

# MMA 662: Decision Analytics

## Assignment 1 on LP: The Story of an Imbalance!

"The balance must be maintained. That is why I have returned." -Thanos

```
In [6]: import gurobipy as gp
        from gurobipy import *
        import numpy as np
```

### Fresh Milk Transportation

*BalancedMilk* is an old dairy-transportation firm known for the freshness of the milk it distributes. It has long-term fixed-amount contracts with eight dairy farms (suppliers) and distributes their production among ten demand markets. Transporting milk for half a century from suppliers to demand markets, *BalancedMilk* is well aware of the milk demand in each demand market and the transportation cost from each supplier to each demand center.

In the following table, the number in front of each supply and demand center represents the supply and demand (in tonnes.) Also, for each supply-demand pair, the transportation cost (dollars per tonne) is depicted.

	$D_1$ (90)	$D_2$ (100)	$D_3$ (150)	$D_4$ (100)	$D_5$ (180)	$D_6$ (240)	$D_7$ (210)	$D_8$ (90)	$D_9$ (160)	$D_{10}$
$S_1$ (110)	20	49	16	30	8	35	21	40	10	12
$S_2$ (80)	15	53	7	20	47	11	16	17	15	44
$S_3$ (100)	22	25	42	22	31	9	11	29	20	5
$S_4$ (240)	45	6	33	35	49	25	30	47	32	31
$S_5$ (100)	9	12	41	15	38	14	53	22	12	13
$S_6$ (280)	21	24	32	49	5	47	30	23	37	8
$S_7$ (130)	12	19	5	28	47	39	15	35	9	51
$S_8$ (440)	34	17	10	21	9	33	14	26	19	45

For example, transporting one tonne of milk from supply center  $S_1$  to demand market  $D_1$  costs 20 dollars.

**Note:** In all of the following problems, the firm cannot supply milk to a demand market **more** than its demand, i.e., cannot **over-supply** it.

### Problem 1: (2.5 pts)

The firm wants to **distribute all the supplied milk as efficiently as possible**. We wrote a code that estimates the optimal milk distribution and the total transportation cost. Run the code and find the optimal milk distribution and the total transportation cost. Report the total transportation cost in dollars.

```
In [ ]: Cost = [[20, 49, 16, 30, 8, 35, 21, 40, 10, 12],
              [15, 53, 7, 20, 47, 11, 16, 17, 15, 44],
              [22, 25, 42, 22, 31, 9, 11, 29, 20, 5],
              [45, 6, 33, 35, 49, 25, 30, 47, 32, 31],
              [9, 12, 41, 15, 38, 14, 53, 22, 12, 13],
              [21, 24, 32, 49, 5, 47, 30, 23, 37, 8],
              [12, 19, 5, 28, 47, 39, 15, 35, 9, 51],
              [34, 17, 10, 21, 9, 33, 14, 26, 19, 45]]

SupplyCenter = ['S_1', 'S_2', 'S_3', 'S_4', 'S_5', 'S_6', 'S_7', 'S_8']
Supply = [110, 80, 100, 240, 100, 280, 130, 440]
DemandMarket = ['D_1', 'D_2', 'D_3', 'D_4', 'D_5', 'D_6', 'D_7', 'D_8', 'D_9', 'D_10']
Demand = [90, 100, 150, 100, 180, 240, 210, 90, 160, 70]

n_supply = len(SupplyCenter)
n_demand = len(DemandMarket)
Range_supply = range(n_supply)
Range_demand = range(n_demand)

#####
# Optimization Problem:
#####

model = gp.Model("BalancedMilk.1")
model.Params.LogToConsole = 0 # Asking Gurobi not to give us all the details!

# Defining the matrix of decision variables:
X = model.addVars(n_supply, n_demand, lb = 0, vtype = GRB.CONTINUOUS, name = ["Supply from "+SupplyCenter[i]+" to "+DemandMarket[j] for i in Range_supply for j in Range_demand])

# Objective: Minimizing the total distribution cost:
exp = gp.quicksum(Cost[i][j]*X[i, j] for i in Range_supply for j in Range_demand)
model.setObjective(exp, GRB.MINIMIZE)

# Constraints:
# 1. Supply constraints:
# For each supply center, the amount of milk supplied from this center should be equal to its production:
for i in Range_supply:
    model.addConstr(sum(X[i, j] for j in Range_demand) == Supply[i], "Supply_"+SupplyCenter[i])

# 2. Demand constraints:
# For each demand market, the amount of milk demanded from this market should not surpass its demand:
for j in Range_demand:
    model.addConstr(sum(X[i, j] for i in Range_supply) <= Demand[j], "Demand_"+DemandMarket[j])

# Optimization
model.optimize()

# Printing the results:

if model.status == GRB.OPTIMAL:
    print('Solution status is optimal, and the minimum cost is: $%g. \n' % model.objVal)
    for v in model.getVars():
        if v.x > 0:
            print(v.varName, ':', v.x,)
else:
    print('Optimization was stopped with status %d' % model.status)

print("\n")

Solution = model.getAttr('x', X)
for j in Range_demand:
    # The demands satisfied from the dummy supplier are not satisfied in reality!
    print(DemandMarket[j],
          "is experiencing a supply shortage of {} tonnes.".format(Demand[j] - sum(Solution[i, j] for i in Range_supply)))
```

