# Optimal Weekly Meal Plan: Meeting Daily Nutritional Requirements using Integer programming

## Contents

## Introduction

In the domain of nutrition and meal planning, achieving an optimal balance between cost-effectiveness, nutritional adequacy, and variety presents a formidable challenge. This challenge is especially pronounced in sectors such as healthcare, hospitality, and food service, where delivering well-rounded meals to accommodate diverse dietary needs and preferences is of utmost importance

## Problem Statement

The objective of this project is to design an algorithm or system capable of generating an optimal weekly meal plan that minimizes cost while maximizing nutrition, ensuring it fulfills daily nutritional requirements. The system will accommodate individual dietary restrictions, preferences, and budget constraints, providing users with diverse and well-balanced meal options throughout the week.

## Objective

The proposed problem statement addresses a pressing need in today's society for a comprehensive solution to meal planning that optimizes both nutrition and cost-effectiveness while accommodating individual dietary preferences and restrictions. With an increasing emphasis on health-conscious eating and the rising awareness of the importance of balanced nutrition, individuals seek convenient yet nutritious meal options that align with their dietary goals and budget constraints. By designing an algorithm or system capable of generating weekly meal plans that minimize cost while maximizing nutrition, the proposed solution offers a practical and efficient way for users to achieve their dietary objectives.

## Existing Solutions

### Eat This Much

Eat This Much is a website and mobile app that helps users plan their meals based on their dietary goals, preferences, and budget. It generates personalized meal plans that meet specific calorie targets and nutritional requirements while taking into account user input such as food preferences, allergies, and cooking abilities.

### Key Features

- Customizable Parameters: Users can set their daily calorie targets, specify dietary preferences (e.g., vegetarian, paleo, ketogenic), and input any dietary restrictions or allergies.
- Nutritional Optimization: The platform uses algorithms to generate meal plans that are nutritionally balanced, ensuring adequate intake of macronutrients (carbohydrates, proteins, fats) and micronutrients (vitamins, minerals).
- Cost Considerations: Eat This Much suggests meals that fit within the user's specified budget, taking into account the cost of ingredients and meals.
- Recipe Suggestions: It provides users with a variety of recipes tailored to their preferences and dietary requirements. Users can also customize meal plans by swapping out recipes or adjusting portion sizes.
- Grocery Lists: The platform generates grocery lists based on the ingredients needed for the selected meal plans, making it convenient for users to shop for and prepare their meals.

**Our Model**

**(4.1) Decision Variable:**

$x_{ij}$ = 1, if meal variety j is selected on a day i

$x_{ij}$ = 0, if meal variety j is not selected on a day i

**(4.2) Objective function:** Minimizing overall cost and Maximizing nutrients

$$\text{Min } Z = \sum_{i=1}^{7}\sum_{j=1}^{12} x_{ij} * c_j - \sum_{i=1}^{7}\sum_{j=1}^{12} x_{ij} * N_J$$

$c_j$ : cost of meal variety j

$N_J$ :Nutrient_score of meal variety j

**(4.3) Subject to:**

**Constraint 1:** Exactly 3 meal variety served per day

$$\sum_{j=1}^{12} x_{ij} == 3 \quad \forall i = 1 .. 7$$

**Constraint 2:** The same meal variety should not be repeated more than thrice a week and each meal should be at least served once a week

$$\sum_{i=1}^{7} x_{ij} <= 3 \quad \forall j = 1 .. 12$$

$$\sum_{i=1}^{7} x_{ij} >= 1 \quad \forall j = 1 .. 12$$

**Constraint 3:** All 3 meal types should be different each day

$x_{i1} + x_{i4} + x_{i7} + x_{i10} == 1 \quad \forall i = 1 .. 7$ Meal type: breakfast {1, 4, 7, 10}

$x_{i2} + x_{i5} + x_{i8} + x_{i11} == 1 \quad \forall i = 1 .. 7$ Meal type: lunch {2, 5, 8, 11}

