
APLICATIVO DE PLANO ALIMENTAR EM DIETA NUTRICIONAL

FERNANDO JOSÉ PORTELLA^{1*}; LUCAS RODRIGUES DA SILVA¹; RAFAEL HENRIQUE BORGES¹, FLORISVALDO CARDOZO BOMFIM¹, LUIZ FERNANDO RIBEIRO DE PAIVA¹

¹Universidade de Uberaba

*e-mail: fjportella@edu.uniube.br

RESUMO - O presente estudo se refere à utilização de algoritmos genéticos no desenvolvimento de um programa voltado à realização de cálculos de otimização dos custos de dieta alimentar, buscando apresentar melhor equilíbrio dos alimentos de acordo com as quantidades de proteínas, gorduras e carboidratos previamente sugeridas. A alimentação equilibrada é um dos principais fatores relacionados à construção de uma boa saúde e à prevenção de doenças. A base de informações dos alimentos utilizados foi desenvolvida a partir de um plano alimentar recomendado a um paciente por uma nutricionista. A partir dos dados do referido plano alimentar, foi elaborada a composição inicial de alimentos, que foi organizada em uma Planilha de Cálculo. Para elaboração do algoritmo genético utilizou-se o *software* científico Scilab, que é semelhante a outro *software* amplamente utilizado, o Matlab. Ao executar o programa, surge no Console do Scilab uma lista de alimentos, vinda da planilha importada, para que o usuário possa escolher os alimentos que irão compor a refeição, conforme suas necessidades. Após a seleção dos alimentos e as escolhas do usuário, o programa inicia o processo de cálculo utilizando um Algoritmo Genético (AG). Após o fim da execução do programa, são apresentados no Console do Scilab os resultados dos cálculos. As saídas correspondem aos melhores indivíduos gerados pelo AG, a partir dos cálculos executados a cada geração.

Palavras-chave: Inteligência Artificial, Algoritmos Genéticos, Planilha, Scilab.

INTRODUÇÃO

A alimentação equilibrada é um dos principais fatores relacionados à construção de uma boa saúde e à prevenção de doenças.

É necessário criar soluções que favoreçam a produção de informações sobre alimentação saudável, pois “existe uma relação direta entre nutrição, saúde e bem-estar físico e mental do indivíduo. As pesquisas comprovam que a boa alimentação tem um papel fundamental na prevenção e no tratamento de doenças”. (Recine; Radaelli, 2002).

O presente estudo se refere à utilização de algoritmos genéticos no desenvolvimento de um programa voltado à realização de

cálculos de otimização dos custos de uma dieta alimentar, buscando apresentar melhor equilíbrio dos alimentos de acordo com as quantidades de proteínas, gorduras e carboidratos previamente sugeridas.

De acordo com Linden (2012, p. 3), Algoritmos Genéticos (AG) são uma técnica de busca extremamente eficiente no seu objetivo de varrer o espaço de soluções e encontrar soluções próximas da solução ótima, quase sem necessitar interferência humana. [...] O problema dos AG é que eles não são tão bons assim em termos de processamento. Logo, eles são mais adequados em problemas especialmente difíceis.

Em 1859, Charles Darwin escreveu o livro “A Origem das Espécies”, no qual explica que, de acordo com a teoria da evolução, todos os seres de um ecossistema (animais, vegetais e insetos) disputam entre si pelos recursos limitados, como comida e água. Aqueles que não tiverem êxito, tendem a ter menor número de descendentes, ocorrendo assim menor possibilidade de que seus genes sejam transmitidos para futuras gerações. Isso é chamado de seleção natural. Os indivíduos sobreviventes, ao se reproduzirem, têm a capacidade de gerar descendentes que são relativamente distintos de seus progenitores, adaptados ao ambiente em que vivem. (Linden, 2012).

Linden (2012) afirma que os primeiros estudos a respeito de Algoritmos Genéticos foram efetuados na década de 1940, no qual os cientistas começaram a se inspirar na natureza para criar um ramo da inteligência artificial. Em 1975, Holland publicou o livro “Adaptation in Natural and Artificial Systems” (Holland, 1975), em que apresenta os algoritmos genéticos, estudando a adaptação e a evolução no mundo real simuladas dentro do computador, a fim de encontrar boas soluções para problemas extremamente difíceis.

