# CS 325 Project 3: Linear Programming

Kyle Guthrie

Michael C. Stramel

Alex Miranda

November 13, 2016

# Problem 1

## Problem 1 Part A

Determine the number of refrigerators to be shipped from plants to warehouses, and then warehouses to retailers to minimize the cost.

## Solution

- In all 1000 units will travel through the network at a minimum cost of $17100.

- Ship 150 units from plant #1 to warehouse #1 at a cost of $1500.

- Ship 200 units from plant #2 to warehouse #1 at a cost of $2200.

- Ship 250 units from plant #2 to warehouse #2 at a cost of $2000.

- Ship 150 units from plant #3 to warehouse #2 at a cost of $1200.

- Ship 100 units from plant #3 to warehouse #3 at a cost of $ 900.

- Ship 150 units from plant #4 to warehouse #3 at a cost of $1200.

- Ship 100 units from warehouse #1 to retailer #1 at a cost of $ 500.

- Ship 150 units from warehouse #1 to retailer #2 at a cost of $ 900.

- Ship 100 units from warehouse #1 to retailer #3 at a cost of $ 700.

- Ship 200 units from warehouse #2 to retailer #4 at a cost of $1600.

- Ship 200 units from warehouse #2 to retailer #5 at a cost of $2000.

- Ship 150 units from warehouse #3 to retailer #6 at a cost of $1800.

- Ship 100 units from warehouse #3 to retailer #7 at a cost of $ 600.

- 150 total units will leave plant #1 (capacity is 150).

- 450 total units will leave plant #2 (capacity is 450).

- 250 total units will leave plant #3 (capacity is 250).

- 150 total units will leave plant #4 (capacity is 150).

- 350 total units will enter warehouse #1, 350 units will leave.

- 400 total units will enter warehouse #2, 400 units will leave.

- 250 total units will enter warehouse #3, 250 units will leave.


- 100 total units will enter retailer #1 (demand is 100).

- 150 total units will enter retailer #2 (demand is 150).

- 100 total units will enter retailer #3 (demand is 100).

- 200 total units will enter retailer #4 (demand is 200).

- 200 total units will enter retailer #5 (demand is 200).

- 150 total units will enter retailer #6 (demand is 150).

- 100 total units will enter retailer #7 (demand is 100).

## Linear Program Formulation

1. Overall idea of problem

   - Refrigerators moving from $n = 4$ plants to $q = 3$ warehouses to $m = 7$ retailers.
   - Not all plants deliver to all warehouses.
   - Not all warehouses deliver to all retailers.
   - Costs of shipping from plants to warehouses vary by pair.
   - Costs of shipping from warehouses to retailers vary by pair.
   - Each plant has a capacity in terms of number of refrigerators it can supply.
   - Each retailer has a capacity in terms of number of refrigerators it demands.

2. What is the goal? What are you trying to achieve?

   - Determine optimal shipping routes ($n$ to $q$ and $q$ to $m$).
   - Determine number of refrigerators moving along each route ($n$ to $q$ and $q$ to $m$).
   - Satisfy the demand of the retailers.
   - Minimize the cost.

3. Identify variables

   - $cp_{ij}$ = cost of moving a refrigerator between plant i and warehouse j
     - ex. $cp_{32} = 8$ = cost of moving from plant 3 to warehouse 2
     - 9 variables
   - $cw_{jk}$ = cost of moving a refrigerator between warehouse j and retailer k
     - ex. $cp_{14} = 10$ = cost of moving from warehouse 1 to retailer 4

- 12 variables
- $s_i$ = capacity (supply) of each plant
  - ex. $s_2 = 450$ = number of refrigerators that plant 2 can supply
  - 4 variables
- $d_k$ = capacity (demand) of each retailer
  - ex. $d_6 = 150$ = number of refrigerators that plant 6 demands
  - 7 variables
- $np_{ij}$ = number of refrigerators shipped from plant i to warehouse j
  - 9 variables
- $nw_{jk}$ = number of refrigerators shipped from warehouse j to retailer k
  - 12 variables

4. Identify constraints

- $s_1 \leq 150$
- $s_2 \leq 450$
- $s_3 \leq 250$
- $s_4 \leq 150$
- $d_1 \geq 100$
- $d_2 \geq 150$
- $d_3 \geq 100$
- $d_4 \geq 200$
- $d_5 \geq 200$
- $d_6 \geq 150$
- $d_7 \geq 100$
- $np_{11} + np_{21} + np_{31} = nw_{11} + nw_{12} + nw_{13} + nw_{14}$
- $np_{12} + np_{22} + np_{32} + np_{42} = nw_{23} + nw_{24} + nw_{25} + nw_{26}$
- $np_{33} + np_{43} = nw_{34} + nw_{35} + nw_{36} + nw_{37}$
- $s_1 = np_{11} + np_{12}$
- $s_2 = np_{21} + np_{22}$
- $s_3 = np_{31} + np_{32} + np_{33}$
- $s_4 = np_{42} + np_{43}$
- $d_1 = nw_{11}$
- $d_2 = nw_{12}$
- $d_3 = nw_{13} + nw_{23}$
- $d_4 = nw_{14} + nw_{24} + nw_{34}$

- $d_5 = nw_{25} + nw_{35}$
- $d_6 = nw_{26} + nw_{36}$
- $d_7 = nw_{37}$
- $np_{11} \geq 0$
- $np_{12} \geq 0$
- $np_{21} \geq 0$
- $np_{22} \geq 0$
- $np_{31} \geq 0$
- $np_{32} \geq 0$
- $np_{33} \geq 0$
- $np_{42} \geq 0$
- $np_{43} \geq 0$
- $nw_{11} \geq 0$
- $nw_{12} \geq 0$
- $nw_{13} \geq 0$
- $nw_{14} \geq 0$
- $nw_{23} \geq 0$
- $nw_{24} \geq 0$
- $nw_{25} \geq 0$
- $nw_{26} \geq 0$
- $nw_{34} \geq 0$
- $nw_{35} \geq 0$
- $nw_{36} \geq 0$
- $nw_{37} \geq 0$

5. Identify inputs and outputs that you can control

- $np_{ij}$
- $nw_{jk}$
- $cost$

6. Specify all quantities mathematically

- Many have been defined above already. A few more will be added here.
- $cost = \sum_{i=1}^{n} \sum_{j=1}^{q} np_{ij} * cp_{ij} + \sum_{j=1}^{q} \sum_{k=1}^{m} nw_{jk} * cw_{jk}$
- $cp_{11} = 10$
- $cp_{12} = 15$

- $cp_{21} = 11$
- $cp_{22} = 8$
- $cp_{31} = 13$
- $cp_{32} = 8$
- $cp_{33} = 9$
- $cp_{42} = 14$
- $cp_{43} = 8$
- $cw_{11} = 5$
- $cw_{12} = 6$
- $cw_{13} = 7$
- $cw_{14} = 10$
- $cw_{23} = 12$
- $cw_{24} = 8$
- $cw_{25} = 10$
- $cw_{26} = 14$
- $cw_{34} = 14$
- $cw_{35} = 12$
- $cw_{36} = 12$
- $cw_{37} = 6$

7. Check the model for completeness and correctness

- All variables are positive.