### The incident!

Due to a cattle virus outbreak in the largest supplier ( $S_8$ ), the Ministry of Health shut down this dairy farm. Left with seven suppliers and unchanged demand, *BalancedMilk* is facing a supply shortage. Therefore, the management establishes two teams to search for short-term and long-term strategies to combat this issue. While the *long-term team* is looking for alternative suppliers or new contracts, we want to evaluate the strategies the *short-term team* offers in the following problems. In all of the following problems, there is no milk supply from  $S_8$ .

**Note:** Before the shutdown of  $S_8$ , the total supply and demand were equal - thereby a "balanced" situation. You are familiar with balanced problems; now, we enter the imbalanced realm!

### Problem 2: (7.5 pts)

Suppose not supplying milk to a demand market is costless. Nevertheless, **the firm must distribute all the milk the remaining seven supply centers provide**. Also, remember that **the firm cannot oversupply a demand market**.

Use the code for the previous part and adjust it for the new situation to get the optimal allocation and cost. Report the optimal cost. Hint: There are many ways to do this, but the simplest way is to adjust the supply vector!

```
In [7]: ## Your code for Problem 2 goes here:
```

### Problem 3: (5 pts)

In the real world, with ruthless competitors ready to attack *BalancedMilk*'s market share, the cost of not supplying milk to a demand center is not zero. Suppose the cost of not delivering each tonne of milk to each demand center is 20 dollars/tonne. Adjust the code you wrote for Problem 2 to give you the optimal cost and allocation. Report the optimal cost. (Remember, like Problem 2, the firm should distribute all the milk provided by the remaining seven supply centers and cannot oversupply a demand market.)

Hint-1: One way to implement this is to adjust the objective function. Hint-2: If done correctly, the new optimal cost must be \$19470.

```
In [ ]: ## Your code for Problem 3 goes here:
```

### Problem 4: (5 pts)

Now, suppose the cost of not delivering each tonne of milk to each demand center is 100 dollars/tonne instead of 20 dollars/tonne. Adjust your code from Problem 3 for the new situation to give you the optimal cost and allocation. (Remember, like Problem 2, the firm should distribute all the milk provided by the remaining seven supply centers and cannot oversupply a demand market.) Report the optimal cost.

```
In [8]: ## Your code for Problem 4 goes here:
```

### Problem 5: (10 pts)

You have the optimal allocations for Problems 2, 3, and 4. Are they different? Are they similar? Why? Justify your answer.

Your explanation for Problem 5: ...

### Problem 6.1: (5 pts)

Until now, we have assumed the firm is committed to distributing all the milk provided by the suppliers. Now, suppose the firm is not committed to distributing all the milk provided by the suppliers. In other words, the firm can choose not to distribute some of the milk provided by the suppliers. Suppose the cost of not distributing each tonne of milk is 20 dollars/tonne. Adjust your code from Problem 3 for the new situation to give you the optimal allocation and cost. Report the optimal cost.

```
In [5]: ## Your code for Problem 6.1 goes here:
```

### Problem 6.2: (7.5 pts)

Compare the optimal allocations for Problem 6.1 and Problem 3. Hint: They are different!

Why? Explain the reason behind this difference.

Your explanation for Problem 6.2: ...

### Problem 7.1: (5 pts)

Suppose the firm is not committed to distributing all the milk provided by the suppliers. In other words, the firm can choose not to distribute some of the milk provided by the suppliers. Suppose the cost of not distributing each tonne of milk is 100 dollars/tonne instead of 20 dollars/tonne. Adjust your code from Problem 4 for the new situation to give you the optimal allocation and cost. Report the optimal cost.

```
In [1]: ## Your code for Problem 7.1 goes here:
```

### Problem 7.2: (7.5 pts)

Compare the optimal allocations for Problem 7.1 and Problem 4. Are they different? Are they the same? Why? Explain your answer.

Your explanation for Problem 7.2: ...

