easily stated as:

Minimize the cost of food eaten during breakfast and subject to the requirements that the breakfast satisfy a

person's nutritional requirements and that not too much of any one food be eaten. Linear programming was developed as a technique by (George B. Dantzig, 1947).

Genetic algorithms are search algorithms based on the mechanics of natural selection and natural genetics. In a GA, two genetic operators, crossover and mutation, are used to simulate the variation. Crossover forms the new population by exchanging some parts between the selected solutions. Mutation slightly modifies some parts of the newly formed solutions to introduce some genetic variation in the new population. The traditional crossover and mutation approach of variation in GAs has been found to be limited for many optimization problems, and therefore, most of the early research in GAs has been focused on the modification of these operators to improve GA performance. In recent years, a probabilistic approach to variation has been proposed where crossover and mutation is replaced by two other operators: distribution estimation and sampling. Distribution estimation is to estimate a probability distribution of solutions from a population, and sampling is to sample the distribution to generate a new population. Algorithms using such an approach to variation are called Estimation of Distribution Algorithms (EDAs) (Muhlenbein & Paaß, 1996; Larranaga & Lozano, 2002). In the literature, EDAs are also called probabilistic model building genetic algorithms (PMBGA) and iterated density estimation algorithms (IDEAs).

Bayesian Optimization Algorithm (BOA) is a specific kind of EDA proposed by (Pelikan et al, 1999) for solving a group of complicated problems efficiently. It combines the idea of using probabilistic models to guide optimization and the methods for learning Bayesian networks. To learn a proper decomposition of the problem, BOA builds a Bayesian network for the set of promising solutions. New candidate solutions are generated by

sampling the built network. BOA generates the initial population of strings at random with a uniform distribution over all possible strings. The population is updated for a number of iterations (generations), each consisting of four steps. First, promising solutions are selected from the current population using a GA selection method, such as tournament and truncation selection. Second, a Bayesian network that fits the population of promising solutions is constructed. Third, new candidate solutions are generated by sampling the built Bayesian network. Fourth, the new candidate solutions are incorporated into the original population, replacing some of the old ones or all of them. The above four steps are repeated until some termination criteria are met.

Bayesian network (Pearl,1998) was used in the proposed Bayesian optimization algorithm to solve the nutrition problem. A Bayesian network is a directed acyclic graph with each node corresponding to one variable, and each variable corresponding to the individual rule by which a nutrition problem will be constructed step by step. The causal relationship between two variables is represented by a directed edge between the two corresponding nodes. The Bayesian optimization algorithm is applied to learn to identify good partial solutions and to complete them by building a Bayesian network of the joint distribution of solutions (Pelikan et al, 1999); (Pelikan and Goldberg, 2000). The conditional probabilities are computed according to an initial set of promising solutions. Subsequently, each new instance for each node is generated by using the corresponding conditional probabilities, until values for all nodes have been generated. If stopping conditions are not met, the conditional probabilities for all nodes in the Bayesian network are updated again using the current set of rule strings. The algorithm thereby tries to explicitly identify and mix promising building blocks.

In our nutrition problem, a linear programming model was constructed using real data to meet the needs of the average daily nutrient requirement for adequate and balanced nutrition with 10 nutrients and 20 foodstuffs for breakfast. Then the model was solved by using Bayesian Optimization Algorithm (BOA), Genetic Algorithm (GA) and Linear Programming (LP). The results were then compared to each other. The results obtained with a Bayesian optimization algorithm show that reduced time and cost savings were achieved.

2. Materials and Methods

The first step is to assign variables for the amounts of food that we are looking for and parameters for the data that we know.

2.1. Linear Programming

In mathematical terms, linear programming is a tool to optimize (minimize or maximize) a linear function of a set of decision variables while respecting multiple linear constraints.

The Nutrition Problem can be formulated mathematically as a linear programming problem as shown below.