## Matlab Code

```matlab
% -----------------------------------------------------------
% reference: array index to variable mapping
% -----------------------------------------------------------
% (1) np11
% (2) np12
% (3) np21
% (4) np22
% (5) np31
% (6) np32
% (7) np33
% (8) np42
% (9) np43
% (10) nw11
% (11) nw12
% (12) nw13
```

```matlab
16  % (13) nw14
17  % (14) nw23
18  % (15) nw24
19  % (16) nw25
20  % (17) nw26
21  % (18) nw34
22  % (19) nw35
23  % (20) nw36
24  % (21) nw37
25  % (22) s1
26  % (23) s2
27  % (24) s3
28  % (25) s4
29  % (26) d1
30  % (27) d2
31  % (28) d3
32  % (29) d4
33  % (30) d5
34  % (31) d6
35  % (32) d7
36
37  % ------------------------------------------------------------
38  % lower bounds vector
39  %    note matlab arrays/vectors start at index 1 (not 0)
40  % ------------------------------------------------------------
41  lb = zeros(32,1);
42  lb(26) = 100;    % d1
43  lb(27) = 150;    % d2
44  lb(28) = 100;    % d3
45  lb(29) = 200;    % d4
46  lb(30) = 200;    % d5
47  lb(31) = 150;    % d6
48  lb(32) = 100;    % d7
49
50  % ------------------------------------------------------------
51  % upper bounds vector
52  %    note matlab arrays/vectors start at index 1 (not 0)
53  % ------------------------------------------------------------
54  ub = Inf(32,1);
55  ub(1)  = 150;    % np11
56  ub(2)  = 150;    % np12
57  ub(3)  = 450;    % np21
58  ub(4)  = 450;    % np22
59  ub(5)  = 250;    % np31
60  ub(6)  = 250;    % np32
61  ub(7)  = 250;    % np33
62  ub(8)  = 150;    % np42
63  ub(9)  = 150;    % np43
64  ub(22) = 150;    % s1
65  ub(23) = 450;    % s2
66  ub(24) = 250;    % s3
67  ub(25) = 150;    % s4
68
69  % ------------------------------------------------------------
```

```matlab
70  % linear inequality matrix and vector
71  %    note matlab arrays/vectors start at index 1 (not 0)
72  % ------------------------------------------------------------
73  A = [];
74  b = [];
75
76  % ------------------------------------------------------------
77  % linear equality matrix and vector
78  %    note matlab arrays/vectors start at index 1 (not 0)
79  %    14 equations in 32 variables
80  % ------------------------------------------------------------
81  Aeq = zeros(14, 32);
82  beq = zeros(14, 1);
83  %np11 + np21 + np31          = nw11 + nw12 + nw13 + nw14
84  %np11 + np21 + np31 - nw11 - nw12 - nw13 - nw14 = 0
85  Aeq(1,[1,3,5,10,11,12,13]) = [1,1,1,-1,-1,-1,-1];
86  %np12 + np22 + np32 + np42 =              nw23 + nw24 + nw25 + nw26
87  %np12 + np22 + np32 + np42 - nw23 - nw24 - nw25 - nw26 = 0
88  Aeq(2,[2,4,6,8,14,15,16,17]) = [1,1,1,1,-1,-1,-1,-1];
89  %              np33 + np43 =                    nw34 + nw35 + nw36 + nw37
90  %np33 + np43 - nw34 - nw35 - nw36 - nw37 = 0
91  Aeq(3,[7,9,18,19,20,21]) = [1,1,-1,-1,-1,-1];
92  %s1 = np11 + np12
93  %s1 - np11 - np12 = 0
94  Aeq(4,[22,1,2]) = [1,-1,-1];
95  %s2 = np21 + np22
96  %s2 - np21 - np22 = 0
97  Aeq(5,[23,3,4]) = [1,-1,-1];
98  %s3 = np31 + np32 + np33
99  %s3 - np31 - np32 - np33 = 0
100 Aeq(6,[24,5,6,7]) = [1,-1,-1,-1];
101 %s4 =         np42 + np43
102 %s4 - np42 - np43 = 0
103 Aeq(7,[25,8,9]) = [1,-1,-1];
104 %d1 = nw11
105 %d1 - nw11 = 0
106 Aeq(8,[26,10]) = [1,-1];
107 %d2 = nw12
108 %d2 - nw12 = 0
109 Aeq(9,[27,11]) = [1,-1];
110 %d3 = nw13 + nw23
111 %d3 - nw13 - nw23 = 0
112 Aeq(10,[28,12,14]) = [1,-1,-1];
113 %d4 = nw14 + nw24 + nw34
114 %d4 - nw14 - nw24 - nw34 = 0
115 Aeq(11,[29,13,15,18]) = [1,-1,-1,-1];
116 %d5 =         nw25 + nw35
117 %d5 - nw25 - nw35 = 0
118 Aeq(12,[30,16,19]) = [1,-1,-1];
119 %d6 =         nw26 + nw36
120 %d6 - nw26 - nw36 = 0
121 Aeq(13,[31,17,20]) = [1,-1,-1];
122 %d7 =                    nw37
123 %d7 - nw37 = 0
```

```matlab
124  Aeq(14,[32,21]) = [1,-1];
125
126  % -------------------------------------------------------------
127  % objective function vector
128  %   note matlab arrays/vectors start at index 1 (not 0)
129  % -------------------------------------------------------------
130  f = zeros(32,1);
131  f(1)  = 10;    % np11(value in f is cp11)
132  f(2)  = 15;    % np12(value in f is cp12)
133  f(3)  = 11;    % np21(value in f is cp21)
134  f(4)  = 8;     % np22(value in f is cp22)
135  f(5)  = 13;    % np31(value in f is cp31)
136  f(6)  = 8;     % np32(value in f is cp32)
137  f(7)  = 9;     % np33(value in f is cp33)
138  f(8)  = 14;    % np42(value in f is cp42)
139  f(9)  = 8;     % np43(value in f is cp43)
140  f(10) = 5;     % nw11(value in f is cw11)
141  f(11) = 6;     % nw12(value in f is cw12)
142  f(12) = 7;     % nw13(value in f is cw13)
143  f(13) = 10;    % nw14(value in f is cw14)
144  f(14) = 12;    % nw23(value in f is cw23)
145  f(15) = 8;     % nw24(value in f is cw24)
146  f(16) = 10;    % nw25(value in f is cw25)
147  f(17) = 14;    % nw26(value in f is cw26)
148  f(18) = 14;    % nw34(value in f is cw34)
149  f(19) = 12;    % nw35(value in f is cw35)
150  f(20) = 12;    % nw36(value in f is cw36)
151  f(21) = 6;     % nw37(value in f is cw37)
152
153  % -------------------------------------------------------------
154  % call solver and obtain solution
155  % -------------------------------------------------------------
156  [x fval] = linprog(f,A,b,Aeq,beq,lb,ub);
157
158  % -------------------------------------------------------------
159  % print the optimum shipping routes and min cost
160  % -------------------------------------------------------------
161  fileID = fopen('partA.out','w');
162  fprintf(fileID, '-----------------------------------------------------\n');
163  fprintf(fileID, 'Project 3 Problem 1 Part A Solution\n');
164  fprintf(fileID, '-----------------------------------------------------\n');
165  fprintf(fileID, '\n');
166  fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
167      'at a cost of $%4.0f.\n'], x(1), 1, 1, x(1) * f(1));
168  fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
169      'at a cost of $%4.0f.\n'], x(2), 1, 2, x(2) * f(2));
170  fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
171      'at a cost of $%4.0f.\n'], x(3), 2, 1, x(3) * f(3));
172  fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
173      'at a cost of $%4.0f.\n'], x(4), 2, 2, x(4) * f(4));
174  fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
175      'at a cost of $%4.0f.\n'], x(5), 3, 1, x(5) * f(5));
176  fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
177      'at a cost of $%4.0f.\n'], x(6), 3, 2, x(6) * f(6));
```