$x_{i3} + x_{i6} + x_{i9} + x_{i12} == 1 \quad \forall i = 1 .. 7$ Meal type: dinner {3, 6, 9, 12}

**Constraint 4:** Maximum and Minimum meal nutrient requirement by our body

$$\sum_{j=1}^{12} x_{ij} * MNV_{jm} >= MinMNV_{m} \quad \forall i = 1 .. 7 \quad \forall m \; in \; meal \; nutrient$$

$$\sum_{j=1}^{12} x_{ij} * MNV_{jm} <= MaxMNV_{m} \quad \forall i = 1 .. 7 \quad \forall m \; in \; meal \; nutrient$$

**(4.4) Parameters**

$MNV_{jm}$: meal nutrient value of nutrient m in meal variety j

$MinMNV_{m}$: minimum meal nutrition value of nutrient m required by our body

$MaxMNV_{m}$: maximum meal nutrition value of nutrient m required by our body

## Dataset

| Nutrients | Vitamin A | Vitamin C | Protein | Carbohydrate | Calories | Fat |
|---|---|---|---|---|---|---|
| **Daily Nutrient Requirement** | 700-1300 | 70-110 | 56-82 | 130-550 | 2000-3000 | 20-35 |

| Meal Variety | Meal Type | Calories (per sreving) | Protein | Carbohydrate | Fat | Vitamin A | Vitamin C | Cost per serving (INR) | Nutrient_Score |
|---|---|---|---|---|---|---|---|---|---|
| **1** | Breakfast | 500 | 20 | 40 | 4 | 200 | 20 | 50 | 784 |
| **2** | Lunch | 900 | 30 | 100 | 10 | 400 | 50 | 60 | 1490 |
| **3** | Dinner | 700 | 21 | 90 | 11 | 450 | 35 | 60 | 1307 |
| **4** | Breakfast | 400 | 15 | 66 | 6 | 150 | 25 | 45 | 662 |
| **5** | Lunch | 1100 | 35 | 120 | 11 | 300 | 45 | 55 | 1611 |
| **6** | Dinner | 810 | 25 | 100 | 12 | 450 | 35 | 42 | 1427 |
| **7** | Breakfast | 520 | 22 | 65 | 7 | 250 | 28.1 | 40 | 892.1 |
| **8** | Lunch | 1500 | 32 | 110 | 15 | 550 | 42 | 70 | 2249 |
| **9** | Dinner | 300 | 24 | 80 | 12 | 450 | 30 | 60 | 896 |
| **10** | Breakfast | 320 | 24 | 50 | 6 | 220 | 25.6 | 55 | 645.6 |
| **11** | Lunch | 1200 | 28 | 90 | 14 | 520 | 40.4 | 65 | 1892.4 |
| **12** | Dinner | 800 | 20 | 122 | 11 | 410 | 30.3 | 72 | 1393.3 |

## Result

| Days (size 7) | Meal_Variety (size 12) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |

## Challenges

When dealing with a larger number of meal varieties, such as the 12 considered here, the objective function takes significantly longer to execute. Conversely, if the number of meal varieties is reduced, the objective function performs better and provides a solution more efficiently.

**Future Extensions**

- User Profiles: Let users set dietary preferences and health goals for personalized meal recommendations.
- Dynamic Nutrition: Adjust meal plans based on age, gender, activity level, and health conditions.
- Ingredient Customization: Allow ingredient swaps for dietary needs and preferences.
- Recipe Generator: Generate new meal ideas and variations based on user preferences and available ingredients.
- Fitness Integration: Sync with fitness trackers to adjust meal plans based on exercise and nutritional needs.
- Community Engagement: Add forums and recipe sharing for user interaction and support.
- Meal Prep Guidance: Offer tips for efficient meal prep and storage.
- Nutritional Education: Provide resources on nutrition and cooking techniques for informed