Sets:

F: Set of foods

N: Set of nutrients

Parameters:

a_{ij} = amount of nutrient j in food $i, \forall i \in F, \forall j \in N$

c_i = costperservingoffoodi, $\forall i \in F$

F_{\min_i} = minimum number of required servings of food $i,$

$\forall i \in F$

F_{\max_i} = maximum allowable number of servings of

food $i, \forall i \in F$

N_{\min_j} = minimum required level of nutrient $j, \forall j \in N$

N_{\max_j} = maximum allowable level of nutrient $j, \forall j \in N$

Decision Variables:

X_i = Number of servings of food i to consume, $\forall i \in F$

Objective Function:

Minimize the total cost of the breakfast.

$$\text{Minimize } \sum_{i \in F} c_i X_i \tag{1}$$

Constraints:

Constraint Set 1: For each nutrient $j \in N$, at least meet the minimum required level.

$$\sum_{i \in F} a_{ij} X_i \geq N_{\min_j}, \forall j \in N \tag{2}$$

Constraint Set 2: For each nutrient $j \in N$, do not exceed the maximum allowable level.

$$\sum_{i \in F} a_{ij} X_i \leq N_{\max_j}, \forall j \in N \tag{3}$$

Constraint Set 3: For each food $i \in F$, select at least the minimum required number of servings.

$$X_i \geq F_{\min_i}, \forall i \in F \tag{4}$$

Constraint Set 4: For each food $i \in F$, do not exceed the maximum allowable number of servings.

$$X_i \leq F_{\max_i}, \forall i \in F \tag{5}$$

2.1.1. Objective function

We express our decision variables as follows. X_1 :Bread, X_2 :Sausage, X_3 :Wiener, X_4 :Salami, X_5 :Milk, X_6 :White Cheese, X_7 :Cheddar Cheese, X_8 : Tulum Cheese, X_9 :Curd Cheese, X_{10} : Egg, X_{11} : Butter, X_{12} : Margarine, X_{13} : Tomato, X_{14} : Cucumber, X_{15} :Black Olives, X_{16} :Green Olives, X_{17} :Jam, X_{18} :Honey, X_{19} :Sesame Oil, X_{20} :Boiled Grape Juice. The prices corresponding to 1 gram of the nutrients eaten for breakfast are given in the [Table 1](#) and Nutritional composition of foods (100 grams) eaten for breakfast are given in the [Table 2](#).

Table 1. The prices corresponding to 1 gram of the nutrients for breakfast

Nutrient	Price	Nutrient	Price
Bread	0.0023902	Butter	0.0216273
Sausage	0.0350945	Margarine	0.0061875
Wiener	0.0274729	Tomato	0.0023685
Salami	0.0273454	Cucumber	0.0018650
Milk	0.0021037	Black Olives	0.0115866
White Cheese	0.0139591	Green Olives	0.0104020
Cheddar Cheese	0.0200363	Jam	0.0091046
Tulum Cheese	0.0169805	Honey	0.0243544
Curd Cheese	0.0165302	Sesame Oil	0.0111757
Egg	0.0003041	Boiled Grape Juice	0.0100229

Table 2. Nutritional composition of foods (100 grams) for breakfast

Nutritional Composition	Nutrients									
	X ₁ (Bread)	X ₂ (Sausage)	X ₃ (Wiener)	X ₄ (Salami)	X ₅ (Milk)	X ₆ (White Cheese)	X ₇ (Cheddar Cheese)	X ₈ (Tulum Cheese)	X ₉ (Curd Cheese)	X ₁₀ (Egg)
Energy (kcal)	276	179	322	450	50	289	404	215	72	158
Protein (gr)	9.1	21	11.3	23.8	3.3	22.5	27	35	12.4	12.1
Oil (gr)	0.8	10.5	29.4	38.1	1.9	21.6	31.7	5.6	1	11.2
Carbohydrate (gr)	56.4	0	2.4	1.2	4.8	0	1.4	3.2	2.7	1.2
Calcium (mg)	19	20	12	14	122	162	700	0	61	56
Iron (mg)	0.7	4.3	1.3	3.6	0.1	0.5	1	0	0.1	2.1
Potassium (mg)	74	0	159	0	154	0	104	0	86	130
Sodium (mg)	585	0	1	0	50	0	710	0	406	138
Vitamin A	0	0	0	0	205	720	1000	1500	37	520
Vitamin C	0	0	0	0	1	0	0	0	0	0
Price (TL)	0.239	3.509	2.747	2.734	0.210	1.395	2.003	1.698	1.653	0.5