9

```matlab
178 fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
179     'at a cost of $%4.0f.\n'], x(7), 3, 3, x(7) * f(7));
180 fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
181     'at a cost of $%4.0f.\n'], x(8), 4, 2, x(8) * f(8));
182 fprintf(fileID, ['Ship %3.0f units from plant #%d to warehouse #%d ', ...
183     'at a cost of $%4.0f.\n'], x(9), 4, 3, x(9) * f(9));
184 fprintf(fileID, '\n');
185 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
186     '#%d at a cost of $%4.0f.\n'], x(10), 1, 1, x(10) * f(10));
187 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
188     '#%d at a cost of $%4.0f.\n'], x(11), 1, 2, x(11) * f(11));
189 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
190     '#%d at a cost of $%4.0f.\n'], x(12), 1, 3, x(12) * f(12));
191 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
192     '#%d at a cost of $%4.0f.\n'], x(13), 1, 4, x(13) * f(13));
193 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
194     '#%d at a cost of $%4.0f.\n'], x(14), 2, 3, x(14) * f(14));
195 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
196     '#%d at a cost of $%4.0f.\n'], x(15), 2, 4, x(15) * f(15));
197 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
198     '#%d at a cost of $%4.0f.\n'], x(16), 2, 5, x(16) * f(16));
199 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
200     '#%d at a cost of $%4.0f.\n'], x(17), 2, 6, x(17) * f(17));
201 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
202     '#%d at a cost of $%4.0f.\n'], x(18), 3, 4, x(18) * f(18));
203 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
204     '#%d at a cost of $%4.0f.\n'], x(19), 3, 5, x(19) * f(19));
205 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
206     '#%d at a cost of $%4.0f.\n'], x(20), 3, 6, x(20) * f(20));
207 fprintf(fileID, ['Ship %3.0f units from warehouse #%d to retailer ', ...
208     '#%d at a cost of $%4.0f.\n'], x(21), 3, 7, x(21) * f(21));
209 fprintf(fileID, '\n');
210 fprintf(fileID, ['%3.0f total units will leave plant #%d (capacity ', ...
211     'is %3.0f).\n'], x(1) + x(2), 1, x(22));
212 fprintf(fileID, ['%3.0f total units will leave plant #%d (capacity ', ...
213     'is %3.0f).\n'], x(3) + x(4), 2, x(23));
214 fprintf(fileID, ['%3.0f total units will leave plant #%d (capacity ', ...
215     'is %3.0f).\n'], x(5) + x(6) + x(7), 3, x(24));
216 fprintf(fileID, ['%3.0f total units will leave plant #%d (capacity ', ...
217     'is %3.0f).\n'], x(8) + x(9), 4, x(25));
218 fprintf(fileID, '\n');
219 temp = x(10) + x(11) + x(12) + x(13);
220 fprintf(fileID, ['%3.0f total units will enter warehouse #%d, %3.0f ', ...
221     'units will leave.\n'], x(1) + x(3) + x(5), 1, temp);
222 temp = x(14) + x(15) + x(16) + x(17);
223 fprintf(fileID, ['%3.0f total units will enter warehouse #%d, %3.0f ', ...
224     'units will leave.\n'], x(2) + x(4) + x(6) + x(8), 2, temp);
225 temp = x(18) + x(19) + x(20) + x(21);
226 fprintf(fileID, ['%3.0f total units will enter warehouse #%d, %3.0f ', ...
227     'units will leave.\n'], x(7) + x(9), 3, temp);
228 fprintf(fileID, '\n');
229 fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
230     'is %3.0f).\n'], x(10), 1, x(26));
231 fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
```

```matlab
232        'is %3.0f).\n'], x(11), 2, x(27));
233    fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
234        'is %3.0f).\n'], x(12) + x(14), 3, x(28));
235    fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
236        'is %3.0f).\n'], x(13) + x(15) + x(18), 4, x(29));
237    fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
238        'is %3.0f).\n'], x(16) + x(19), 5, x(30));
239    fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
240        'is %3.0f).\n'], x(17) + x(20), 6, x(31));
241    fprintf(fileID, ['%3.0f total units will enter retailer #%d (demand ', ...
242        'is %3.0f).\n'], x(21), 7, x(32));
243    fprintf(fileID, '\n');
244    total = x(22) + x(23) + x(24) + x(25);
245    fprintf(fileID, ['In all %3.0f units will travel through the network ', ...
246        'at a minimum cost of $%5.0f.\n'], total, fval);
247    fclose(fileID);
```

## Problem 1 Part B

Determine the number of refrigerators to be shipped from plants to warehouses, and then warehouses to retailers to minimize the cost. For part B warehouse 2 is closed along with all associated routes. Changes to the problem statement or solution relative to Problem 1 Part A are highlighted in red.

## Solution

There is no solution when warehouse 2 is closed. The following is the error message that is returned by Matlab function linprog():

Exiting: One or more of the residuals, duality gap, or total relative error has grown 100000 times greater than its minimum value so far:
    the primal appears to be infeasible (and the dual unbounded).
    (The dual residual < TolFun=1.00e-08.)

So Matlab tells us that there is no feasible solution. Why is that? If you look back at the network diagram and the supply and demand tables, you'll note that with warehouse 2 out of commission, retailers 5, 6, and 7 can only receive shipments from warehouse 3 and their total demand is 450 units. At the same time, warehouse 3 can only receive shipments from plant 3 and 4. The total supply capacity of those plants is only 400 units. Therefore, warehouse 3 gets 50 less units from plants 3 and 4 than are demanded from retailers 5, 6, and 7.

## Linear Program Formulation

1. Overall idea of problem

    - Refrigerators moving from $n = 4$ plants to $q = 2$ warehouses to $m = 7$ retailers.
    - Not all plants deliver to all warehouses.
    - Not all warehouses deliver to all retailers.
    - Costs of shipping from plants to warehouses vary by pair.
    - Costs of shipping from warehouses to retailers vary by pair.
    - Each plant has a capacity in terms of number of refrigerators it can supply.
    - Each retailer has a capacity in terms of number of refrigerators it demands.
    - Warehouse 2 has closed and all associate routes have been eliminated.

2. What is the goal? What are you trying to achieve?

    - Unchanged from part A.

3. Identify variables

    - Unchanged from part A.

4. Identify constraints

   - All constraints from part A remain in effect with the addition of two new constraints:

   - $np_{12} + np_{22} + np_{32} + np_{42} = 0$

   - $nw_{23} + nw_{24} + nw_{25} + nw_{26} = 0$

5. Identify inputs and outputs that you can control

   - Unchanged from part A.

6. Specify all quantities mathematically

   - Unchanged from part A.

7. Check the model for completeness and correctness

   - All variables are positive.

