Assignment 1

Ran Liu rliu361@gatech.edu
903515184

Problem 1

question 1

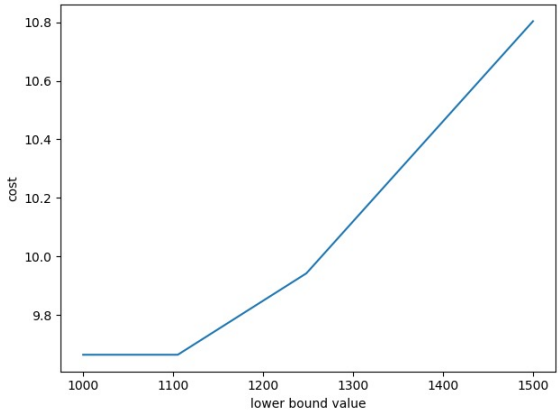

Figure 1: This figure is the optimal objective value (cost of the optimal menu) versus the calorie lower bound.

More details about the code can be found in file diet.py.

question 2

The result has three "phases", which means that the plotted function is a combination of three linear functions.

The plotted function is composed of three lines with the first "transition point" around 1100, the second around 1250. During the first phase, the optimization result is not affected by the lower bound, and the optimal food choice is always $salad * 0.478961 + milk * 9.51903$ with a cost of 9.66455. During the second phase, the optimization result changes with the increase of the amount of the hamburger, the decrease of the amount of the salad and milk, and the increase of the optimal objective value (the cost). An example is $hamburger * 0.0902636 + salad * 0.435985 + milk * 9.41477$ with a cost of 9.6895. During the third phase, the ice cream replaces the salad. An example is $hamburger * 0.971908 + milk * 8.23932 + icecream * 0.219957$ with a cost of 10.1028.

Problem 2

question 1

Indices: $t = 1, 2, 3, ..., 7$ for days in a week, where 1 is Monday, 2 is Tuesday, and so on. $i = 1, 2, 3$ for different types of work week, where 1 is no work on weekend day, 2 is one weekend day for work, and 3 is two weekend days for work.

Data: D_t is the minimum number of workers required on day t. C_t is the dollars per week that needs to be paid to worker who works start on day t. c_i is the dollars per week that needs to be paid to a worker who works on i type of work week, where $c_1 = C_1$, $c_2 = C_2 = C_7$, and $c_3 = C_3 = C_4 = C_5 = C_6$.

Decision variables: x_i is how many workers start their work on day i.

Objective: minimize $\sum_{t=1}^{7} x_t C_t$

Constraints: $\sum_{t=1}^{5} x_t >= D_5$, $\sum_{t=2}^{6} x_t >= D_6$, $\sum_{t=3}^{7} x_t >= D_7$, $x_1 + \sum_{t=4}^{7} x_t >= D_1$, $\sum_{t=1}^{2} + \sum_{t=5}^{7} x_t >= D_2$, $\sum_{t=1}^{3} + \sum_{t=6}^{7} x_t >= D_3$, $x_7 + \sum_{t=1}^{4} x_t >= D_4$.

Bounds: x_t needs to be integers.

Nonnegativity: x_t needs not to be negative numbers.

question 2

The logging file output is as below:

Solution count 2: 405.221 694.108

Optimal solution found (tolerance 1.00e-04)

Best objective 4.052211173001e+02, best bound 4.052211173001e+02, gap 0.0000%

More information can be found inside file call center.py

Problem 3

question 1

Indices: $i = 1, 2, ..., n$ for different fare classes. $k = 1, 2, ..., m$ for different scenarios.

Data: r_i is the revenue for fare classes i, p_i is the penalty for fare classes i, C_i is the capacity for fare classes i, $\alpha_{i,k}$ is the fraction of customers that shows up for fare classes i and scenario k, π_k is the probability of scenario k.

Decision variables: x_i is the number of tickets to sell in fare class i.

Auxiliary variables: $s_{i,k}$ is the number of customers that really show up and get on the plane finally in fare class i and scenario k. In other words, $s_{i,k}$ is the number of customers that really sit on the airplane in fare class i and scenario k.

Objective: maximize $\sum_{k=1}^{m} \pi_k \sum_{i=1}^{n} x_i(r_i - p_i) + s_{i,k}p_i$

Constraints: $s_{i,k} <= C_i$ and $s_{i,k} <= x_i\alpha_{i,k}$ for every k and i, $\sum_{i=1}^{n} x_i <= T$.

Nonnegativity: x_i and $s_{i,k}$ all need not to be negative numbers.

question 2

The logging file output is as below:

Solved in 151 iterations and 0.00 seconds.

Optimal objective 2.667884952e+04

More information can be found inside file goldfarb sit.py.

Problem 4

question 1

The logging file output is as below:

Solved in 399 iterations and 0.00 seconds
Optimal objective 2.741791829e+05
More information can be found inside file airline.py.

question 2

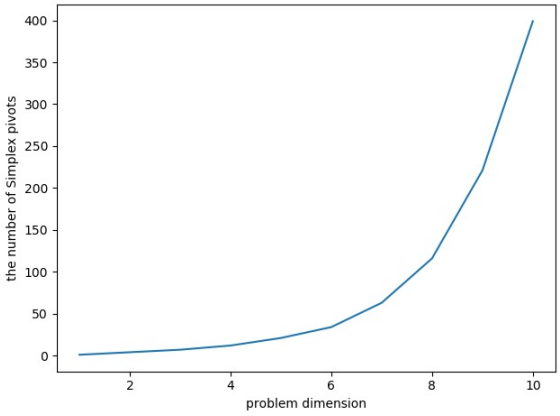

Figure 2: This is the plot that the number of Simplex pivots needed to solve an n–dimensional Goldfarb–Sit problem against the problem dimension n (on the x–axis).

We can see that this plot grows exponentially.