Algoritmos Genéticos (*Gas-Genetic Algorithms*) constituem uma técnica de busca e otimização, altamente paralela, inspirada no princípio Darwiniano de seleção natural e reprodução genética. (Pacheco, 1999, p. 1)

REFERENCIAL TEÓRICO

Materiais

O presente estudo foi desenvolvido a partir de um plano alimentar recomendado a um paciente por uma nutricionista. A partir dos dados do referido plano alimentar foi elaborada a composição inicial de alimentos, que foi organizada em uma Planilha de Cálculo, conforme ilustrado na Figura 1, a seguir.

Para elaboração do algoritmo genético, utilizou-se o *software* científico Scilab, semelhante a outro amplamente utilizado, o

Matlab. O Scilab, no entanto, é um *software* distribuído de forma gratuita sob a licença GNU General Public License (GPL).

METODOLOGIA

A planilha com a base de dados (Figura 1) contém 16 (dezesseis) referências a alimentos retirados da lista de alimentos prescritos para o almoço no Plano Alimentar a ser utilizado para importação no Scilab.

	A	B	C	D	E	F	G	H	I
1	Calorias	195,92	119,7	310,48					
2	Gramas	48,98	13,3	77,62					
3	CÓDIGO	PRODUTOS	PROTEÍNA	GORDURA	CARBOIDRATOS	KCAL	%DIÁRIO	PORÇÃOg	R\$/kg
4	1	Beterraba Crua	0,49	0	0,11	0,49	0,0002	1	0,005
5	2	Tomate	0,011	0,002	0,031	0,15	0,0075	1	0,004
6	3	Cenoura ralada	0,013	0,002	0,077	0,34	0,00017	1	0,004
7	4	Brócolis Cozido	0,02	0,01	0,04	0,25	0,01	1	0,0035
8	5	Aroz	0,025	0,002	0,291	1,28	0,00067	1	0,017
9	6	Feijão Carioca	0,048	0,005	0,136	0,76	0,00038	1	0,007
10	7	Lentilha	0,063	0,005	0,163	0,93	0,00465	1	0,0105
11	8	Cabotia	0,014	0,007	0,108	0,48	0,00024	1	0,004
12	9	Mandioca Cozida	0,01	0	0,3	1,25	0,06	1	0,006
13	10	Abobrinha	0,0064	0,0005	0,0393	0,16	0,008	1	0,00455
14	11	Berinjela	0,007	0	0,045	0,19	0,0095	1	0,004
15	12	Couve refogada manteiga	0,017	0,066	0,087	0,9	0,00045	1	0,017
16	13	Batata Doce	0,006	0,001	0,184	0,77	0,000385	1	0,005
17	14	File Frango Grelhado	0,3	0,025	0	1,39	0,000795	1	0,018
18	15	Ovo	0,16	0,19	0,01	2,4	0,12	1	0,0375
19	16	Carne Vemetha	0,359	0,079	0	2,19	0,1095	1	0,02639

Figura 1: Planilha com a composição dos alimentos

As duas primeiras linhas, indicam para o Algoritmo Genético (AG) as necessidades em calorias e gramas de Proteína, Gordura e Carboidrato escolhidas pelo usuário para uma refeição (almoço). Essa composição foi elaborada com base em uma sugestão de almoço, considerando os valores de referências indicados no *site* Vitat (Vitat, 2023). Essas informações servirão de base para que o algoritmo genético calcule a quantidade de gramas para cada alimento. Nas linhas contidas no intervalo de células entre 4 e 19, encontram-se as informações de cada alimento (Proteínas, Gorduras, Carboidratos e Preço), na proporção de 1g, também obtidas no *site* Vitat (Figura 2).

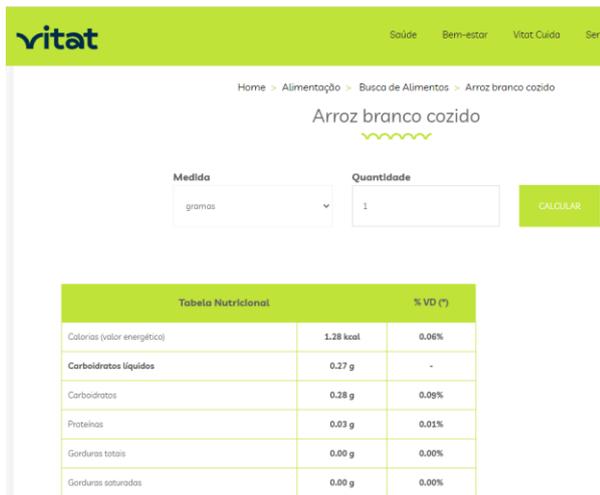


Tabela Nutricional		% VD (*)
Calorias (valor energético)	1.28 kcal	0.06%
Carboidratos líquidos	0.27 g	-
Carboidratos	0.28 g	0.09%
Proteínas	0.03 g	0.01%
Gorduras totais	0.00 g	0.00%
Gorduras saturadas	0.00 g	0.00%