## Matlab Code

Code minimally changed from part A. Only changes are 2 additional constraints (16 total equations) in the linear equality matrix and vector. Identical code from part A is not shown below (to save space).

```
1  % ---------------------------------------------------------------
2  % linear equality matrix and vector
3  %    note matlab arrays/vectors start at index 1 (not 0)
4  %    16 equations in 32 variables
5  % ---------------------------------------------------------------
6  Aeq = zeros(16, 32);
7  beq = zeros(16, 1);
8  %np11 + np21 + np31           = nw11 + nw12 + nw13 + nw14
9  %np11 + np21 + np31 - nw11 - nw12 - nw13 - nw14 = 0
10 Aeq(1,[1,3,5,10,11,12,13]) = [1,1,1,-1,-1,-1,-1];
11 %np12 + np22 + np32 + np42 =            nw23 + nw24 + nw25 + nw26
12 %np12 + np22 + np32 + np42 - nw23 - nw24 - nw25 - nw26 = 0
13 Aeq(2,[2,4,6,8,14,15,16,17]) = [1,1,1,1,-1,-1,-1,-1];
14 %               np33 + np43 =                nw34 + nw35 + nw36 + nw37
15 %np33 + np43 - nw34 - nw35 - nw36 - nw37 = 0
16 Aeq(3,[7,9,18,19,20,21]) = [1,1,-1,-1,-1,-1];
17 %s1 = np11 + np12
18 %s1 - np11 - np12 = 0
19 Aeq(4,[22,1,2]) = [1,-1,-1];
20 %s2 = np21 + np22
21 %s2 - np21 - np22 = 0
22 Aeq(5,[23,3,4]) = [1,-1,-1];
23 %s3 = np31 + np32 + np33
24 %s3 - np31 - np32 - np33 = 0
25 Aeq(6,[24,5,6,7]) = [1,-1,-1,-1];
26 %s4 =           np42 + np43
27 %s4 - np42 - np43 = 0
```

13

```
28  Aeq(7,[25,8,9]) = [1,-1,-1];
29  %d1 = nw11
30  %d1 - nw11 = 0
31  Aeq(8,[26,10]) = [1,-1];
32  %d2 = nw12
33  %d2 - nw12 = 0
34  Aeq(9,[27,11]) = [1,-1];
35  %d3 = nw13 + nw23
36  %d3 - nw13 - nw23 = 0
37  Aeq(10,[28,12,14]) = [1,-1,-1];
38  %d4 = nw14 + nw24 + nw34
39  %d4 - nw14 - nw24 - nw34 = 0
40  Aeq(11,[29,13,15,18]) = [1,-1,-1,-1];
41  %d5 =          nw25 + nw35
42  %d5 - nw25 - nw35 = 0
43  Aeq(12,[30,16,19]) = [1,-1,-1];
44  %d6 =          nw26 + nw36
45  %d6 - nw26 - nw36 = 0
46  Aeq(13,[31,17,20]) = [1,-1,-1];
47  %d7 =                          nw37
48  %d7 - nw37 = 0
49  Aeq(14,[32,21]) = [1,-1];
50  %np12 + np22 + np32 + np42 = 0
51  Aeq(15,[2,4,6,8]) = [1,1,1,1];
52  %nw23 + nw24 + nw25 + nw26 = 0
53  Aeq(16,[14,15,16,17]) = [1,1,1,1];
```

## Problem 1 Part C

Determine the number of refrigerators to be shipped from plants to warehouses, and then warehouses to retailers to minimize the cost. For part C warehouse 2 is limited to just 100 units per week. Changes to the problem statement or solution relative to Problem 1 Part A are highlighted in blue.

## Solution

- In all 1000 units will travel through the network at a minimum cost of $18300.

- Ship 150 units from plant #1 to warehouse #1 at a cost of $1500.

- Ship 350 units from plant #2 to warehouse #1 at a cost of $3850.

- Ship 100 units from plant #2 to warehouse #2 at a cost of $800.

- Ship 0 units from plant #3 to warehouse #2 at a cost of $ 0.

- Ship 250 units from plant #3 to warehouse #3 at a cost of $2250.

- Ship 150 units from plant #4 to warehouse #3 at a cost of $1200.

- Ship 100 units from warehouse #1 to retailer #1 at a cost of $ 500.

- Ship 150 units from warehouse #1 to retailer #2 at a cost of $ 900.

- Ship 100 units from warehouse #1 to retailer #3 at a cost of $ 700.

- Ship 150 units from warehouse #1 to retailer #4 at a cost of $1500.

- Ship 50 units from warehouse #2 to retailer #4 at a cost of $ 400.

- Ship 50 units from warehouse #2 to retailer #5 at a cost of $ 500.

- Ship 150 units from warehouse #3 to retailer #5 at a cost of $1800.

- Ship 150 units from warehouse #3 to retailer #6 at a cost of $1800.

- Ship 100 units from warehouse #3 to retailer #7 at a cost of $ 600.

- 150 total units will leave plant #1 (capacity is 150).

- 450 total units will leave plant #2 (capacity is 450).

- 250 total units will leave plant #3 (capacity is 250).

- 150 total units will leave plant #4 (capacity is 150).

- 500 total units will enter warehouse #1, 500 units will leave.

- 100 total units will enter warehouse #2, 100 units will leave.

- 400 total units will enter warehouse #3, 400 units will leave.

- 100 total units will enter retailer #1 (demand is 100).

- 150 total units will enter retailer #2 (demand is 150).

- 100 total units will enter retailer #3 (demand is 100).

- 200 total units will enter retailer #4 (demand is 200).

- 200 total units will enter retailer #5 (demand is 200).

- 150 total units will enter retailer #6 (demand is 150).

- 100 total units will enter retailer #7 (demand is 100).

## Linear Program Formulation

1. Overall idea of problem

   - Refrigerators moving from $n = 4$ plants to $q = 3$ warehouses to $m = 7$ retailers.
   - Not all plants deliver to all warehouses.
   - Not all warehouses deliver to all retailers.
   - Costs of shipping from plants to warehouses vary by pair.
   - Costs of shipping from warehouses to retailers vary by pair.
   - Each plant has a capacity in terms of number of refrigerators it can supply.
   - Each retailer has a capacity in terms of number of refrigerators it demands.
   - Warehouse 2 is limited to just 100 units in and out per week.

2. What is the goal? What are you trying to achieve?

   - Unchanged from part A.

3. Identify variables

   - Unchanged from part A.

4. Identify constraints

   - All constraints from part A remain in effect with the addition of two new constraints:
   - $np_{12} + np_{22} + np_{32} + np_{42} = 100$
   - $nw_{23} + nw_{24} + nw_{25} + nw_{26} = 100$

5. Identify inputs and outputs that you can control

   - Unchanged from part A.

6. Specify all quantities mathematically

   - Unchanged from part A.

7. Check the model for completeness and correctness

   - All variables are positive.