Nutritional Composition	Nutrients									
	X ₁₁ (Butter)	X ₁₂ (Margarine)	X ₁₃ (Tomato)	X ₁₄ (Cucumber)	X ₁₅ (Black Olive)	X ₁₆ (Green Olive)	X ₁₇ (Jam)	X ₁₈ (Honey)	X ₁₉ (Sesame Oil)	X ₂₀ (Boiled Grape Juice)
Energy (kcal)	717	719	22	15	207	144	272	315	516	293
Protein (gr)	0.9	0.9	1.1	0.9	1.8	1.5	0.6	0.3	10.5	0.6
Oil (gr)	81.1	80.5	0.2	0.1	21	13.5	0.1	0	2.8	0.1
Carbohydrate (gr)	0.1	0.9	4.7	3.4	1.1	2.8	70	78.4	53.5	70.6
Calcium (mg)	24	30	13	25	77	90	20	15	91	400
Iron (mg)	0.2	0	0.5	1.1	1.6	2	1	0.8	9	10
Potassium (mg)	26	43	244	160	0	0	88	0	0	0
Sodium (mg)	826	943	3	6	0	0	12	0	0	0
Vitamin A	3058	3307	900	250	60	300	10	0	0	0
Vitamin C	0	0	23	11	0	0	2	4	0	0
Price (TL)	2.162	0.618	0.236	0.186	1.158	1.040	0.910	2.435	1.117	1.002

We created objective function with the help of the Eq.1 is as follows.

$$\begin{aligned} \text{Min } z = & 0.00239020x_1 + 0.0350945x_2 + \\ & 0.0274729x_3 + 0.0273454x_4 + 0.00210370x_5 + \\ & 0.0139591x_6 + 0.0200363x_7 + 0.0169805x_8 + \\ & 0.0165302x_9 + 0.005x_{10} + 0.0216273x_{11} + \\ & 0.00618750x_{12} + 0.00236850x_{13} + 0.00186500x_{14} + \\ & 0.0115866x_{15} + 0.0104020x_{16} + 0.00910460x_{17} + \\ & 0.0243544x_{18} + 0.0111757x_{19} + 0.0100229x_{20} \end{aligned}$$

In this problem, we will be looking for the values of X. The other values are provided as input. The problem constraints can then be written as follows. Coefficients are corresponding to 1 gram of the constraint and objective function

2.1.2. Constraints

Daily minimum standards of nutritional elements for Turkey (Paker, 1996) are given in the Table 3.

Table 3. Daily minimum standards of nutritional elements for Turkey

Nutrients	Average Values
Energy (calorie)	2246
Protein (gr)	68
Oil (gr)	70
Carbohydrate (gr)	326
Calcium (mg)	603
Iron (mg)	13
Potassium (mg)	3500
Sodium (mg)	2400
Vitamin A	4485
Vitamin C	61

Our model is set up for breakfast in the morning, the weighted averages (thought to be three meals a day) 1/3 per cent constants were determined on the right-hand side. For example; energy's the weighted average value is 2246 calorie for a day and we used it $2246/3 \approx 750$ for a breakfast.

We created constraints with the help of the Eq.2, Eq.3, Eq.4 and Eq.5 are as follows.

The minimum energy requirements for breakfast is set at 750 calorie per day.

$$\begin{aligned} & 2.76X_1 + 1.79X_2 + 3.22X_3 + 4.50X_4 + 0.50X_5 + \\ & 2.89X_6 + 4.04X_7 + 2.15X_8 + 0.72X_9 + 1.58X_{10} + \\ & 7.17X_{11} + 7.19X_{12} + 0.22X_{13} + 0.15X_{14} + 2.07X_{15} + \\ & 1.44X_{16} + 2.72X_{17} + 3.15X_{18} + 5.16X_{19} + 2.93X_{20} \geq 750 \end{aligned}$$

The minimum protein requirements for breakfast is set at 23 gram per day.

$$\begin{aligned} & 0.091X_1 + 0.21X_2 + 0.113X_3 + 0.238X_4 + 0.033X_5 + \\ & 0.225X_6 + 0.27X_7 + 0.35X_8 + 0.124X_9 + 0.121X_{10} + \\ & 0.009X_{11} + 0.009X_{12} + 0.011X_{13} + 0.009X_{14} + \\ & 0.018X_{15} + 0.015X_{16} + 0.006X_{17} + \\ & 0.003X_{18} + 0.105X_{19} + 0.006X_{20} \geq 23 \end{aligned}$$

The minimum oil requirements for breakfast is set at 24 gram per day.

$$\begin{aligned} & 0.008X_1 + 0.105X_2 + 0.294X_3 + 0.381X_4 + 0.019X_5 + \\ & 0.216X_6 + 0.317X_7 + 0.056X_8 + 0.01X_9 + 0.112X_{10} + \\ & 0.811X_{11} + 0.805X_{12} + 0.002X_{13} + 0.001X_{14} + 0.21X_{15} + \\ & 0.135X_{16} + 0.001X_{17} + 0.28X_{19} + 0.001X_{20} \geq 24 \end{aligned}$$

The minimum carbohydrate requirements for breakfast is set at 109 gram per day.