Figura 2: Tabela Nutricional Alimentos

Ao executar o programa, surge no Console do Scilab, uma lista de alimentos, vinda da planilha importada, para que o usuário possa escolher os alimentos que irão compor a refeição, conforme as necessidades do usuário (Figura 3).

```

Scilab 6.1.1 Console

***** LISTA DE ALIMENTOS *****

"1"  "Beterraba Crua"
"2"  "Tomate"
"3"  "Cenoura ralada"
"4"  "Brócolis Cozido"
"5"  "Arroz"
"6"  "Feijão Carioca"
"7"  "Lentilha"
"8"  "Cabotiá"
"9"  "Mandioca Cozida"
"10" "Abobrinha"
"11" "Berinjela"
"12" "Couve refogada manteiga"
"13" "Batata Doce"
"14" "Filé Frango Grelhado"
"15" "Ovo"
"16" "Carne Vemelha"

*****

***** CÓDIGOS PARA EXECUÇÃO *****

"0 - Executar Cálculo // -2 = Cancelar Operação"

*****

Digite o código do Alimento:

```

Figura 3: Tela Inicial do Programa para Seleção dos Alimentos

Após a seleção dos alimentos, e a escolha do usuário para a execução do cálculo, o programa inicia o processo de cálculo do Algoritmo Genético, que segue o ciclo de funcionamento do AG, conforme ilustrado na Figura 4.

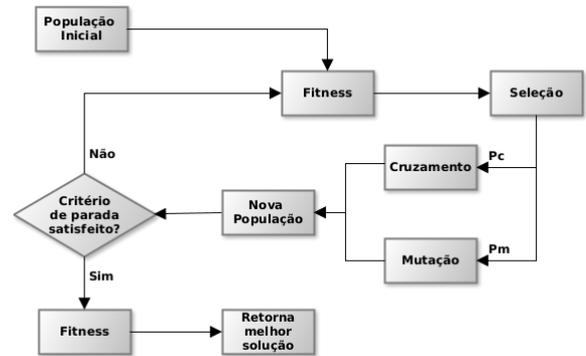


Figura 4: Funcionamento de um Algoritmo Genético Padrão

A população para esse sistema será composta de 500 indivíduos, inicialmente gerada de forma aleatória. Após a inicialização da população inicial, são definidas as quantidades de “gerações” ou “épocas”, o que corresponde à quantidade de vezes que o algoritmo é executado, sendo também este o critério que determina o ponto de parada. A cada nova geração, uma nova população é gerada.

Um indivíduo de um AG é uma abstração de um indivíduo do mundo real. A definição de um indivíduo envolve simplificar aspectos do mundo real e representa uma possível solução para o problema em questão. Nesse contexto, “[...] um conjunto de indivíduos forma a população de um algoritmo genético” (Kato; Rodrigo, 2021, p. 1).

De acordo com Linden (2012, p. 163), a inicialização da população, na maioria dos trabalhos feitos na área, é feita da forma mais simples possível, fazendo-se uma escolha aleatória independente para cada indivíduo da população inicial. A lei das probabilidades sugere que teremos uma distribuição que cobre praticamente todo o espaço de soluções, mas isto não pode ser garantido, pois a população tem tamanho finito. [...] A prática da área, como dito acima, é usar a estratégia mais simples de inicialização, que consiste em simplesmente escolher n indivíduos de forma aleatória. Isto não é devido a preguiça, mas sim ao fato de que a inicialização aleatória de forma geral gera uma boa distribuição das soluções no espaço de busca (Linden, 2012, p. 71).

