**IN402 Unit 8 Assignment**

Laurence T. Burden

Purdue Global University

IN402: Modeling and Predictive Analysis

Steward Huang

January 9, 2024

As a baker, I will look to find the best way to utilize 60 lbs. of peanut butter to create 500 regular and 350 low-fat cookies. The number of cookies represents the constraints in this problem. The pulp Python package will be used to perform prescriptive analysis on the supplied variables.

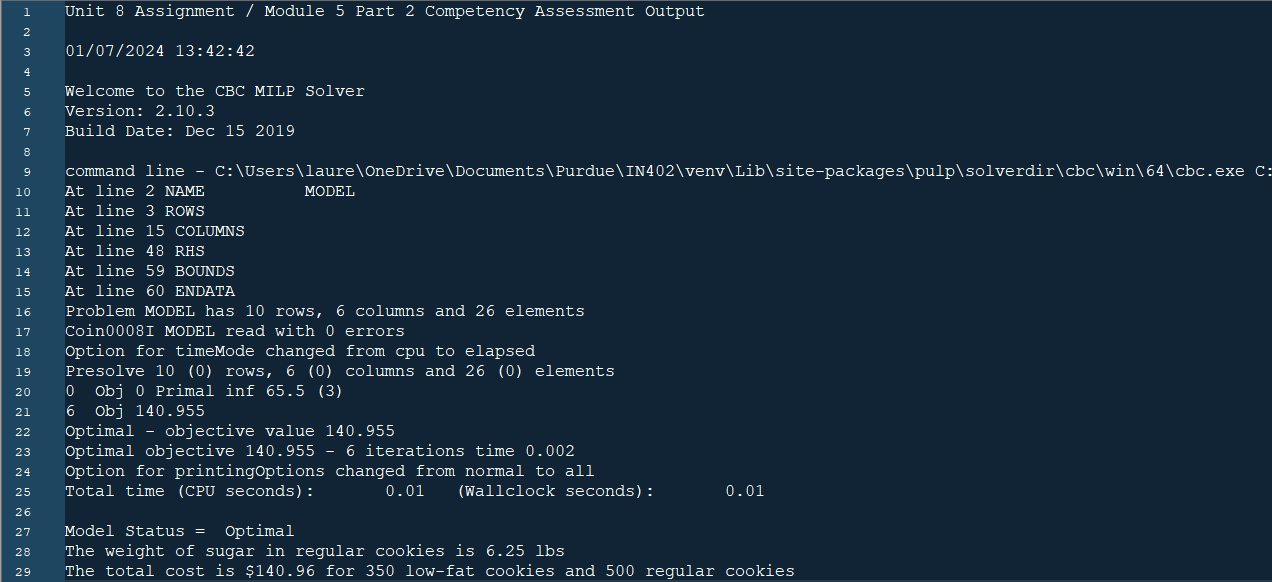
There are two lists of variables used for this script. The first list consists of the cookie types, which are low-fat and regular. The second list contains the ingredients to make a cookie. These are peanut butter, eggs, and sugar. Each variable combination also needs to account for the weight in pounds. A tuple is created to store this information.

Next, the objective function is created to calculate the sum of a list of linear expressions. This uses the lpSum function provided by the pulp package. The values of 4.32 is used for the peanut butter weight, 2.46 for eggs, and 1.86 for sugar.

The cookie count constraints are then added to the model. We also add that regular cookies contain an amount of peanut butter that is greater than or equal to 60% and low-fat cookies contain an amount of peanut butter that is greater than or equal to 40%. The sugar content in each cookie must be less than or equal to 25%.

The amount of materials that we have on hand is added to the model next. This consists of 30 pounds of peanut butter, 20 pounds of eggs, and 17 pounds of sugar. Finally, the model is solved using the solve function of the pulp package. The output of the script is displayed in Figure 1 below.

**Figure 1**

****

The pulp package has been used to create and solve the final model. It shows that the status is considered optimal for our provided constraints. The weight of sugar needed in regular cookies is 6.25 pounds. The final cost for all cookies is $140.96.