$$0.564X_1 + 0.024X_3 + 0.012X_4 + 0.048X_5 + 0.014X_7 + 0.032X_8 + 0.027X_9 + 0.012X_{10} + 0.001X_{11} + 0.009X_{12} + 0.047X_{13} + 0.034X_{14} + 0.011X_{15} + 0.028X_{16} + 0.70X_{17} + 0.784X_{18} + 0.535X_{19} + 0.706X_{20} \geq 109$$

The minimum calcium requirements for breakfast is set at 201 miligram per day.

$$0.19X_1 + 0.20X_2 + 0.12X_3 + 0.14X_4 + 1.22X_5 + 1.62X_6 + 7X_7 + 0.61X_9 + 0.56X_{10} + 0.24X_{11} + 0.30X_{12} + 0.13X_{13} + 0.25X_{14} + 0.77X_{15} + 0.90X_{16} + 0.20X_{17} + 0.15X_{18} + 0.91X_{19} + 4X_{20} \geq 201$$

The minimum iron requirements for breakfast is set at 5 miligram per day.

$$0.007X_1 + 0.043X_2 + 0.013X_3 + 0.036X_4 + 0.001X_5 + 0.005X_6 + 0.01X_7 + 0.001X_9 + 0.021X_{10} + 0.002X_{11} + 0.005X_{13} + 0.011X_{14} + 0.016X_{15} + 0.02X_{16} + 0.01X_{17} + 0.008X_{18} + 0.09X_{19} + 0.10X_{20} \geq 5$$

The maximum potassium requirements for breakfast is set at 1167 miligram per day.

$$0.74X_1 + 1.59X_3 + 1.54X_5 + 1.04X_7 + 0.86X_9 + 1.30X_{10} + 0.26X_{11} + 0.43X_{12} + 2.44X_{13} + 1.60X_{14} + 0.88X_{17} \leq 1167$$

The maximum sodium requirements for breakfast is set at 800 miligram per day.

$$5.85X_1 + 0.01X_3 + 0.50X_5 + 7.10X_7 + 4.06X_9 + 1.38X_{10} + 8.26X_{11} + 9.43X_{12} + 0.03X_{13} + 0.06X_{14} + 0.12X_{17} \leq 800$$

The maximum Vitamin A requirements for breakfast is set at 1495 per day.

$$2.05X_5 + 7.20X_6 + 10X_7 + 15X_8 + 0.37X_9 + 5.20X_{10} + 30.58X_{11} + 33.07X_{12} + 9X_{13} + 2.50X_{14} + 0.6X_{15} + 3X_{16} + 0.10X_{17} \leq 1495$$

The minimum Vitamin C requirements for breakfast is set at 21 per day.

$$0.01X_5 + 0.23X_{13} + 0.11X_{14} + 0.02X_{17} + 0.04X_{18} \geq 21$$

and the other constraints as follows:

$$\begin{aligned} x_1 &\leq 150, x_2 \leq 100, x_3 \leq 100, \\ x_4 &\leq 100, x_5 \leq 100, x_6 \leq 100, \\ x_7 &\leq 100, x_8 \leq 100, x_9 \leq 100, \\ x_{10} &\leq 60, x_{11} \leq 100, x_{12} \leq 100, \\ x_{13} &\leq 100, x_{14} \leq 100, x_{15} \leq 100, \\ x_{16} &\leq 100, x_{17} \leq 100, x_{18} \leq 100, \\ x_{19} &\leq 100, x_{20} \leq 100 \\ x_j &\geq 0 \\ j &= 1, 2, \dots, n \end{aligned}$$

The objective function is the value that we are trying to minimize. In our problem, the objective function is the cost of the entire foods. In our nutrition problem, the number of generations is fixed (up to 100), and the target is to provide effective and adequate nutrition.

2.2. Genetic Algorithm for Nutrition Problem

Genetic Algorithm is a random search algorithm that provides a robust method for searching for the optimum solution to complex problems (Goldberg, 1989). In a GA, the problem is represented by a population of strings (or chromosomes, in biological terminology). Each string comprises a number of blocks, which represent the individual decision variables of the problem (genes). The variables represented in the string can be processed in an evaluation function or fitness function, which is in effect the objective function. Strings are processed and combined according to their fitness (objective function value) in order to generate new strings that contain the best features of two parent strings. Strings with the highest fitness have the greatest chance of contributing to future generations, as in the process of natural selection. Excellent introductions to GAs are given by (Goldberg, 1989) and (Michalewicz, 1992).

