

ASSIGNMENT 1: LINEAR PROGRAMMING—THE DIET PROBLEM REVISITED

Alex McCorriston
MSDS 460: Decision Analytics
Northwestern University
Dr. Thomas Miller
April 7, 2024

1. Packaged Food Items Selection

The packaged food items I chose are all from Ralph's. I chose Oatly Original Oatmilk, San Louis Sourdough Bread, Fage Total Plain 0% Milkfat Nonfat Greek Yogurt Tub, Chicken of the Sea Chunk Light Tuna in Water, and Kraft Singles American Sliced Cheese. It was particularly challenging to find items that had non-zero values for Vitamin D, so this nutritional constraint will have the fewest number (three) of food items.

To calculate the price per serving, I took the full price of the item and divided that by the number of servings in each item, which is listed in the nutrition facts. The calculations are:

- Oatmilk: $\frac{6.29}{8} = \$0.79/\text{serving}$
- Sourdough bread: $\frac{5.99}{15} = \$0.40/\text{serving}$
- Greek yogurt: $\frac{7.29}{5} = \$1.46/\text{serving}$
- Tuna: $\frac{1.69}{1} = \$1.69/\text{serving}$
- Cheese: $\frac{5.29}{16} = \$0.33/\text{serving}$



Figure 1. Nutrition Facts Labels.

2. Linear Programming Problem in Standard Form

The following setup is the standard form of my linear programming problem:

Decision Variables:

- oatmilk = number of servings of Oatly Original Oatmilk
- bread = number of servings of San Louis Sourdough Bread
- yogurt = number of servings of Fage Total Plain 0% Milkfat Nonfat Greek Yogurt Tub
- tuna = number of servings of Chicken of the Sea Chunk Light Tuna in Water
- cheese = number of servings of Kraft Singles American Sliced Cheese

Objective Function:

- Minimize $Z = 0.79 * \text{oatmilk} + 0.40 * \text{bread} + 1.46 * \text{yogurt} + 1.69 * \text{tuna} + 0.33 * \text{cheese}$

Subject to the Restrictions:

- $100 * \text{oatmilk} + 230 * \text{bread} + 65 * \text{yogurt} + 270 * \text{tuna} + 230 * \text{cheese} \leq 35000$
- $120 * \text{oatmilk} + 110 * \text{bread} + 90 * \text{yogurt} + 110 * \text{tuna} + 60 * \text{cheese} \geq 14000$
- $3 * \text{oatmilk} + 3 * \text{bread} + 18 * \text{yogurt} + 24 * \text{tuna} + 4 * \text{cheese} \geq 350$
- $3.6 * \text{oatmilk} + 0.3 * \text{tuna} \geq 140$
- $350 * \text{oatmilk} + 6 * \text{bread} + 200 * \text{yogurt} + 330 * \text{cheese} \geq 9100$
- $0.3 * \text{oatmilk} + 1.4 * \text{bread} + 1.08 * \text{tuna} \geq 126$
- $390 * \text{oatmilk} + 30 * \text{bread} + 260 * \text{yogurt} + 188 * \text{tuna} + 60 * \text{cheese} \geq 32900$

and

- $\text{oatmilk}, \text{bread}, \text{yogurt}, \text{tuna}, \text{cheese} \geq 0$

In plain English, this problem aims to determine the optimal number of servings for the five packaged food items I have selected for my weekly diet. The objective is to minimize my

total weekly food expense while ensuring that my consumption meets the U.S. Food and Drug Administration's (FDA) recommended dietary allowances for seven essential nutrients. The minimized cost is calculated by multiplying the number of servings of each food item, determined through problem-solving, by its cost per serving, which I determined in Part I. The seven nutritional components considered are sodium, energy, protein, vitamin D, calcium, iron, and potassium. All the constraints are minimum constraints except for sodium, which is a maximum constraint. The numbers on the right-hand side of the constraints represent the weekly recommended allowances. Finally, there are non-negativity constraints on all the food items because there cannot be negative servings of food.

3. Solve the Linear Programming Problem

The linear programming problem setup code and solution text can be found in the GitHub repository and is labeled as "Problem 1" (McCorriston 2024). I was able to get an optimal solution to the problem. The calculated number of weekly servings of each food item and my minimum weekly cost is:

- **Minimum cost = \$91.45**
- Oatmilk: 78.73 servings
- Sourdough bread: 73.13 servings
- Greek Yogurt: 0 servings
- Tuna: 0 servings
- Cheese: 0 servings

The results indicate that to minimize my food spending for the week while adhering to all the constraints, I should only consume 78.73 servings of oatmilk and 73.13 servings of

sourdough bread. This will result in a weekly expenditure of \$91.45 for food. This is obviously not a sustainable diet and there is evidently a lack of variety, so this issue will be explored next.