## Matlab Code

Code minimally changed from part A. Only changes are 2 additional constraints (16 total equations) in the linear equality matrix and vector. Identical code from part A is not shown below (to save space).

```matlab
% -------------------------------------------------------------
% linear equality matrix and vector
%    note matlab arrays/vectors start at index 1 (not 0)
%    16 equations in 32 variables
% -------------------------------------------------------------
Aeq = zeros(16, 32);
beq = zeros(16, 1);
%np11 + np21 + np31          = nw11 + nw12 + nw13 + nw14
%np11 + np21 + np31 - nw11 - nw12 - nw13 - nw14 = 0
Aeq(1,[1,3,5,10,11,12,13]) = [1,1,1,-1,-1,-1,-1];
%np12 + np22 + np32 + np42 =          nw23 + nw24 + nw25 + nw26
%np12 + np22 + np32 + np42 - nw23 - nw24 - nw25 - nw26 = 0
Aeq(2,[2,4,6,8,14,15,16,17]) = [1,1,1,1,-1,-1,-1,-1];
%           np33 + np43 =                    nw34 + nw35 + nw36 + nw37
%np33 + np43 - nw34 - nw35 - nw36 - nw37 = 0
Aeq(3,[7,9,18,19,20,21]) = [1,1,-1,-1,-1,-1];
%s1 = np11 + np12
%s1 - np11 - np12 = 0
Aeq(4,[22,1,2]) = [1,-1,-1];
%s2 = np21 + np22
%s2 - np21 - np22 = 0
Aeq(5,[23,3,4]) = [1,-1,-1];
%s3 = np31 + np32 + np33
%s3 - np31 - np32 - np33 = 0
Aeq(6,[24,5,6,7]) = [1,-1,-1,-1];
%s4 =          np42 + np43
%s4 - np42 - np43 = 0
Aeq(7,[25,8,9]) = [1,-1,-1];
%d1 = nw11
%d1 - nw11 = 0
Aeq(8,[26,10]) = [1,-1];
%d2 = nw12
%d2 - nw12 = 0
Aeq(9,[27,11]) = [1,-1];
```

```
35  %d3 = nw13 + nw23
36  %d3 - nw13 - nw23 = 0
37  Aeq(10,[28,12,14]) = [1,-1,-1];
38  %d4 = nw14 + nw24 + nw34
39  %d4 - nw14 - nw24 - nw34 = 0
40  Aeq(11,[29,13,15,18]) = [1,-1,-1,-1];
41  %d5 =          nw25 + nw35
42  %d5 - nw25 - nw35 = 0
43  Aeq(12,[30,16,19]) = [1,-1,-1];
44  %d6 =          nw26 + nw36
45  %d6 - nw26 - nw36 = 0
46  Aeq(13,[31,17,20]) = [1,-1,-1];
47  %d7 =                      nw37
48  %d7 - nw37 = 0
49  Aeq(14,[32,21]) = [1,-1];
50  %np12 + np22 + np32 + np42 = 100
51  Aeq(15,[2,4,6,8]) = [1,1,1,1];
52  beq(15) = 100;
53  %nw23 + nw24 + nw25 + nw26 = 100
54  Aeq(16,[14,15,16,17]) = [1,1,1,1];
55  beq(16) = 100;
```

## Problem 1 Part D

Formulate a generalized linear programming model for the transshipment problem. Give the objective function and constraints as mathematical formulas.

## Note

I can't help but feel that I've already done this in Parts A through C but I'll repeat (and try to expand) here in a more general form.

## Objective Function

Whereas...

$n$ = number of plants $\geq 1$
$q$ = number of warehouses $\geq 1$
$m$ = number of retailers $\geq 1$
$np_{ij}$ = number of refrigerators shipped from plant $p_i$ to warehouse $w_j \geq 0$
$cp_{ij}$ = cost of moving a refrigerator between plant $p_i$ and warehouse $w_j > 0$
$nw_{jk}$ = number of refrigerators shipped from warehouse $w_j$ to retailer $r_k \geq 0$
$cw_{jk}$ = cost of moving a refrigerator between warehouse $w_j$ and retailer $r_k > 0$

The **objective function** is to...

minimize $cost = \sum_{i=1}^{n} \sum_{j=1}^{q} np_{ij} * cp_{ij} + \sum_{j=1}^{q} \sum_{k=1}^{m} nw_{jk} * cw_{jk}$

## Constraints

The objective function (designed to minimize $cost$) is **subject to** the following contraints...

$0 \leq$ capacity of each plant $= s_i \leq$ some maximum
$0 \leq$ throughput capability of each warehouse $= q_j \leq$ some maximum
$0 \leq$ demand of each retailer $= m_k \leq$ some maximum

and also...

$\sum_{i=1}^{n} np_{ij}$ = capacity of a given plant $p_i$

$\sum_{i=1}^{n} np_{ij} = \sum_{k=1}^{m} nw_{jk}$ = throughput capability for a given warehouse $w_j$

$\sum_{k=1}^{m} nw_{jk}$ = demand of a given retailer $r_k$

# Problem 2

## Problem 2 Part A

Determine the combination of ingredients that minimizes calories but meets all nutritional requirements.

i. Formulate the problem as a linear program with an objective function and all constraints.

We can formulate this problem as the following linear program:

$$\text{minimize} \quad 21 \cdot w_{\text{tomato}} + 16 \cdot w_{\text{lettuce}} + 40 \cdot w_{\text{spinach}} + 41 \cdot w_{\text{carrot}}$$
$$+ 585 \cdot w_{\text{sunflower seed}} + 120 \cdot w_{\text{smoked tofu}} + 164 \cdot w_{\text{chickpea}} + 884 \cdot w_{\text{oil}}$$

$$\text{subject to} \quad -0.85 \cdot w_{\text{tomato}} - 1.62 \cdot w_{\text{lettuce}} - 2.86 \cdot w_{\text{spinach}} - 0.93 \cdot w_{\text{carrot}}$$
$$- 23.4 \cdot w_{\text{sunflower seed}} - 16 \cdot w_{\text{smoked tofu}} - 9 \cdot w_{\text{chickpea}} - 0 \cdot w_{\text{oil}} \leq -15,$$

$$-0.33 \cdot w_{\text{tomato}} - 0.2 \cdot w_{\text{lettuce}} - 0.39 \cdot w_{\text{spinach}} - 0.24 \cdot w_{\text{carrot}}$$
$$- 48.7 \cdot w_{\text{sunflower seed}} - 5 \cdot w_{\text{smoked tofu}} - 2.6 \cdot w_{\text{chickpea}} - 100 \cdot w_{\text{oil}} \leq -2,$$

$$0.33 \cdot w_{\text{tomato}} + 0.2 \cdot w_{\text{lettuce}} + 0.39 \cdot w_{\text{spinach}} + 0.24 \cdot w_{\text{carrot}}$$
$$+ 48.7 \cdot w_{\text{sunflower seed}} + 5 \cdot w_{\text{smoked tofu}} + 2.6 \cdot w_{\text{chickpea}} + 100 \cdot w_{\text{oil}} \leq 8,$$

$$-4.64 \cdot w_{\text{tomato}} - 2.37 \cdot w_{\text{lettuce}} - 3.63 \cdot w_{\text{spinach}} - 9.58 \cdot w_{\text{carrot}}$$
$$- 15 \cdot w_{\text{sunflower seed}} - 3 \cdot w_{\text{smoked tofu}} - 27 \cdot w_{\text{chickpea}} - 0 \cdot w_{\text{oil}} \leq -4,$$