Para o correto funcionamento do algoritmo genético, é necessário definir cuidadosamente o tamanho da população para que haja eficácia no funcionamento do programa desenvolvido e para que boas soluções sejam encontradas. Nesse sentido é necessário se apoiar no que afirma Linden (2012, p. 163):

O desempenho do algoritmo genético é extremamente sensível ao tamanho da população. [...] Caso este número seja pequeno demais, não haverá espaço para termos uma variedade genética suficientemente grande dentro da nossa população, o que fará com que o algoritmo seja incapaz de achar boas soluções e caso este número seja grande demais, o algoritmo demorará demais e poderemos estar nos aproximando perigosamente de uma busca exaustiva.

O cromossomo é uma representação dos indivíduos de uma população. Como na genética biológica, um cromossomo é composto por genes. Em AG, um cromossomo é uma sequência de *bits* e cada bit é um gene. Para este trabalho, o número de cromossomos será exatamente igual ao número de alimentos selecionados pelo usuário na tela inicial do programa. O número de cromossomos está também ligado à geração da população, uma vez que para cada alimento será gerada uma população de 500 indivíduos. Supondo que o usuário tenha selecionado 8 alimentos, o programa irá gerar uma matriz com 500 linhas (número de indivíduos) e 8 colunas (número de cromossomos).

De acordo com Linden (2012, p. 65), a representação cromossomial [...] é uma maneira de traduzir a informação do nosso problema em uma maneira viável de ser tratada pelo computador. [...] cada pedaço indivisível desta representação é chamado de um gene, por analogia com as partes fundamentais que compõem um cromossomo biológico. [...] o cromossomo nada mais é do que uma sequência de bits e cada gene é somente um *bit*.

Na função fitness será inserida a fórmula para a otimização do custo da dieta. Para garantir que apenas sejam utilizadas as

populações que atendam aos requisitos mínimos de Proteínas, Gorduras e Carboidratos, é utilizada uma fórmula denominada Função de Contorno, que compara as quantidades de proteínas, gorduras e carboidratos desejados com aquelas geradas por meio da multiplicação da população com as proteínas de cada alimento da planilha, e caso as quantidades calculadas não atinjam as quantidades desejadas (de proteínas, gorduras e carboidratos), é aplicada uma penalidade, para que estes valores sejam rejeitados e não sejam utilizados nos cálculos futuros.

A cada interação do sistema, por meio da variável “geração” (que determina a quantidade de vezes o sistema será executado), o custo da dieta é atualizado de acordo com as novas populações, até que o sistema chegue em um ponto de estabilização, gerando assim o melhor custo e também a melhor população, que servirá de quantitativo para cada alimento que irá formar a dieta.

O Método Torneio é uma técnica utilizada em algoritmos genéticos, a fim de selecionar de forma aleatória os futuros pais para a reprodução que passará os seus genes para a próxima geração. Sobre isso, torna-se necessário retomar o autor Linden (2012, p. 163) que explica: “O método de torneio, como o próprio nome diz, consiste em selecionar uma série de indivíduos da população e fazer com que eles entrem em competição direta pelo direito de ser pai, usando como arma a sua avaliação”.

O valor mínimo (k) será de 2 participantes, não havendo limite mínimo definido. Caso o número de participantes seja igual ao tamanho da população, o vencedor sempre será o melhor de todos os indivíduos. E caso o número de participantes seja próximo ao tamanho da população, o resultado tenderá para os melhores indivíduos da população. (Linden, 2012, p. 204).

Conforme demonstrado na Figura 5, a seguir, no exemplo de uma população de 8 indivíduos, utilizando torneio tamanho $k=3$, caso os sorteados sejam x_1 , x_7 e x_8 , cujos valores respectivos são 200, 1 e 4, o vencedor

do torneio será o cromossomo x1, que possui maior valor. Este valor será submetido ao operador *crossover* ou mutação. Na Figura 5, à esquerda, teremos a população com avaliação de cada indivíduo e, à direita, os elementos sorteados com os vencedores marcados com fundo cinza. Como o sorteio é aleatório, nada impede de que um indivíduo seja sorteado mais de uma vez. (Linden, 2012, p. 204-205).

Indivíduo	Fitness	Torneios		
X1	200	X1	X7	X8
X2	100	X2	X3	X5
X3	9500	X6	X4	X4
X4	100	X2	X7	X1
X5	100	X5	X5	X5
X6	10000	X3	X4	X2
X7	1	X4	X2	X6
X8	40	X4	X6	X5

Figura 5: Exemplo de aplicação do método do torneio com $k=3$

Para este trabalho, será utilizado o torneio a 3, semelhante ao demonstrado no exemplo da Figura 5, porém, selecionando dentre os 3 números, o indivíduo que possuir o menor valor, para que se possa encontrar o menor custo.