4. Revised Linear Programming Problem

To combat the lack of variety of my initial linear programming problem solution, I revised the problem to include an additional constraint that requires at least one serving of each food item during the week. The linear programming problem setup code and solution text can be found in the GitHub repository and is labeled as “Problem 2” (McCorriston 2024). The resulting solution, which again was optimal, is:

- **Minimum cost = \$93.73**
- Oatmilk: 77.47 servings
- Sourdough bread: 73.63 servings
- Greek Yogurt: 1.0 servings
- Tuna: 1.0 servings
- Cheese: 1.0 servings

The above results show that the minimum cost went up from \$91.45 to \$93.73, an increase of \$2.28. Notably, only one serving each of yogurt, tuna, and cheese is included to adhere to the constraints and minimize the overall food expenditure. Thus, there is still quite a lack of variety in my options for the week even with requiring that I get at least one serving of every food. Another way I could add variety to my diet is by including constraints for macronutrients. There is already a constraint for protein, but I could add in additional constraints for fat and carbohydrates, which also have FDA guidelines (Center for Food Safety and Applied Nutrition n.d). Alternatively, a weekly meal rotation plan could introduce thematic diversity into

my diet. Assigning specific themes to each day, such as "Meatless Monday" or "Taco Tuesday," could introduce excitement and variety to meal planning.

5. Large Language Model Diet Problem Creation and Solution

To investigate how well a large language model (LLM) can model and solve my diet problem, I will be using ChatGPT 3.5 (OpenAI 2024). I found that providing it with a very thorough initial prompt enabled it to accurately set up the problem using Python's PuLP package.

Part of the conversation I had with ChatGPT is as follows:



You

Please return the standard form setup for a minimization problem that aims to determine the optimal number of servings for five food items while minimizing total weekly food expenditure and ensuring dietary allowances for seven essential nutrients are met. The five food items are from Ralph's: Oatly Original Oatmilk, San Louis Sourdough Bread, Fage Total Plain 0% Milkfat Nonfat Greek Yogurt Tub, Chicken of the Sea Chunk Light Tuna in Water, and Kraft Singles American Sliced Cheese. The cost per serving for each item is 0.79 for oatmilk, 0.40 for bread, 1.46 for yogurt, 1.69 for tuna, and 0.33 for cheese. The weekly dietary allowance constraints are a maximum of 35000 mg of sodium, a minimum of 14000 kcal, a minimum of 350 g of protein, a minimum of 140 mcg of vitamin d, a minimum of 9100 mg of calcium, a minimum of 126 mg of iron, and a minimum of 32900 mg of potassium. There are also non-negativity constraints for each food item. Note that the amount of each nutritional component for a serving of each food item is as follows:

oatmilk: 100 mg of sodium, 120 kcal, 3 g of protein, 3.6 mcg of vitamin d, 350 mg of calcium, 0.3 mg of iron, and 390 g of potassium

bread: 230 mg of sodium, 110 kcal, 3 g of protein, 6 mg of calcium, 1.4 mg of iron, and 30 g of potassium

yogurt: 65 mg of sodium, 90 kcal, 18 g of protein, 200 mg of calcium, and 260 g of potassium

tuna: 270 mg of sodium, 110 kcal, 24 g of protein, 0.3 mcg of vitamin d, 1.08 mg of iron, and 188 g of potassium

cheese: 230 mg of sodium, 60 kcal, 4 g of protein, 330 mg of calcium, and 60 g of potassium



ChatGPT

To set up the minimization problem in standard form, we need to define decision variables, objective function, and constraints.

Let:

- x_1 = servings of Oatly Original Oatmilk
- x_2 = servings of San Louis Sourdough Bread
- x_3 = servings of Fage Total Plain 0% Milkfat Nonfat Greek Yogurt Tub
- x_4 = servings of Chicken of the Sea Chunk Light Tuna in Water
- x_5 = servings of Kraft Singles American Sliced Cheese

Objective Function:

Minimize:

$$0.79x_1 + 0.40x_2 + 1.46x_3 + 1.69x_4 + 0.33x_5$$


Subject to the following constraints:

1. Nutritional Constraints:

- Sodium: $100x_1 + 230x_2 + 65x_3 + 270x_4 + 230x_5 \leq 35000$
- Calories: $120x_1 + 110x_2 + 90x_3 + 110x_4 + 60x_5 \geq 14000$
- Protein: $3x_1 + 3x_2 + 18x_3 + 24x_4 + 4x_5 \geq 350$
- Vitamin D: $3.6x_1 + 0.3x_4 \geq 140$
- Calcium: $350x_1 + 6x_2 + 200x_3 + 1.08x_4 + 330x_5 \geq 9100$
- Iron: $0.3x_1 + 1.4x_2 + 1.08x_4 \geq 126$
- Potassium: $390x_1 + 30x_2 + 260x_3 + 188x_4 + 60x_5 \geq 32900$

2. Non-negativity Constraints:

- $x_1, x_2, x_3, x_4, x_5 \geq 0$

This standard form setup ensures that the objective is to minimize the total weekly food expenditure while meeting the dietary allowances for the seven essential nutrients and the non-negativity constraints for each food item. 



