

# Homework 11: ISYE 6501 - Introduction to Analytics Modeling

## Question 14.1

### Prompt

In the videos, we saw the “diet problem”. (The diet problem is one of the first large-scale optimization problems to be studied in practice. Back in the 1930’s and 40’s, the Army wanted to meet the nutritional requirements of its soldiers while minimizing the cost.) In this homework you get to solve a diet problem with real data. The data is given in the file `diet.xls`.

1. Formulate an optimization model (a linear program) to find the cheapest diet that satisfies the maximum and minimum daily nutrition constraints, and solve it using PuLP.

Turn in your code and the solution. (The optimal solution should be a diet of air-popped popcorn, poached eggs, oranges, raw iceberg lettuce, raw celery, and frozen broccoli. UGH!)

2. Please add to your model the following constraints (which might require adding more variables) and solve the new model:

- a. If a food is selected, then a minimum of 1/10 serving must be chosen. (Hint: now you will need two variables for each food  $i$ : whether it is chosen, and how much is part of the diet. You’ll also need to write a constraint to link them.)
- b. Many people dislike celery and frozen broccoli. So at most one, but not both, can be selected.
- c. To get day-to-day variety in protein, at least 3 kinds of meat/poultry/fish/eggs must be selected. [If something is ambiguous (e.g., should bean-and-bacon soup be considered meat?), just call it whatever you think is appropriate – I want you to learn how to write this type of constraint, but I don’t really care whether we agree on how to classify foods!]

If you want to see what a more full-sized problem would look like, try solving your models for the file `diet_large.xls`, which is a low-cholesterol diet model (rather than minimizing cost, the goal is to minimize cholesterol intake). I don’t know anyone who’d want to eat this diet – the optimal solution includes dried chrysanthemum garland, raw beluga whale flipper, freeze-dried parsley, etc. – which shows why it’s necessary to add additional constraints beyond the basic ones we saw in the video! [Note: there are many optimal solutions, all with zero cholesterol, so you might get a different one.

It probably won’t be much more appetizing than mine.]

## 14.1 Part 1: Optimize Cheapest Diet

- Optimization model (a linear program)
- Find the cheapest diet that satisfies the maximum and minimum daily nutrition constraints
- Solve using **PuLP**
- Solution: Optimal solution diet contains air-popped popcorn, poached eggs, oranges, raw iceberg lettuce, raw celery, and frozen broccoli

PuLP is an linear and mixed integer programming modeler written in Python

PuLP is an LP modeler written in python. PuLP can generate MPS or LP files and call GLPK, COIN CLP/CBC, CPLEX, and GUROBI to solve linear problems.

For optimization, a linear program is developed. A linear program is the simplest type of mathematical program. To determine the mathematical program is a linear program the decision variables must be real variables, the objective must be a linear expression, and the constraints must be linear expressions.

### Load and Inspect Data

```
import pandas as pd

# Load the diet data into a DataFrame
diet_data_path = "~/projects/ISYE6501/HW11/data/diet.xls"
diet_data = pd.read_excel(diet_data_path)

# Get the column names of the dataframe
keys = diet_data.keys()
print(f"Number of columns (keys): {len(keys)}\n{keys}")

## Number of columns (keys): 14
## Index(['Foods', 'Price/ Serving', 'Serving Size', 'Calories', 'Cholesterol mg',
##        'Total_Fat g', 'Sodium mg', 'Carbohydrates g', 'Dietary_Fiber g',
##        'Protein g', 'Vit_A IU', 'Vit_C IU', 'Calcium mg', 'Iron mg'],
##        dtype='object')
```

```
# Print first 5 rows of data
diet_data.head(5)
```

```
##           Foods  Price/ Serving  ... Calcium mg  Iron mg
## 0    Frozen Broccoli          0.16  ...      159.0      2.3
## 1      Carrots,Raw          0.07  ...       14.9      0.3
## 2      Celery, Raw          0.04  ...       16.0      0.2
## 3    Frozen Corn          0.18  ...        3.3      0.3
## 4  Lettuce,Iceberg,Raw          0.02  ...        3.8      0.1
##
## [5 rows x 14 columns]
```

At first glance, the data looks adequate. Check if there are any missing values and output the quantity missing for each column.

```
# Check for nan values in the diet data
nan_vals = diet_data.isna().sum()
print(nan_vals)
```

```
## Foods          3
## Price/ Serving  3
## Serving Size    1
## Calories        1
## Cholesterol mg  1
## Total_Fat g     1
## Sodium mg       1
## Carbohydrates g 1
## Dietary_Fiber g 1
## Protein g       1
## Vit_A IU        1
## Vit_C IU        1
## Calcium mg      1
## Iron mg         1
## dtype: int64
```

## Define Decision Variables

## Setup Objective Function

## Define Constraints

## Solve Linear Program

## Results

### Part 2: Optimize Cheapest Diet with Constraints

#### Part 2a: Linking Two Variables with Constraint

- If food is selected, then a minimum of 1/10 serving must be chosen.
- Variables: Was it chosen (y/n), How much nutrition (quantity)
- Write a constraint to link the variables

#### Part 2b: Celery or Broccoli Not Both

- Many people dislike celery and frozen broccoli. So at most one, but not both, can be selected.

#### Part 2c:

- Day-to-day variety in protein, at least 3 kinds of meat/poultry/fish/eggs must be selected.
  - If ambiguous, make an assumption (e.g., bean and bacon soup)

## References