Sergio Valentini April 6, 2024

MSDS 460 Assignment 1: The Diet Problem

For this Diet Problem, I used five food items from Wild Fork, a grocery store chain that recently opened a location near me. The linear programming problem in standard form for my Diet Problem is as follows:

Decision variables:

- Skin-on Sockeye Salmon with Blackened Butter = x_1
- Herb & Olive Oil Seasoned Pork Tenderloin = x_2
- Mexican Style Street Corn = x_3
- Skin-on Goldband Snapper with Mediterranean Butter = x_4
- Mushroom & Truffle Pizza = x_5

Minimize: $7.14x_1 + 0.98x_2 + 1.33x_3 + 6.57x_4 + 2.99x_5$

subject to:

$570x_1 + 270x_2 + 150x_3 + 310x_4 + 700x_5 <= 35{,}000$ (Sodium constraint in milligrams)

$410x_1 + 130x_2 + 160x_3 + 350x_4 + 350x_5 >= 14{,}000$ (Calorie constraint)

$31x_1 + 23x_2 + 5x_3 + 35x_4 + 12x_5 >= 350$ (Protein constraint in grams)

$17x_4 >= 140$ (Vitamin D constraint in micrograms)

$27x_1 + 13x_2 + 64x_3 + 74x_4 + 158x_5 >= 9{,}100$ (Calcium constraint in milligrams)

$1x_1 + 1x_2 + 1x_3 + 1x_4 + 0.79x_5 >= 126$ (Iron constraint in milligrams)

$527x_1 + 547x_2 + 225x_3 + 734x_4 + 262x_5 >= 32{,}900$ (Potassium constraint in milligrams)

$x_1 >= 0$
$x_2 >= 0$
$x_3 >= 0$
$x_4 >= 0$
$x_5 >= 0$

The objective function is designed to minimize the cost per serving dollar amount, so each coefficient represents the approximate cost of one serving for each food item. For example, salmon is approximately $7.14 per serving. When looking at the constraints, each coefficient represents the nutritional value that each food item contains. As another example, in the sodium constraint, salmon contains approximately 570 milligrams of sodium per serving. Each of the constraints were multiplied by 7 to represent a weekly diet. Lastly, it is not possible to consume negative food items, so each of the decision variables have a non-negativity constraint.

Using the Python PuLP library, I was able to find the optimal solution for this linear programming problem. The minimum serving solution per week is 110.75 servings (12,514.75 grams) of corn, 8.88 servings (1,207.68 grams) of pizza, 8.24 servings (1,631.52 grams) of snapper, and 0 servings of both pork and salmon. In terms of minimum cost, this translates to $147.30 of corn, $54.14 of snapper, and $26.55 of pizza, for a total weekly minimum cost of $227.99. Clearly, this is a financially (and gastronomically) unreasonable diet, though it does meet the nutrition requirements required by the assignment.

If we revise this linear program to require at least one serving of each food item, we get the slightest difference in results. Now, the minimum serving solution per week is 108.11 servings (12,216.43 grams) of corn, 9.7 servings (1,319.2 grams) of pizza, 1 serving (112 grams) of pork, 1 serving (170 grams) of salmon, and 8.24 servings (1,631.52 grams) of snapper. In terms of cost, this translates to $143.79 of corn, $29 of pizza, $0.98 of pork, $7.14 of salmon, and $54.14 of snapper, for a total weekly minimum cost of $235, and an increase of $7.01 from the original minimum weekly cost.

Consistently, the model recommends large quantities of Mexican Style Street Corn, and this makes sense: it is low sodium, low cost, and provides moderate amounts of most nutrients this assignment is concerned with. To introduce more variety in the diet, adjustments should be made to target the nutrients that corn is less abundant in.

For instance, corn provides the least protein by a significant margin, so increasing the weekly protein requirement would likely cause the model to recommend less corn and more salmon/pork/snapper. Additionally, corn provides the least potassium, so increasing the potassium requirement would likely have a similar effect: less corn, more salmon/pork/snapper.

For Part 5 of this assignment, I used ChatGPT 3.5 to attempt to generate Python code for this assignment, and was able to successfully do so. I began with the prompt 'Provide a linear programming model for the Diet Problem', and ChatGPT provided a mathematical model of how a classic Diet Problem would be set up. I then prompted 'Write Python code that can solve a Diet Problem linear programming problem', and it successfully wrote a Python script using PuLP to find the minimum cost of 6 random food items. I then provided ChatGPT foods, nutrition values, and constraints, and asked it to rewrite the code using the values I provided. Not only did it provide a perfect Python script that ran in my editor without any edits, it understood that I had added a maximum constraint of 35,000 grams of sodium, even though the code it originally wrote only contained minimum constraints. I ran the code it provided in VS Code, and it produced the exact same results that I achieved in my code.

For smaller scale linear programming problems, an LLM agent could absolutely be used. In my case, I was able to successfully use code written by ChatGPT without any edits. And in some aspects, I found ChatGPT's code more readable than my own. In the case of more complex optimization problems, ChatGPT will likely still prove a valuable resource, though a keen coding eye will be required to spot areas that may need adjustments. Lastly, my code request was self-contained - I do not need to integrate the code that ChatGPT provided into a larger codeset. In instances like this, it may be more difficult to explain all the necessary context to an LLM agent, but it may still provide helpful guidance.