This script set out to show if it was possible to create the desired number of low-fat and regular cookies within a bakery, given a set of ingredient constraints. The prescriptive analysis shows that it is possible and provides the method to do so. This will lead to a successful holiday weekend as predicted by the company.

**Appendix A**

**Code**

############################################################

# Author: Laurence Burden

# For: Purdue University Global

#

# Unit 8 Assignment / Module 5 Part 2 Competency Assessment

# ##########################################################

#

# Library imports

import sys

import pulp

from datetime import datetime

# Ignoring warnings

if not sys.warnoptions:

import warnings

warnings.simplefilter("ignore")

# Output Header

print('Unit 8 Assignment / Module 5 Part 2 Competency Assessment Output\n')

print(datetime.now().strftime("%m/%d/%Y %H:%M:%S"), '\n')

# Instantiate the problem class

model = pulp.LpProblem("Cost\_minimising\_blending\_problem", pulp.LpMinimize)

# Create decision variable lists

cookie\_types = ['low-fat', 'regular']

ingredients = ['peanut butter', 'eggs', 'sugar']

# create tuple indices

ing\_weight = pulp.LpVariable.dicts("weight lbs",

((i, j) for i in cookie\_types for j in ingredients),

lowBound=0, cat='Continuous')

# Calculate the sum of a list of linear expressions (objective function)

model+= (

pulp.lpSum([

4.32 \* ing\_weight[(i, 'peanut butter')]

+ 2.46 \* ing\_weight[(i, 'eggs')]

+ 1.86 \* ing\_weight[(i, 'sugar')]

for i in cookie\_types])

)

# Create an objective function and add constraints

# Add constraints for 500 regular and 350 light patties at 0.05 of lbs.

model+= pulp.lpSum([ing\_weight['low-fat', j] for j in ingredients])== 350 \* 0.05

model+= pulp.lpSum([ing\_weight['regular', j] for j in ingredients])== 500 \* 0.05

# Low fat cookie has >= 40% peanut butter, regular >= 60%

model += ing\_weight['low-fat', 'peanut butter'] >= (

0.4 \* pulp.lpSum([ing\_weight['low-fat', j] for j in ingredients]))

model+= ing\_weight['regular', 'peanut butter']>= (

0.6 \* pulp.lpSum([ing\_weight['regular', j] for j in ingredients]))

# Cookies must be <= 25%

model += ing\_weight['low-fat', 'sugar'] <= (

0.25 \* pulp.lpSum([ing\_weight['low-fat', j] for j in ingredients]))

model+= ing\_weight['regular', 'sugar']<= (

0.25 \* pulp.lpSum([ing\_weight['regular', j] for j in ingredients]))

# You have already bought 30 lbs of peanut butter, 20 lbs of eggs and 17 lbs of sugar.

model+= pulp.lpSum([ing\_weight[i, 'peanut butter'] for i in cookie\_types]) <= 30

model+= pulp.lpSum([ing\_weight[i, 'eggs'] for i in cookie\_types]) <= 20

model+= pulp.lpSum([ing\_weight[i, 'sugar'] for i in cookie\_types]) <= 17

# We have at least 23 lbs of peanut butter

model+= pulp.lpSum([ing\_weight[i, 'peanut butter'] for i in cookie\_types]) >= 23

# Solve the problem / Show the model status

model.solve()

print("Model Status = ", pulp.LpStatus[model.status])

# \*\*\*\*\* Status Codes \*\*\*\*\*

# OPTIMAL - Optimal solution exists and is found.

# INFEASIBLE - The problem has no feasible solution.

# UNBOUNDED - The cost function is unbounded.

# UNDEFINED - Feasible solution hasn't been found (but may exist).

# Not Solved - Is the default setting before a problem has been solved.

# Solve the problem - Show the model parameter results

for var in ing\_weight:

var\_value = ing\_weight[var].varValue

print("The weight of {0} in {1} cookies is {2} lbs".format(var[1], var[0], var\_value))

# Calculate the total cost for 500 regular and 350 low-fat cookies.

total\_cost = pulp.value(model.objective)

print("The total cost is ${} for 350 low-fat cookies and 500 regular cookies". format( round(total\_cost, 2)))