Three fundamental operators are involved in manipulating strings and moving to a new generation: selection, crossover, and mutation. The approach taken to the operators of selection, crossover, and mutation can influence the results obtained, and different problems may require different approaches.

The main data structures in the Genetic Algorithm toolbox are chromosomes, objective function values and fitness values. The chromosome data structure stores an entire population in a single matrix of size $N_{ind} \times L_{ind}$, where N_{ind} is the number of individuals in the population and L_{ind} is the length of the genotypic representation of those individuals. The decision variables in the genetic algorithm are obtained by applying some mapping from the chromosome representation into the decision variable space. An objective function is used to evaluate the performance of the decision variables in the problem domain. Fitness values are derived from objective function values through a scaling or ranking function.

The genetic algorithm uses three main types of rules at each step to create the next generation from the current population (Schmitt and Lothar, 2001), (Schmitt and Lothar, 2004):

- Selection rules select the individuals, called parents, that contribute to the population at the next generation.
- Crossover rules combine two parents to form children for the next generation.
- Mutation rules apply random changes to individual parents to form children.

```
%function to minimize a linear equation
function z = objfun(x)
z = (0.00239020*x(1)+0.0350945*x(2)+0.0274729*x(3)+0.0273454*x(4)+0.00210370*x(5)
+0.0139591*x(6)+0.0200363*x(7)+0.0169805*x(8)+0.0165302*x(9)+0.005*x(10)
+0.0216273*x(11)+0.00618750*x(12)+0.00236850*x(13)+0.00186500*x(14)
+0.0115866*x(15)+0.0104020*x(16)+0.00910460*x(17)+0.0243544*x(18)
+0.0111757*x(19)+0.0100229*x(20));
```

Figure 1. M- File showing defined linear equation

In our nutrition problem, we used the Matlab – Genetic Algorithm toolbox for computational results of Genetic algorithm. We used “gatool” and minimize the our objective function (cost of the breakfast). We defined the objective function in a separate m – file as shown in Figure 1.

2.3. Bayesian Optimization Algorithm for Nutrition Problem

This section discusses the proposed Bayesian optimization algorithm for the nutrition problem, including the construction of a Bayesian network.

2.3.1. The Construction of a Bayesian Network

Bayesian networks are also called directed graphical models, in which each node corresponds to one variable, and each variable corresponds to one position in the strings representing the solutions. The relationship between two variables is represented by a directed edge between the two corresponding nodes.

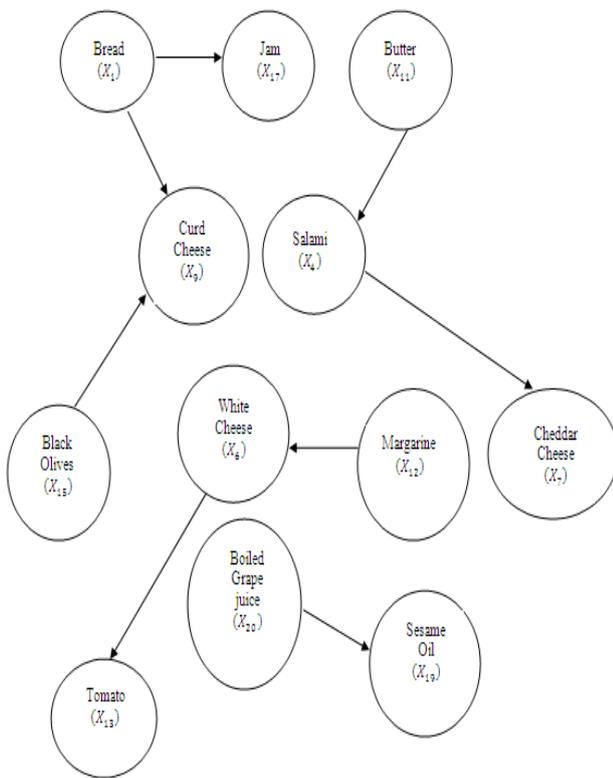


Figure 2. A Bayesian network for breakfast

Bayesian networks are often used to model multinomial data with both discrete and continuous variables by encoding the relationship between the variables contained in the modeled data, which represents the structure of a problem. Furthermore, they are used to generate new data instances or variables instances with similar properties as those of given data. Figure 2 is the Bayesian network constructed for the nutrition problem, which is an acyclic directed graph representing the solution structure of the problem. Since a solution has no parents, it will be chosen from nodes according to their probabilities. The next will be chosen from nodes according to the probabilities conditioned on the previous nodes. This building process is repeated until the last node has been chosen. A path

from nutrient 1 to nutrient m is thus formed where m is the number of the nutrient, representing a new potential solution. Since all probability values for each nutrient are normalized, we suggest the tournament method as a suitable strategy for rule selection (Goldberg, 1989).