No operador genético Crossover SBX, ocorre o cruzamento de 2 pais para formação de 2 filhos, que herdam características de cada um dos pais, para gerarem uma nova população, considerando que:

O cruzamento binário simulado (SBX) é um operador de recombinação de parâmetros reais comumente utilizado na literatura do algoritmo evolutivo (EA). O operador envolve um parâmetro que dita a disseminação das soluções descendentes em relação às soluções parentais. Em todas as aplicações do SBX até agora, os pesquisadores mantiveram um valor fixo durante toda a execução da simulação (Deb; Sindhya; Okabe, 2007, p. 2).

No processo de geração das novas populações com os AG, têm-se um processo baseado na reprodução dos seres vivos, conforme explica Linden (2012, p. 33-34),

Nos organismos que utilizam a reprodução sexual, como os humanos e as moscas, cada progenitor fornece um pedaço de material genético chamado gametas. Estas

gametas são resultado de um processo denominado *crossing-over* ou *crossover*, que permitem que os filhos herdem características de ambos os pais mas não sejam exatamente iguais a estes. [...] Após serem duplicados, os cromossomos realizam o *crossover*, processo no qual um pedaço de cada cromossomo é trocado com o seu par.

Linden (2012) cita que após selecionado os pais, um ponto de corte é selecionado, separando os pais em duas partes. O primeiro filho é composto da junção da parte do pai, à esquerda do ponto de corte, com a parte do segundo pai, à direita do ponto de corte, sendo o segundo filho formado com a junção das partes restantes, conforme ilustrado na Figura 6, a seguir.

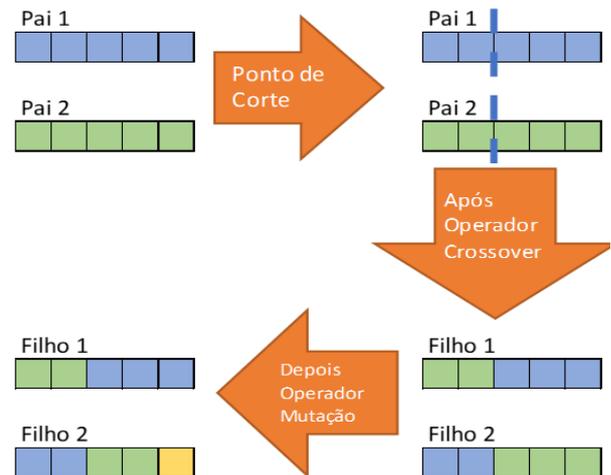


Figura 6: Descrição da operação do operador de *crossover* e um ponto de mutação.

Conforme foi exibido na Figura 4, é estabelecida uma probabilidade de *crossover* (p_c). No presente estudo, a taxa de probabilidade de *crossover* é de 90%, o que significa que há uma alta probabilidade de ocorrer *crossover* entre os indivíduos, promovendo a diversificação e combinação de características para explorar diferentes soluções em busca de melhores resultados.

A Mutação é um processo que possibilita mudanças aleatórias nos genes dos indivíduos, permitindo a geração de indivíduos com novas características, a fim de que sejam exploradas diferentes soluções, evitando que se chegue a soluções não tão boas. Conforme apresentado na Figura 4, é estabelecida uma probabilidade

de Mutação (pm). Neste trabalho, a taxa de chance de ocorrer mutação é de 2%.

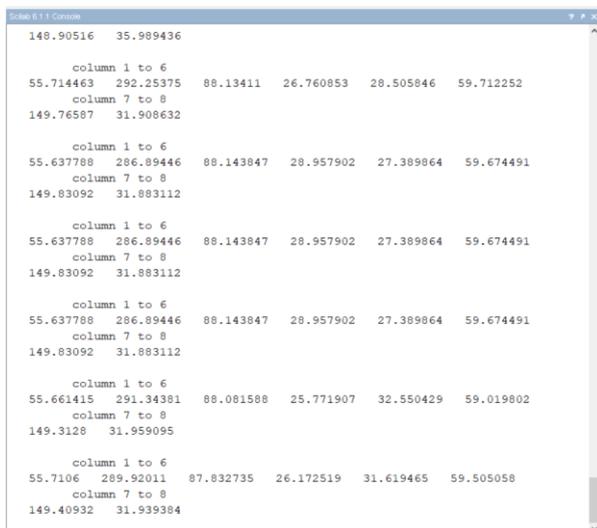
De acordo com Linden (2012, p. 87), depois de compostos os filhos, entra em ação o operador de mutação. Este opera da seguinte forma: ele tem associada uma probabilidade extremamente baixa (da ordem de 0,5%) e nós sorteamos um número entre 0 e 1. Se ele for menor que a probabilidade predeterminada então o operador atua sobre o gene em questão, alterando-lhe o valor aleatoriamente. Repete-se então o processo para todos os genes componente dos dois filhos.

