ISYE 6501 Week 7 Homework

Import the Packages

```
from pulp import *

import pandas as pd
```

Reading the Dataset

```
grub = pd.read_excel("./dietSummer2018.xls")
grub = grub[0:64].values.tolist()
```

Variables for Optimization

```
meals = [d[0] for d in grub]
exp = dict([(d[0], float(d[1])) for d in grub])
Cal = dict([(d[0], float(d[3])) for d in grub])
chlest = dict([(d[0], float(d[4])) for d in grub])
fat = dict([(d[0], float(d[5])) for d in grub])
sodium = dict([(d[0], float(d[6])) for d in grub])
carbs = dict([(d[0], float(d[7])) for d in grub])
fiber = dict([(d[0], float(d[8])) for d in grub])
protein = dict([(d[0], float(d[9])) for d in grub])
vita_A = dict([(d[0], float(d[10])) for d in grub])
vita_C = dict([(d[0], float(d[11])) for d in grub])
calcium = dict([(d[0], float(d[12])) for d in grub])
iron = dict([(d[0], float(d[13])) for d in grub])

diet = LpProblem("Diet Optimization",LpMinimize)
```

For this section, we have created all the variables in order to formulate the optimization problem. In these variables, we are going to focus on the nutrients in order to have the most optimal diet at the most optimal cost based on the given nutrients.

Creating the Initial Variables

```
var_f = LpVariable.dicts("Foods", meals, lowBound = 0 )
varc = LpVariable.dicts("Chosen", meals, lowBound = 0, upBound = 1, cat
= "Binary")
```

Minimizing the Total Cost

```
diet += lpSum([exp[f]*var_f[f] for f in meals]), "Total Cost"
```

This is an objective function to minimize the total cost in order to keep the budget at minimum.

Setting Up the Constraints

```
diet += lpSum([Cal[f]*var_f[f] for f in meals]) >= 1500, 'Minimum
Calories Consumption'
diet += lpSum([Cal[f]*var_f[f] for f in meals]) <= 2500, 'Maximum
Calories Consumption'

diet += lpSum([chlest[f]*var_f[f] for f in meals]) >= 30, 'Minimum
Cholestrol Recommended'
diet += lpSum([chlest[f]*var_f[f] for f in meals]) <= 240, 'Maximum
Cholestrol Limit'

diet += lpSum([fat[f]*var_f[f] for f in meals]) >= 20, 'Minimum Fat
Content'
diet += lpSum([fat[f]*var_f[f] for f in meals]) <= 70, 'Maximum Fat
Limit'

diet += lpSum([sodium[f]*var_f[f] for f in meals]) >= 800, 'Maximum
Sodium Content'
diet += lpSum([sodium[f]*var_f[f] for f in meals]) <= 2000, 'Maximum
Sodium Limit'

diet += lpSum([carbs[f]*var_f[f] for f in meals]) >= 130, 'Minimum
Carbohydrate Content'
diet += lpSum([carbs[f]*var_f[f] for f in meals]) <= 450, 'Maximum
Carbohydrate Limit'

diet += lpSum([fiber[f]*var_f[f] for f in meals]) >= 125, 'Minimum
Fiber Content'
diet += lpSum([fiber[f]*var_f[f] for f in meals]) <= 250, 'Maximum
Fiber Limit'

diet += lpSum([protein[f]*var_f[f] for f in meals]) >= 60, 'Minimum
Protein Content'
diet += lpSum([protein[f]*var_f[f] for f in meals]) <= 100, 'Maximum
Protein Limit'

diet += lpSum([vita_A[f]*var_f[f] for f in meals]) >= 1000, 'Minimum
Vitamin A Content'
diet += lpSum([vita_A[f]*var_f[f] for f in meals]) <= 10000, 'Maximum
Vitamin A Limit'

diet += lpSum([vita_C[f]*var_f[f] for f in meals]) >= 400, 'Minimum
Vitamin C Content'
diet += lpSum([vita_C[f]*var_f[f] for f in meals]) <= 5000, 'Maximum
```

```
Vitamin C Limit'

diet += lpSum([calcium[f]*var_f[f] for f in meals]) >= 700, 'Minimum
Calcium Recommended'
diet += lpSum([calcium[f]*var_f[f] for f in meals]) <= 1500, 'Maximum
Calcium Limit'

diet += lpSum([iron[f]*var_f[f] for f in meals]) >= 10, 'Minimum Iron
Recommended'
diet += lpSum([iron[f]*var_f[f] for f in meals]) <= 40, 'Maximum Iron
Limit'
```

All of these constraints are all referring to the problem in the diet optimization problem. These are all based on the Calories, Cholestrol, Fat, Sodium, Carbohydrate, Fiber, Protein, A Vitamin, C Vitamin, calcium, and Iron limits.