The goal of learning is to find the variable values of all nodes that maximize the likelihood of the training data containing a number of independent cases. Thus, the learning in our case amounts to ‘counting’ based on a multinomial distribution. We used R programming for the Bayesian network. It calculates the conditional probabilities of each possible value for each node given all possible values of its parent nodes. Conditional probabilities as follows in Table 4.

Table 4. Conditional probabilities and value of probability

Conditional Probability of Each Decision Variable	Value of Probability
$P(x_1)$	0.4949495
$P(x_2)$	0.4848485
$P(x_3)$	0.4848485
$P(x_4) = P(x_4 x_{11})$	0.4
$P(x_5)$	0.4949495
$P(x_6) = P(x_6 x_{12})$	0.6538462
$P(x_7) = P(x_7 x_4)$	0.5740741
$P(x_8) = P(x_8 x_7)$	0.4
$P(x_9) = P(x_9 x_1, x_{15})$	0.6666667
$P(x_{10})$	0.5151515
$P(x_{11})$	0.5555556
$P(x_{12})$	0.5252525
$P(x_{13}) = P(x_{13} x_6)$	0.4897959
$P(x_{14})$	0.4949495
$P(x_{15})$	0.4747475
$P(x_{16})$	0.5454545
$P(x_{17}) = P(x_{17} x_1)$	0.4081633
$P(x_{18})$	0.4545455
$P(x_{19}) = P(x_{19} x_{20})$	0.3658537
$P(x_{20})$	0.4141414

These probability values were used to generate new solutions. Then, the goal function with each decision variable being a probabilistic value corresponding to 1 gram of the objective coefficients were obtained by multiplying the price value.

2.3.2. A Bayesian Optimization Algorithm

This section introduces a Bayesian optimization algorithm for the nutrition problem and based on the estimation of conditional probabilities.

In the Bayesian optimization algorithm, the first population is generated at random. From the current population, a set of better strings is selected. Any selection method biased towards better fitness can be used, and in this paper, tournament selection is applied. The conditional probabilities of each node in the Bayesian network are computed. New strings are generated by using these conditional probability values, and are added into

the population, replacing some of the old strings. The steps of the Bayesian optimization algorithm for nutrition problem are:

1. Set $t = 0$, and generate an initial population $P(0)$ at random;
2. Use tournament to select a set of promising strings $S(t)$ from $P(t)$;
3. Compute the conditional probabilities of each node according to this set of promising solutions;
4. For the each food, the tournament method is used to select string according to the conditional probabilities of all available nodes, thus obtaining a new string. A set of new strings $O(t)$ will be generated in this way;
5. Create a new population $P(t+1)$ by replacing some strings from $P(t)$ with $O(t)$, and set $t = t + 1$;
6. If the termination conditions are not met (we use 100 generations), go to step 2.

3. Computational Results and Discussions

In this section we describe the computational experiments that were used to test BOA. For all experiments, real data sets as given to us by the TUIK are available. Data set consists of 20 nutrients and their prices. For all data instances, we used the following set of fixed parameters to implement our experiments. These parameters are based on our experience and intuition and thus are not necessarily the best for each instance. We have kept them the same for consistency.

3.1. Details of Algorithms

The detailed computational results over these 20 instances are listed in the tables.

- LP: optimal solutions found with linear programming software (Dowland and Thompson, 2000);
- GA: optimal solutions found with genetic algorithm software (Matlab(R2009b))
- BOA: optimal solutions found with Bayesian optimization algorithm
- Stopping criterion: number of generations = 100, or an optimal solution is found
- Population size = 100
- The number of solutions kept in each generation = 20

- Number of runs per data instance = 20
- Creation Function = uniform
- Selection Function = tournament
- Crossover Function = two point
- Hybrid Function = pattern search

The BOA was coded in Matlab (R2009b), and all experiments were run on the same Pentium (R) 1.60GHz PC with 1GB RAM using the Windows XP operating system. To test the robustness of the BOA, each data instance was run twenty times by fixing the above parameters and varying the pseudorandom number seed at the beginning.

3.2. Analysis of Results

First, let us see the results obtained from all of three methods.