Ainda, sobre o processo de mutação Linden (2012, p. 149) complementa:

O operador de mutação é fundamental para um AG, pois é ele que garante a continuidade da existência de diversidade genética na população, enquanto que o operador de *crossover* contribui fortemente para a igualdade entre os indivíduos. [...] podemos dizer que o operador de mutação é uma heurística exploratória, injetando novos cromossomos na população e permitindo que o AG busque soluções fora dos limites definidos pela população inicial.

RESULTADOS E DISCUSSÃO

Durante a execução do programa, são exibidas em tempo real no console do Scilab as quantidades de gramas calculadas em cada geração do AG, conforme está ilustrado na Figura 7.



```

Scilab 6.1.1 Console
148.90516 35.989436
column 1 to 6
55.714463 292.25375 88.13411 26.760853 28.505846 59.712252
column 7 to 8
149.76587 31.908632
column 1 to 6
55.637788 286.89446 88.143847 28.957902 27.389864 59.674491
column 7 to 8
149.83092 31.883112
column 1 to 6
55.637788 286.89446 88.143847 28.957902 27.389864 59.674491
column 7 to 8
149.83092 31.883112
column 1 to 6
55.637788 286.89446 88.143847 28.957902 27.389864 59.674491
column 7 to 8
149.83092 31.883112
column 1 to 6
55.661415 291.34381 88.081588 25.771907 32.550429 59.019802
column 7 to 8
149.3128 31.959095
column 1 to 6
55.7106 289.92011 87.832735 26.172519 31.619465 59.505058
column 7 to 8
149.40932 31.939384

```

Figura 7: Execução dos Cálculos

Após o fim da execução do programa, são apresentados no Console do Scilab os resultados dos cálculos, conforme apresentado na Figura 8. As saídas correspondem aos melhores indivíduos gerados pelo AG, a partir dos cálculos executados a cada geração.



```

Scilab 6.1.1 Console
"=====RESULTADO FINAL=====
"NECESSIDADES DIÁRIAS
"Proteínas = 48.98gr(s)
"Gorduras = 13.3gr(s)
"Carboidratos = 77.62gr(s)
"-----
"CUSTO R$"
"Valor R$ = 6.9895554
"-----
"QUANTIDADE PRODUTOS
"1 - Beterraba Crua = 55.807519gr(s)
"2 - Tomate = 296.53666gr(s)
"5 - Arroz = 87.578798gr(s)
"6 - Feijão Carioca = 23.570545gr(s)
"8 - Cabotia = 31.304319gr(s)
"9 - Mandioca Cozida = 60.306365gr(s)
"12 - Couve refogada manteiga = 149.43131gr(s)
"16 - Carne Vemella = 31.949582gr(s)
"=====RESULTADO FINAL=====

```

Figura 8: Resultados

São apresentados ao usuário os seguintes resultados:

- Os valores das necessidades diárias, para que o usuário possa comparar com as necessidades presentes na planilha originalmente importada (Figura 1).
- O melhor custo encontrado, que será o valor da refeição em Reais (R\$);
- A quantidade de produtos, que corresponde à quantidade de gramas para cada alimento. Valores estes correspondentes aos melhores indivíduos calculados durante as gerações.

Diante dos resultados obtidos, verificamos que, embora o projeto consiga chegar nos valores das necessidades diárias, ainda há necessidade de ajustes no código, para que a quantidade de gramas possa ser o mais próximo possível da realidade diária, uma vez

que as saídas das quantidades de gramas para cada alimento ainda não condizem realmente com uma solução que possa ser disponibilizada para o consumidor final.