Question 15.2 Part 1

```
print("Solving Part 1.....................")
diet.solve()
print("Status:", LpStatus[diet.status])
for c in diet.variables():
    if c.varValue != 0.0: # Only print items that are not zero
        print(c.name, "=", c.varValue)

print ("Total expenses for food is $%.2f" % value(diet.objective))
```

```
Solving Part 1.....................
Status: Optimal
Foods_Celery,_Raw = 52.64371
Foods_Frozen_Broccoli = 0.25960653
Foods_Lettuce,Iceberg,Raw = 63.988506
Foods_Oranges = 2.2929389
Foods_Poached_Eggs = 0.14184397
Foods_Popcorn,Air_Popped = 13.869322
Total expenses for food is $4.34
```

By running the first part of the optimization, the total expenses for food is $4.34. However, it is as expected from the question on the homework that it will come out to these items returned in the overall diet problem. While looking at the list for the diet, it has been indicated that the majority should be allocated to ceelry, lettuce, and popcorn.

Question 15.2 Part 2 Section A

```
for f in meals:
     diet += var_f[f] <= 10000000*varc[f]
     diet += var_f[f] >= .1*varc[f]
```

Section B: Can Only Choose One, Not Both

```
diet += varc['Frozen Broccoli'] + varc['Celery, Raw'] <=1
```

Section C: Select At Least 3 Meats

```
diet += varc['Tofu'] + varc['Roasted Chicken'] + \
varc['Poached Eggs']+varc['Scrambled Eggs']+varc['Bologna,Turkey'] \
+varc['Frankfurter, Beef']+varc['Ham,Sliced,Extralean'] \
+varc['Kielbasa,Prk']+varc['Hamburger W/Toppings'] \
+varc['Hotdog, Plain']+varc['Pork'] +varc['Sardines in Oil'] \
+varc['Pizza W/Pepperoni'] \
+varc['White Tuna in Water'] >= 3
```

Obtaining Results for Part B

```
print("Part B.......................")
diet.solve()
print("Status:", LpStatus[diet.status])
for a in diet.variables():
    if a.varValue != 0.0: # Excluding the zero values
        print(a.name, "=", a.varValue)

print("Additional cost of foods with additional constraints sums up as
$%.2f" % value(diet.objective))
```

```
Part B.......................
Status: Optimal
Chosen_Celery,_Raw = 1.0
Chosen_Kielbasa,Prk = 1.0
Chosen_Lettuce,Iceberg,Raw = 1.0
Chosen_Oranges = 1.0
Chosen_Peanut_Butter = 1.0
Chosen_Poached_Eggs = 1.0
Chosen_Popcorn,Air_Popped = 1.0
Chosen_Scrambled_Eggs = 1.0
Foods_Celery,_Raw = 42.399358
Foods_Kielbasa,Prk = 0.1
Foods_Lettuce,Iceberg,Raw = 82.802586
Foods_Oranges = 3.0771841
Foods_Peanut_Butter = 1.9429716
Foods_Poached_Eggs = 0.1
Foods_Popcorn,Air_Popped = 13.223294
Foods_Scrambled_Eggs = 0.1
Additional cost of foods with additional constraints sums up as $4.51
```

By including the meats for the optimization model, it turns out more expensive at $4.51.

However, the meats can be based on preferences of the individual's choice. In addition, the diet turns out to be much more optimal for the diet instead of the nasty tastes from the first model. Although, it does come at a cost for extra $0.17 to have more items there. In addition, the diet still consists of the celery, lettuce, and popcorn for the majority of the diet as with the original optimization solution though, but at a lesser amount for celery and popcorn. More emphasis has been placed on lettuce though.