The results obtained from the linear programming solution WinQSB package program: Bread ($x_1 = 120.7768$), Milk ($x_5 = 83.175$), Egg ($x_{10} = 15.5074$), Margarine ($x_{12} = 3.2020$), Tomato ($x_{13} = 90.9427$), Sesame Oil ($x_{19} = 60.5612$) and value of goal function (min.) is 1.4532.

The results obtained from the genetic algorithm solution: Bread ($x_1 = 101.954$), Wiener ($x_3 = 0.125$), Salami ($x_4 = 0.125$), Milk ($x_5 = 99.998$), White Cheese ($x_6 = 1.515$), Cheddar Cheese ($x_7 = 0.291$), Curd Cheese ($x_8 = 1.977$), Egg ($x_{10} = 38.507$), Butter ($x_{11} = 0.001$), Margarine ($x_{12} = 9.743$), Tomato ($x_{13} = 54.963$), Cucumber ($x_{14} = 65.54$), Black Olive ($x_{15} = 0.044$), Green Olive ($x_{16} = 0.028$), Jam ($x_{17} = 7.452$), Sesame Oil ($x_{19} = 29.598$), Boiled Grape Juice ($x_{20} = 28.629$) and value of goal function (min.) is 1.7131.

The results obtained from the Bayesian optimization algorithm solution: Bread ($x_1 = 0.413$), Sausage ($x_2 = 0.166$), White Cheese ($x_6 = 0.257$), Cheddar Cheese ($x_7 = 0.233$), Curd Cheese ($x_8 = 0.259$), Egg ($x_{10} = 0.009$), Margarine ($x_{12} = 0.042$), Jam ($x_{17} = 0.121$), Honey ($x_{18} = 0.288$), Sesame Oil ($x_{19} = 0.206$), Boiled Grape Juice ($x_{20} = 0.107$) and value of goal function (min.) is 1.017.

Table 5. All of the results obtained from Linear Programming, Genetic Algorithm and Bayesian Optimization Algorithm

Nutrients	Daily Minimum	Linear Programming		Genetic Algorithm		Bayesian Optimization Algorithm	
	Standard	The Amount	The Deviation (%)	The Amount	The Deviation (%)	The Amount	The Deviation (%)
Energy (calorie)	750	754.959	0.04	751.99	0.0199	-	
Protein (gr)	23	22.999	0.00001	23	0	-	
Oil (gr)	24	16.95	0.07	28.71	0.04	-	
Carbohydrate (gr)	109	109	0	108.99	0.000001	-	
Calcium (mg)	201	201	0	336.909	1.359	-	8%
Iron (mg)	5	7.159	0.02	5.60	0.006	-	
Potassium (mg)	1167	460.90	7.061	529.72	6.37	-	
Sodium (mg)	800	802.45	0.0245	799.99	0.00002	-	
Vitamin A	1495	1175.52	3.19	1430.30	0.646	-	
Vitamin C	21	21.74	0.0074	20.99	0.0000009	-	
Cost		1.45		1.71		1.017	
Total deviation		10%		8%		8%	
Solution time		35sn		15sn		11sn	

Comparing the computational results on three methods, without learning the conditional probabilities, the results are much weaker. For the linear programming, 6 out of 20 foods are solved to optimality and this ratio for the genetic algorithm is 16 out of foods and for BOA is 11 out of 20 foods. Bayesian optimization algorithm solved the problem of nutrition in 11 seconds, genetic algorithm solved it in 15 seconds and linear programming solved it in 35 seconds. Thus, BOA has provided the solution to our problem in a shorter time than GA and LP. In terms of daily nutrient needs of the deviations, BOA had deviations lower than GA and LP. BOA solution is used to calculate the values of deviations of the posterior values, but the value of the deviation of each constraint is not possible to obtain from BOA. Therefore, only the total deviation value is written. The behavior of an individual run of the Bayesian algorithm is as expected. The optimal solution cost is 1.017 Turkish Lira. This is the least cost all of three methods. As a result, nutrient needs, and the cost advantage that deviations from the same to reduce the time to solve the problem in terms of nutrition for breakfast, Bayesian optimization algorithm can be used in an effective manner. All of the results are in the [Table 5](#).

4. Conclusion

In this paper, BOA has been proposed to solve a nutrition problem. How to meet the daily needs of nutrients from food, at minimal cost and focusing on how much should be eaten. For this purpose, a linear programming model was created using real data, Matlab programming language software package, WinQSB packet program and R - Programming were analyzed. At the end of the comparisons, for the problem of a recommended diet for breakfast, BOA approach gives more effective results than the other mentioned methods in terms of time and the cost. BOA is an effective method to solve the problem about how to implement explicit learning from past solutions. Unlike most existing approaches, Bayesian Optimization Algorithm has the ability to build schedules by using flexible, rather than fixed rules. Experimental results from real-world nutrition problems have demonstrated the strength of the Bayesian optimization algorithm.