## Conclusion

The project "Optimal Weekly Meal Plan" tackles the challenge of creating a cost-effective, nutritious meal plan that meets daily dietary needs. Utilizing multi-objective optimization, it balances cost and nutrition while integrating constraints like calorie intake and food preferences. Leveraging comprehensive nutrition data, the algorithm generates personalized meal plans tailored to individual requirements. With applications in healthcare and nutrition counseling, the project enhances meal planning accuracy and effectiveness. Future directions involve incorporating machine learning and real-time data for further advancements in personalized nutrition and wellness optimization.

# Appendix

## Data

```
SheetConnection sheet("dataset.xlsx");


//Meal_Variety from SheetRead(sheet,"Sheet1!A2:A13");
Meal_Type from SheetRead(sheet,"Sheet1!B2:B13");
Meal_Nutrient from SheetRead(sheet,"Sheet1!C1:H1");
Meal_Nutrient_Value from SheetRead(sheet,"Sheet1!C2:H13");
Cost_per_serving from SheetRead(sheet,"Sheet1!I2:I13");
Nutrients_score from SheetRead(sheet,"Sheet1!J2:J13");
//
Min_Meal_Nutrient_Requirement from SheetRead(sheet,"Sheet1!C17:H17");
Max_Meal_Nutrient_Requirement from SheetRead(sheet,"Sheet1!C21:H21");
```

## Code

```
//input parameters
int M_V=12; // meal variety
range Days = 1..7;
{string} Meal_Type=...;
{string} Meal_Nutrient=...;
range Meal_Variety = 1..M_V;
float Meal_Nutrient_Value[Meal_Variety][Meal_Nutrient]=...;
float Cost_per_serving[Meal_Variety]=...;
float Nutrients_score[Meal_Variety]=...;
float Min_Meal_Nutrient_Requirement[Meal_Nutrient]=...;
float Max_Meal_Nutrient_Requirement[Meal_Nutrient]=...;


// Define decision variables
dvar boolean x[Days][Meal_Variety]; // Binary variable representing whether meal i is selected on day j


// Define objective function
minimize sum(i in Days, j in Meal_Variety) x[i][j]*(Cost_per_serving[j]-Nutrients_score[j]); // minimizing cost, maximizing nutrient
```

```
// Define constraints
subject to {
    // Exactly 3 Meal_Variety served per day

    forall(i in Days) {
        sum(j in Meal_Variety) x[i][j] == 3;
      }
         //Same variety should not repeat more than 3 times a week and each meal should be atleast served once in a week
     forall(j in Meal_Variety) {
        sum(i in Days) x[i][j] <= 3;
        sum(i in Days) x[i][j] >= 1;
      }
        //all 3 Meal_Type should be different
         forall(i in Days){
           x[i][1]+x[i][4]+x[i][7]+x[i][10]==1;
           x[i][2]+x[i][5]+x[i][8]+x[i][11]==1;
           x[i][3]+x[i][6]+x[i][9]+x[i][12]==1;
         }
    // Nutritional constraints
    forall(i in Days ,m in Meal_Nutrient)
       sum(j in Meal_Variety) x[i][j] * Meal_Nutrient_Value[j][m] >= Min_Meal_Nutrient_Requirement[m];

      forall(i in Days, m in Meal_Nutrient)
       sum(j in Meal_Variety) x[i][j] * Meal_Nutrient_Value[j][m] <= Max_Meal_Nutrient_Requirement[m];
        }
```

**References**

1. Maillot M, Drewnowski A, Vieux F, Darmon N. Quantifying the contribution of foods with unfavourable nutrient profiles to nutritionally adequate diets. Br J Nutr. 2011;105(8):1133–7.

2. DeSalvo KB, Olson R, Casavale KO. Dietary guidelines for Americans. JAMA 2016;315(5):457–8

3. The American Journal of Clinical Nutrition