Percebe-se, também que, caso o usuário selecione alimentos que não possuem ou possuem quantidades baixíssimas de alguns dos itens necessários para a composição da dieta – Proteínas, Gorduras e Carboidratos, o AG não consegue encontrar a melhor solução, por não conseguir atingir o resultado esperado para alguns desses 3 itens.

A estrutura que foi definida para os cálculos utilizando o algoritmo genético (AG) foi de buscar o menor custo para a refeição, respeitando os valores das necessidades de Proteínas, Gorduras e Carboidratos. Porém, o AG tem uma tendência de selecionar os alimentos com menor preço que atendam às necessidades sugeridas de proteínas, gorduras e carboidratos. Por esse motivo, alimentos com menor custo, como por exemplo tomate, obteve preferência em relação a alimentos mais caros, como por exemplo carne vermelha.

Contudo, embora os valores ainda não estejam ainda dentro do esperado, e que ainda precisam de ajustes, podemos considerar que o resultado não foi totalmente inadequado. Verifica-se, assim, que AG empregado cumpriu seu papel, pois conseguiu encontrar o menor custo e a quantidade de gramas para cada alimento, que atendem às necessidades de Proteínas, Gorduras e Carboidratos para essa refeição.

CONSIDERAÇÕES FINAIS

No presente estudo, foi desenvolvido um algoritmo genético para calcular o menor custo de uma refeição, levando-se em consideração as necessidades diárias de proteínas, gorduras e carboidratos. Embora tenham sido identificadas algumas limitações e áreas que requerem ajustes adicionais e melhorias, conclui-se que o algoritmo genético cumpriu seu propósito geral de encontrar soluções que atendam às necessidades nutricionais e minimizem o custo da refeição.

Como melhorias a serem implementadas no futuro, a fim de aprimorar a experiência do usuário, pode-se destacar:

- Utilizar uma outra linguagem de programação para a criação de uma interface mais amigável para o usuário;
- Utilizar um *software* de banco de dados, ao invés de planilha Microsoft Excel®;
- Ajustes no código para que o resultado possa sair o mais próximo possível das quantidades reais;
- Possibilitar ao usuário a indicação da quantidade Proteínas, Gorduras e Carboidratos de que ele necessita na refeição.

REFERÊNCIAS

- DEB, K., & AGRAWAL, R. B. (1995), Simulated binary crossover for continuous search space. *Complex Systems, India*, 9(2), 115-148. Disponível em: <https://content.wolfram.com/uploads/sites/13/2018/02/09-2-2.pdf>. Acesso em: 22 dez. 2022.
- DEB, K., SINDHYA, K., & OKABE, T. (2007), Self-adaptive simulated binary crossover for real-parameter optimization. In *GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation* (pp. 1187-1194). ACM. Disponível em: <https://dl.acm.org/doi/10.1145/1276958.1277190>. Acesso em: 01 mar. 2023.
- HOLLAND, J. H. (1975), *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor. The MIT Press.
- KATO, R. (2021), *Algoritmos Genéticos*. Bioinfo. Disponível em: <https://bioinfo.com.br/algoritmos-geneticos/>. Acesso em: 02 jun. 2023.

LINDEN, R. (2012). Algoritmos genéticos (3rd ed.). Rio de Janeiro: Editora Moderna Ltda.

PACHECO, M. A. C. (1999). Algoritmos genéticos: princípios e aplicações. Laboratório de Inteligência Computacional Aplicada. Disponível em:
http://www.inf.ufsc.br/~mauro.roisenberg/ine5377/Cursos-ICA/CE-intro_apost.pdf. Acesso em: 8 dez. 2022.

RECINE, E., & RADAELLI, P. (2002). Alimentação saudável. Brasília: Ministério da Saúde. Secretaria de Políticas de Saúde. Departamento de Atenção Básica. Área Técnica Alimentação e Nutrição Universidade de Brasília. Faculdade de Ciências da Saúde. Departamento de Nutrição. Disponível em:
https://bvsmms.saude.gov.br/bvs/publicacoes/alimentacao_saudavel.pdf. Acesso em: 31 mar. 2023.

SILVA, I., *et al.* (2010). Redes neurais artificiais para engenharia e ciências aplicadas: curso prático (2nd ed.). São Paulo: Artliber Editora Ltda.

VITAT. (2023). Vitat - Busca por informações relativas a alimentos. Higienópolis (SP): Vitat. Disponível em:
<https://vitat.com.br/alimentacao/busca-de-alimentos>. Acesso em: 27 mai. 2023.