$$9 \cdot w_{\text{tomato}} + 28 \cdot w_{\text{lettuce}} + 65 \cdot w_{\text{spinach}} + 69 \cdot w_{\text{carrot}}$$
$$+ 3.8 \cdot w_{\text{sunflower seed}} + 120 \cdot w_{\text{smoked tofu}} + 78 \cdot w_{\text{chickpea}} + 0 \cdot w_{\text{oil}} \leq 200,$$

$$0.4 \cdot w_{\text{tomato}} - 0.6 \cdot w_{\text{lettuce}} - 0.6 \cdot w_{\text{spinach}} + 0.4 \cdot w_{\text{carrot}}$$
$$+ 0.4 \cdot w_{\text{sunflower seed}} + 0.4 \cdot w_{\text{smoked tofu}} + 0.4 \cdot w_{\text{chickpea}} + 0.4 \cdot w_{\text{oil}} \leq 0,$$

$$-w_{\text{tomato}} \leq 0$$
$$-w_{\text{lettuce}} \leq 0$$
$$-w_{\text{spinach}} \leq 0$$
$$-w_{\text{carrot}} \leq 0$$
$$-w_{\text{sunflower seed}} \leq 0$$
$$-w_{\text{smoked tofu}} \leq 0$$
$$-w_{\text{chickpea}} \leq 0$$
$$-w_{\text{oil}} \leq 0$$

where the $w$ parameters are the weight of each ingredient, in 100's of grams. The equations, from top to bottom, are:

1) Minimize total calories

2) Subject to total protein $\geq 15$ grams

3) Subject to total fat $\geq 2$ grams

4) Subject to total fat $\leq 8$ grams

5) Subject to total carbohydrates $\geq 4$ grams

6) Subject to total sodium $\leq 200$ milligrams

7) Subject to total leafy green mass $\geq 40\%$ of total mass

8) Subject to individual ingredient weights $\geq 0$

ii. Determine the optimal solution for the linear program using any software you want. Include a copy of the code/file in the report.

The following MATLAB code was used to generate the solution:

```
1  f = [21, 16, 40, 41, 585, 120, 164, 884];
2
3  A = [-0.85, -1.62, -2.86, -0.93, -23.4, -16, -9, 0;
4      -0.33, -0.2, -0.39, -0.24, -48.7, -5, -2.6, -100;
5      0.33, 0.2, 0.39, 0.24, 48.7, 5, 2.6, 100;
6      -4.64, -2.37, -3.63, -9.58, -15, -3, -27, 0;
7      9, 28, 65, 69, 3.8, 120, 78, 0;
8      0.4, -0.6, -0.6, 0.4, 0.4, 0.4, 0.4, 0.4;
9      -eye(8)];
10
11  b = [-15; -2; 8; -4; 200;  0; zeros(8,1)];
12
13  [X,FVAL,EXITFLAG] = linprog(f,A,b);
```

where X stores the resulting weights of ingredients (in 100's of grams), FVAL stores the minimized number of calories, and EXITFLAG stores the status of the linprog optimization.

iii. What is the cost of the low calorie salad?

The optimal low calorie salad contains the following weights of ingredients (in 100's of grams):

$$w_{\text{tomato}} \approx 0$$
$$w_{\text{lettuce}} \approx 0.5855$$
$$w_{\text{spinach}} \approx 0$$
$$w_{\text{carrot}} \approx 0$$
$$w_{\text{sunflower seed}} \approx 0$$
$$w_{\text{smoked tofu}} \approx 0.8782$$
$$w_{\text{chickpea}} \approx 0$$
$$w_{\text{oil}} \approx 0$$

The optimal low calorie salad costs approximately \$2.33. It contains approximately 114.75 kcal

## Problem 2 Part B

Veronica realizes that it is also important to minimize the cost associated with the new salad. Unfortunately some of the ingredients can be expensive. Determine the combination of ingredients that minimizes cost.

i. Formulate the problem as a linear program with an objective function and all constraints.

We can formulate this problem as the following linear program:

$$
\begin{aligned}
\text{minimize} \quad & 1 \cdot w_{\text{tomato}} + 0.75 \cdot w_{\text{lettuce}} + 0.5 \cdot w_{\text{spinach}} + 0.5 \cdot w_{\text{carrot}} \\
& + 0.45 \cdot w_{\text{sunflower seed}} + 2.15 \cdot w_{\text{smoked tofu}} + 0.95 \cdot w_{\text{chickpea}} + 2.00 \cdot w_{\text{oil}}
\end{aligned}
$$

$$
\begin{aligned}
\text{subject to} \quad & -0.85 \cdot w_{\text{tomato}} - 1.62 \cdot w_{\text{lettuce}} - 2.86 \cdot w_{\text{spinach}} - 0.93 \cdot w_{\text{carrot}} \\
& - 23.4 \cdot w_{\text{sunflower seed}} - 16 \cdot w_{\text{smoked tofu}} - 9 \cdot w_{\text{chickpea}} - 0 \cdot w_{\text{oil}} \leq -15,
\end{aligned}
$$

$$
\begin{aligned}
& -0.33 \cdot w_{\text{tomato}} - 0.2 \cdot w_{\text{lettuce}} - 0.39 \cdot w_{\text{spinach}} - 0.24 \cdot w_{\text{carrot}} \\
& - 48.7 \cdot w_{\text{sunflower seed}} - 5 \cdot w_{\text{smoked tofu}} - 2.6 \cdot w_{\text{chickpea}} - 100 \cdot w_{\text{oil}} \leq -2,
\end{aligned}
$$

$$
\begin{aligned}
& 0.33 \cdot w_{\text{tomato}} + 0.2 \cdot w_{\text{lettuce}} + 0.39 \cdot w_{\text{spinach}} + 0.24 \cdot w_{\text{carrot}} \\
& + 48.7 \cdot w_{\text{sunflower seed}} + 5 \cdot w_{\text{smoked tofu}} + 2.6 \cdot w_{\text{chickpea}} + 100 \cdot w_{\text{oil}} \leq 8,
\end{aligned}
$$

$$
\begin{aligned}
& -4.64 \cdot w_{\text{tomato}} - 2.37 \cdot w_{\text{lettuce}} - 3.63 \cdot w_{\text{spinach}} - 9.58 \cdot w_{\text{carrot}} \\
& - 15 \cdot w_{\text{sunflower seed}} - 3 \cdot w_{\text{smoked tofu}} - 27 \cdot w_{\text{chickpea}} - 0 \cdot w_{\text{oil}} \leq -4,
\end{aligned}
$$

$$
\begin{aligned}
& 9 \cdot w_{\text{tomato}} + 28 \cdot w_{\text{lettuce}} + 65 \cdot w_{\text{spinach}} + 69 \cdot w_{\text{carrot}} \\
& + 3.8 \cdot w_{\text{sunflower seed}} + 120 \cdot w_{\text{smoked tofu}} + 78 \cdot w_{\text{chickpea}} + 0 \cdot w_{\text{oil}} \leq 200,
\end{aligned}
$$

$$
\begin{aligned}
& 0.4 \cdot w_{\text{tomato}} - 0.6 \cdot w_{\text{lettuce}} - 0.6 \cdot w_{\text{spinach}} + 0.4 \cdot w_{\text{carrot}} \\
& + 0.4 \cdot w_{\text{sunflower seed}} + 0.4 \cdot w_{\text{smoked tofu}} + 0.4 \cdot w_{\text{chickpea}} + 0.4 \cdot w_{\text{oil}} \leq 0,
\end{aligned}
$$

$$
\begin{aligned}
& -w_{\text{tomato}} \leq 0 \\
& -w_{\text{lettuce}} \leq 0 \\
& -w_{\text{spinach}} \leq 0 \\
& -w_{\text{carrot}} \leq 0 \\
& -w_{\text{sunflower seed}} \leq 0 \\
& -w_{\text{smoked tofu}} \leq 0 \\
& -w_{\text{chickpea}} \leq 0 \\
& -w_{\text{oil}} \leq 0
\end{aligned}
$$

where the $w$ parameters are the weight of each ingredient, in 100's of grams. The equations, from top to bottom, are:

1) Minimize total cost

2) Subject to total protein $\geq$ 15 grams

3) Subject to total fat $\geq$ 2 grams

4) Subject to total fat $\leq$ 8 grams

5) Subject to total carbohydrates $\geq$ 4 grams

6) Subject to total sodium $\leq$ 200 milligrams

7) Subject to total leafy green mass $\geq$ 40% of total mass

8) Subject to individual ingredient weights $\geq$ 0

ii. Determine the optimal solution for the linear program using any software you want. Include a copy of the code/file in the report.

The following MATLAB code was used to generate the solution:

```
1  f = [1, 0.75, 0.5, 0.5, 0.45, 2.15, 0.95, 2.0];
2
3  A = [-0.85, -1.62, -2.86, -0.93, -23.4, -16, -9, 0;
4      -0.33, -0.2, -0.39, -0.24, -48.7, -5, -2.6, -100;
5      0.33, 0.2, 0.39, 0.24, 48.7, 5, 2.6, 100;
6      -4.64, -2.37, -3.63, -9.58, -15, -3, -27, 0;
7      9, 28, 65, 69, 3.8, 120, 78, 0;
8      0.4, -0.6, -0.6, 0.4, 0.4, 0.4, 0.4, 0.4;
9      -eye(8)];
10
11  b = [-15; -2; 8; -4; 200;  0; zeros(8,1)];
12
13  [X,FVAL,EXITFLAG] = linprog(f,A,b);
```

where X stores the resulting weights of ingredients (in 100's of grams), FVAL stores the minimized number of calories, and EXITFLAG stores the status of the linprog optimization.

iii. How many calories are in the low cost salad?

The optimal low cost salad contains the following weights of ingredients (in 100's of grams):

$$w_{\text{tomato}} \approx 0$$
$$w_{\text{lettuce}} \approx 0$$
$$w_{\text{spinach}} \approx 0.8323$$
$$w_{\text{carrot}} \approx 0$$
$$w_{\text{sunflower seed}} \approx 0.0961$$
$$w_{\text{smoked tofu}} \approx 0$$
$$w_{\text{chickpea}} \approx 1.1524$$
$$w_{\text{oil}} \approx 0$$

The optimal low cost salad costs approximately \$1.55. It contains approximately 278.49 kcal

## Problem 2 Part C

Compare the results from part A and B. Veronicas goal is to create a Very Veggie Salad that is both low calorie and low cost. She would like to sell the salad for $5.00 and still have a profit of at least $3.00. However if she can advertise that the salad has under 250 calories then she may be able to sell more.

i. Suggest some possible ways that she select a combination of ingredients that is near optimal for both objectives. This is a type of multi-objective optimization.

What Veronica should do is attempt to find a "pareto optimal" solution that accounts for both calories and cost. In order to do this, she should introduce a new parameter $\lambda$ to her optimization formulation. In essence, she would like to:

$$\begin{aligned} \text{minimize} \quad & (1 - \lambda) \cdot \text{CALORIES} + \lambda \cdot \text{PRICE} \\ \text{subject to} \quad & \text{CONSTRAINTS} \end{aligned}$$

for values of $\lambda$ between 0 and 1. When $\lambda = 0$, she will be finding the minimum calorie combination of ingredients. When $\lambda = 1$, she will be finding the minimum price combination of ingredients. For all values of $\lambda$ inbetween, she will be finding the "pareto optimal" combination of ingredients. So, she should solve the optimization problem for values of $\lambda$ between 0 and 1. She can then examine these pareto optimal combinations of ingredients to determine which would best meet her goals of $3.00 profit and under 250 calories.

We can express this as the following linear program:

$$\text{minimize} \quad (1 - \lambda) \cdot (21 \cdot w_{\text{tomato}} + 16 \cdot w_{\text{lettuce}} + 40 \cdot w_{\text{spinach}} + 41 \cdot w_{\text{carrot}}$$
$$+ 585 \cdot w_{\text{sunflower seed}} + 120 \cdot w_{\text{smoked tofu}} + 164 \cdot w_{\text{chickpea}} + 884 \cdot w_{\text{oil}})$$
$$+ \lambda \cdot (1 \cdot w_{\text{tomato}} + 0.75 \cdot w_{\text{lettuce}} + 0.5 \cdot w_{\text{spinach}} + 0.5 \cdot w_{\text{carrot}}$$
$$+ 0.45 \cdot w_{\text{sunflower seed}} + 2.15 \cdot w_{\text{smoked tofu}} + 0.95 \cdot w_{\text{chickpea}} + 2.00 \cdot w_{\text{oil}})$$

$$\text{subject to} \quad -0.85 \cdot w_{\text{tomato}} - 1.62 \cdot w_{\text{lettuce}} - 2.86 \cdot w_{\text{spinach}} - 0.93 \cdot w_{\text{carrot}}$$
$$- 23.4 \cdot w_{\text{sunflower seed}} - 16 \cdot w_{\text{smoked tofu}} - 9 \cdot w_{\text{chickpea}} - 0 \cdot w_{\text{oil}} \leq -15,$$

$$-0.33 \cdot w_{\text{tomato}} - 0.2 \cdot w_{\text{lettuce}} - 0.39 \cdot w_{\text{spinach}} - 0.24 \cdot w_{\text{carrot}}$$
$$- 48.7 \cdot w_{\text{sunflower seed}} - 5 \cdot w_{\text{smoked tofu}} - 2.6 \cdot w_{\text{chickpea}} - 100 \cdot w_{\text{oil}} \leq -2,$$

$$0.33 \cdot w_{\text{tomato}} + 0.2 \cdot w_{\text{lettuce}} + 0.39 \cdot w_{\text{spinach}} + 0.24 \cdot w_{\text{carrot}}$$
$$+ 48.7 \cdot w_{\text{sunflower seed}} + 5 \cdot w_{\text{smoked tofu}} + 2.6 \cdot w_{\text{chickpea}} + 100 \cdot w_{\text{oil}} \leq 8,$$

$$-4.64 \cdot w_{\text{tomato}} - 2.37 \cdot w_{\text{lettuce}} - 3.63 \cdot w_{\text{spinach}} - 9.58 \cdot w_{\text{carrot}}$$
$$- 15 \cdot w_{\text{sunflower seed}} - 3 \cdot w_{\text{smoked tofu}} - 27 \cdot w_{\text{chickpea}} - 0 \cdot w_{\text{oil}} \leq -4,$$