You

Using Python and the PuLP package, develop the setup code and solution code for this linear programming problem.



ChatGPT

Here's the Python code using the PuLP package to set up and solve the linear programming problem:

```
python Copy code

import pulp

# Create a LP minimization problem
prob = pulp.LpProblem("Minimize_Food_Expenditure", pulp.LpMinimize)

# Define decision variables
x1 = pulp.LpVariable("Servings_Oatmilk", lowBound=0, cat='Continuous')
x2 = pulp.LpVariable("Servings_Bread", lowBound=0, cat='Continuous')
x3 = pulp.LpVariable("Servings_Yogurt", lowBound=0, cat='Continuous')
x4 = pulp.LpVariable("Servings_Tuna", lowBound=0, cat='Continuous')
x5 = pulp.LpVariable("Servings_Cheese", lowBound=0, cat='Continuous')

# Objective Function
prob += 0.79*x1 + 0.40*x2 + 1.48*x3 + 1.69*x4 + 0.33*x5, "Total_Weekly_Food_Expenditure"

# Constraints
# Nutritional Constraints
prob += 100*x1 + 230*x2 + 85*x3 + 270*x4 + 230*x5 <= 35000, "Sodium"
prob += 120*x1 + 110*x2 + 90*x3 + 110*x4 + 60*x5 >= 14000, "Calories"
prob += 3*x1 + 3*x2 + 18*x3 + 24*x4 + 4*x5 >= 350, "Protein"
prob += 3.6*x1 + 0.3*x4 >= 140, "Vitamin_D"
prob += 350*x1 + 8*x2 + 200*x3 + 1.08*x4 + 330*x5 >= 9100, "Calcium"
prob += 0.3*x1 + 1.4*x2 + 1.08*x4 >= 126, "Iron"
prob += 390*x1 + 30*x2 + 280*x3 + 188*x4 + 60*x5 >= 32900, "Potassium"

# Solve the problem
prob.solve()

# Print the results
print("Status:", pulp.LpStatus[prob.status])
print("Optimal Solution:")
for v in prob.variables():
    print(v.name, "=", v.varValue)

print("Total Weekly Food Expenditure: $", pulp.value(prob.objective))
```

Make sure you have the PuLP package installed (`pip install pulp`). This code sets up the problem, defines decision variables, objective function, and constraints as described earlier, and then solves the linear programming problem. Finally, it prints the status, optimal solution for decision variables, and total weekly food expenditure.

Figure 2. Partial ChatGPT 3.5 Conversation.

Upon running this code in my local environment, I obtained results identical to those of my initial linear programming problem, with the exception that the number of servings and minimum cost were not rounded to two decimal places due to the lack of specification in my prompt. Consequently, I asked ChatGPT to update the code to accommodate this rounding, and it executed the task successfully. The full conversation I had with ChatGPT can be found in the GitHub repository for this assignment (McCorriston 2024).

I am thoroughly impressed with this LLM's capabilities. It consistently generates responses within seconds and includes comments in the code for enhanced clarity and comprehension. Additionally, I sought clarification from ChatGPT regarding the final components in the objective function ("Total_Weekly_Food_Expenditure") and the nutritional constraints ("Sodium", "Calories", "Protein", etc.), which I did not have in my personal code. ChatGPT provided an insightful explanation, highlighting the utility of these names for readability and identification of specific components when reviewing the code and results. Thus, I would say that ChatGPT provided more thorough code than my own by enhancing the readability of the problem setup.

Overall, the conversation was highly productive because I was able to build on the initial prompt and get ChatGPT to customize its code to suit my specific requirements. This experience leads me to believe that an LLM agent can absolutely be used to complete this assignment. However, I recognize the critical importance of constructing a specific initial prompt in order to get a desirable output.

References

- Center for Food Safety and Applied Nutrition. n.d. "Daily Value on the Nutrition and Supplement Facts Labels." Accessed on April 7, 2024. U.S. Food & Drug Administration. Updated March 5, 2024. <https://www.fda.gov/food/nutrition-facts-label/daily-value-nutrition-and-supplement-facts-labels>.
- McCorriston, Alex. 2024. "MSDS-460-Assignment-1-Diet-Problem." GitHub Repository. Accessed April 7, 2024. <https://github.com/alexmc808/MSDS-460-Assignment-1-Diet-Problem/tree/main>.
- OpenAI. 2024. Accessed April 7, 2024. <https://chat.openai.com/>.