It is also hoped that this research will give some preliminary answers about how can we have a best price and nutritious breakfasts and may therefore be of interest to practitioners and researchers in areas of nutrition and evolutionary computation. In future, we will try to consider the problems of nutritional people or clinical examination of patients in special cases.

Acknowledgement

The authors thank the anonymous referees for their interesting and helpful comments.

References

- [1] Alpaslan F. (1996), "Türkiye'de 6 Büyük İilde Doğrusal Programlama ile Optimum Beslenme Maliyetinin Minimizasyonu (1994-1997)". Ondokuz Mayıs University, Fen-Edebiyat Fakültesi Araştırma Fonu. Yayın No: F.150.s. 6-8.
- [2] Anderson A.M. and Earle M.D. (1983), "Diet Planning in the Third World by Linear and Goal Programming". J. Opl. Res. Soc. Vol.34. pp.9-16.
- [3] Dantzig, G.B (1947), "Maximization of a linear function of variables subject to linear inequalities, T.C. Koopmans (ed.): Activity Analysis of Production and Allocation", New York-London 1951 (Wiley & Chapman-Hall), pp. 339-347.
- [4] Dowland KA, Thompson JM, (2000), Solving a nurse scheduling problem with knapsacks, networks and tabu search. J Oper Res Soc 51:825-833.
- [5] Edwardson W. (1974), *The Design of Nutritional Food Products for a Developing Country*. A Thesis for the Degree of Ph. D. in Product Development, Massey University.
- [6] Kaldırım E. and Köse Z. (2006), "Application of a Multi-objective Genetic Algorithm to the Modified Diet Problem", *Genetic and Evolutionary Computation Congress (GECCO), Undergraduate Student Workshop*, Seattle, USA.
- [7] Larranaga P., Lozano J.A., (2002), *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, Kluwer Academic Publishers, Boston.
- [8] Lv Y. (2009), "Multi - Objective Nutritional Diet Optimization Based on Quantum Genetic Algorithm"; in *Proc. ICNC (4)*, pp. 336-340.
- [9] Goldberg D.E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, USA.
- [10] Michalewicz, Z. (1992). Genetic Algorithms + data Structures = evolution programs, Springer-Verlag, New York, Inc., New York.
- [11] Muhlenbein H., Paaß G., (1996), "From recombination of genes to the estimation of distributions I. binary parameters", *PPSN IV: Proceedings of the 4th International Conference on Parallel Problem Solving from Nature*, Springer-Verlag, pp. 178-187.
- [12] Paker H.S. (1996), *Besinlerin Yenebilen 100 gramlarının Enerji ve Besin Öğelleri Değerleri, Sporda Beslenme*, Gen Matbaacılık ve Reklamcılık, Ankara.
- [13] Pearl J., 1988. *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufman Publishers, Palo Alto, CA.
- [14] Pelikan M., Goldberg D.E., Cant'ü-Paz E., (1999). BOA: The Bayesian optimization algorithm. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-99)*, 1, 525-532. Also IlliGAL Report No. 99003.
- [15] Pelikan M., Goldberg D.E., (2000), "Hierarchical problem solving by the Bayesian optimization algorithm, in D. Whitley, D. Goldberg, E. Cant'ü-Paz, L. Spector, I. Parmee & H.-G. Beyer (eds), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2000)*, Morgan Kaufmann, pp. 267-274.
- [16] Sahingoz S.A. and Sanlier N., (2011), "Compliance with Mediterranean Diet Quality Index (KIDMED) and nutrition knowledge levels in adolescents. A case study from Turkey", *Appetite*, Volume 57, Issue 1, August 2011, Pages 272-277.
- [17] Schmitt, Lothar M.,(2001) "Theory of Genetic Algorithms", *Theoretical Computer Science*, pp. 1-61.
- [18] Schmitt, Lothar M., (2004) "Theory of Genetic Algorithms II: models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling", *Theoretical Computer Science*, pp. 181-231.
- [19] Sukhatme P.V., (1961), "The World's Hunger and Future Needs in Food Supplies", *Journal of Royal Statistical Society, Series A*, Vol. 124, pp 463-525.
- [20] TUIK (2012). Consumer Price Index. Available at: <http://www.tuik.gov.tr>. Accessed on 3 June 2013.