$$9 \cdot w_{\text{tomato}} + 28 \cdot w_{\text{lettuce}} + 65 \cdot w_{\text{spinach}} + 69 \cdot w_{\text{carrot}}$$
$$+ 3.8 \cdot w_{\text{sunflower seed}} + 120 \cdot w_{\text{smoked tofu}} + 78 \cdot w_{\text{chickpea}} + 0 \cdot w_{\text{oil}} \leq 200,$$

$$0.4 \cdot w_{\text{tomato}} - 0.6 \cdot w_{\text{lettuce}} - 0.6 \cdot w_{\text{spinach}} + 0.4 \cdot w_{\text{carrot}}$$
$$+ 0.4 \cdot w_{\text{sunflower seed}} + 0.4 \cdot w_{\text{smoked tofu}} + 0.4 \cdot w_{\text{chickpea}} + 0.4 \cdot w_{\text{oil}} \leq 0,$$

$$-w_{\text{tomato}} \leq 0$$
$$-w_{\text{lettuce}} \leq 0$$
$$-w_{\text{spinach}} \leq 0$$
$$-w_{\text{carrot}} \leq 0$$
$$-w_{\text{sunflower seed}} \leq 0$$
$$-w_{\text{smoked tofu}} \leq 0$$
$$-w_{\text{chickpea}} \leq 0$$
$$-w_{\text{oil}} \leq 0$$

ii. What combination of ingredient would you suggest and what is the associated cost and calorie.

Varying $\lambda$ between 0 and 1, and solving the resulting optimization problems, we obtain the following pareto optimal combinations of total calories and total price:

Cost vs. Calorie Optimization
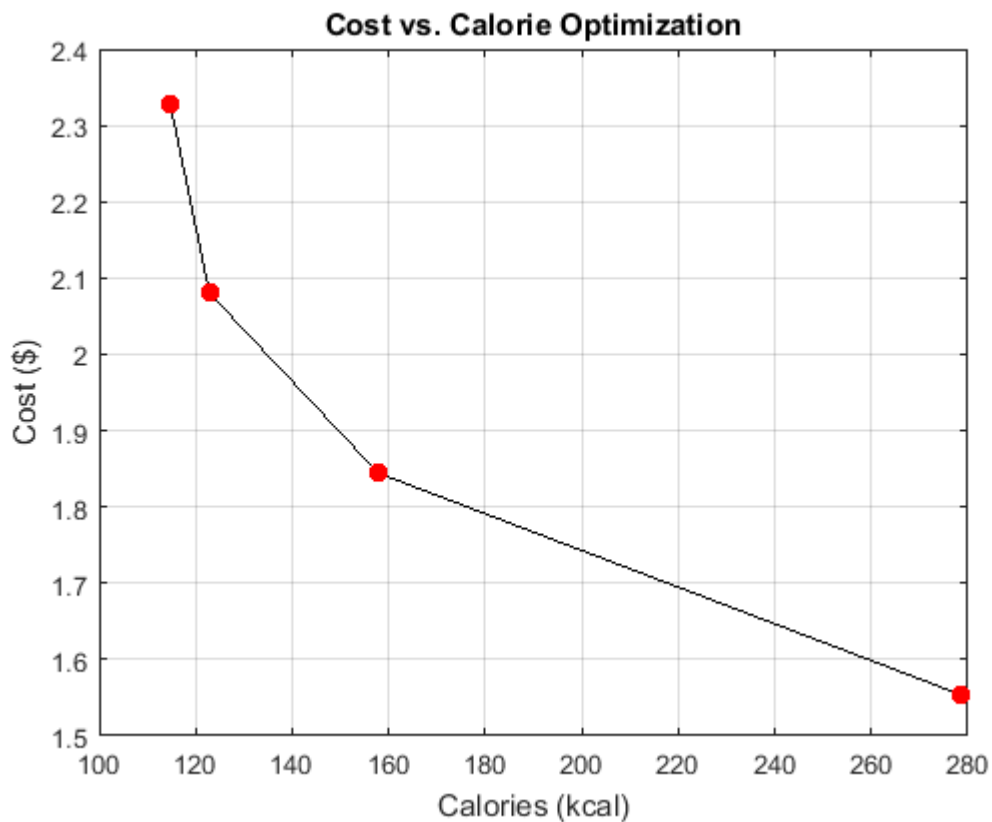
Based on this result, I would suggest the following combination of ingredients:

$$w_{\text{tomato}} \approx 0$$
$$w_{\text{lettuce}} \approx 0$$
$$w_{\text{spinach}} \approx 0.5346$$
$$w_{\text{carrot}} \approx 0$$
$$w_{\text{sunflower seed}} \approx 0.0865$$
$$w_{\text{smoked tofu}} \approx 0.7154$$
$$w_{\text{chickpea}} \approx 0$$
$$w_{\text{oil}} \approx 0$$

This pareto optimal salad costs approximately \$1.84. It contains approximately 157.86 kcal. This meets both of Veronicas goals.

iii. Note: There is not one right answer. Discuss how you derived your solution.

The following MATLAB code was used to derive the answer, based on the discussion above.

```
1  lambda = 0:0.0005:1;
2
```

```matlab
for j = 1:numel(lambda)
    L = lambda(j);
    f1 = [21, 16, 40, 41, 585, 120, 164, 884];
    f2 = [1, 0.75, 0.5, 0.5, 0.45, 2.15, 0.95, 2.0];

    f = (1 - L)*f1 + L*f2;

    A = [-0.85, -1.62, -2.86, -0.93, -23.4, -16, -9, 0;
        -0.33, -0.2, -0.39, -0.24, -48.7, -5, -2.6, -100;
        0.33, 0.2, 0.39, 0.24, 48.7, 5, 2.6, 100;
        -4.64, -2.37, -3.63, -9.58, -15, -3, -27, 0;
        9, 28, 65, 69, 3.8, 120, 78, 0;
        0.4, -0.6, -0.6, 0.4, 0.4, 0.4, 0.4, 0.4;
        -eye(8)];

    b = [-15; -2; 8; -4; 200;  0; zeros(8,1)];

    [X,FVAL,EXITFLAG] = linprog(f,A,b);

    calories_per_ingredient = [21, 16, 40, 41, 585, 120, 164, 884];
    cost_per_ingredient = [1, 0.75, 0.5, 0.5, 0.45, 2.15, 0.95, 2];

    if (EXITFLAG == 1)
        cost(j) = cost_per_ingredient * X;
        calories(j) = calories_per_ingredient * X;
    else
        cost(j) = NaN;
        calories(j) = NaN;
    end
end

figure(1)
plot(calories, cost, 'k-')
hold on
plot(calories, cost, 'r.', 'markersize', 25)
grid on
xlabel('Calories (kcal)')
ylabel('Cost ($)')
title('Cost vs. Calorie Optimization')
```

# Problem 3

a) What are the lengths of the shortest paths from vertex $a$ to all other vertices?

b) If a vertex $z$ is added to the graph for which there is no path from vertex $a$ to vertex $z$, what will be the result when you attempt to find the lengths of shortest paths as in part a).

c) What are the lengths of the shortest paths from each vertex to vertex $m$? How can you solve this problem with just one linear program?

d) Suppose that all paths must pass through vertex $i$. How can you calculate the length of the shortest path from any vertex $x$ to vertex $y$ that pass through vertex $i$ (for all $x, y \in V$)? Calculate the lengths of these paths for the given graph. (Note: for some vertices $x$ and $y$, it may be impossible to pass through vertex $i$).