### Problem 8.1: (10 pts)

We have assumed the no-supply cost is the same for all demand centers until now. However, this may not be the case. The estimations by the *short-term team* indicate that the cost of not supplying a tone of milk to each demand center is as follows:

	$D_1$	$D_2$	$D_3$	$D_4$	$D_5$	$D_6$	$D_7$	$D_8$	$D_9$	$D_{10}$
Cost of not supplying (dollars/tonne)	15	20	25	30	28	35	32	15	25	10

Additionally, like Problems 2, 3, and 4, the firm should distribute all the milk provided by the remaining seven supply centers. Use your code for Problems 2 or 3 or 4, and adjust it for the new situation. Get the optimal allocation and the optimal cost. Report the optimal cost.

Hint: One way to do this is to adjust the objective function.

```
In [17]: ## Your code for Problem 8.1 goes here:
```

### Problem 8.2: (10 pts)

Compare the optimal allocation for Problem 8.1 and Problems 2, 3, or 4. Are they different? Are they the same? Why? Explain your answer.

Your explanation for Problem 8.2: ...

### Problem 9.1: (5 pts)

Suppose not supplying milk to a demand market is costless, and the firm is committed to supplying all the available milk - like Problem 2. But this time, the firm decides to supply the milk so that each demand center receives at least 50% of its demand. Use the code for Problem 2 and adjust it for the new situation to get the optimal allocation and cost. Report the optimal cost.

```
In [ ]: ## Your code for Problem 9.1 goes here:
```

### Problem 9.2: (7.5 pts)

Compare the optimal cost with your answer to Problem 2. Is it more, or is it less than the optimal cost for Problem 2? Why?

Your explanation for Problem 9.2: ...

### Prolem 9.3: (7.5 pts)

Suppose the firm sets this bar to 90%. Every demand center must receive at least 90% of its demand. Is it possible? Why? What is the maximum possible percentage?

Your answer for Problem 9.3: ...

## Sensitivity Analysis

(To obtain the sensitivity analysis for each problem, use the following code to give you the optimal cost and the sensitivity analysis instead of the optimal allocation.)

```
In [ ]: # Printing the results:

if model.status == GRB.OPTIMAL:
    print('Solution status is optimal, and the minimum cost is: $%g.' % model.objVal)
#    for v in model.getVars():
#        if v.x > 0:
#            print(v.varName, ':', v.x,)
#    header = ["<15] | {<6} | {<6} | {<6} | {<6} | {<10} | {<10} |".format("Name", "Sense", "Slack", "Shadow Price", "RHS", "SARHSLow", "SARHSUp")]
#    print(header)
#    print("-" * len(header)) # Print a separator line
#    for constr in model.getConstrs():
#        name = constr.ConstrName
#        sense = constr.Sense
#        slack = constr.Slack
#        pi_val = constr.Pi
#        rhs = constr.RHS
#        sarhslow = constr.SARHSLow
#        sarhsup = constr.SARHSUp
#
#        row = ["<15] | {<6} | {<6.2f} | {<6.2f} | {<10.2f} | {<10.2f} |".format(name, sense, slack, pi_val, rhs, sarhslow, sarhsup)]
#        print(row)
#    else:
#        print('Optimization was stopped with status %d' % model.status)
```

Explanations:

- Sense: the nature of the constraint. If it is >=, <=, or ==.
- RHS: the right-hand side of the constraint.
- Slack: The gap between the left-hand side of a constraint and the RHS in the optimal point.
- Shadow Price: Please refer to the slides for the definition of the shadow price.
- SARHSLow: The lowest value of the RHS for which we can still use this sensitivity analysis table to estimate the optimal cost, and we do not need to re-optimize the problem to estimate the optimal cost.
- SARHSUp: The highest value of the RHS for which we can still use this sensitivity analysis table to estimate the optimal cost, and we do not need to re-optimize the problem to estimate the optimal cost.

(SARHSLow and SARHSUp give us the allowable range of the RHS for which the geometry of the optimal solution remains unchanged.)

### Problem 10.1: (15 pts)

Rewrite the solution for Problem 2 here, but replace the last part with the code above to get the sensitivity analysis. If done correctly, the shadow price for all supply centers is positive. Now, answer the following questions.

- If we increase the supply of  $S_1$  by 10 tonnes from 110 to 120, how much would the optimal cost change?
- Suppose we want to estimate the optimal cost without re-optimizing the problem. How much can we increase or decrease the supply of  $S_1$  to calculate the optimal cost without re-optimizing the problem?
- Explain what a positive shadow price in a minimization problem means, then explain why in Problem 2 the shadow price for all supply centers is positive.

```
In [4]: ## Your code for Problem 10.1 goes here:
```

Your explanation for Problem 10.1: ...

### Problem 10.2: (10 pts)

In the settings of Problem 2, what is even more interesting than positive shadow prices for all supply centers is that the shadow price for some demand markets is negative! How is it possible that increasing the demand for a demand market decreases the optimal cost?

Hint: If you want to gain some intuition, you can change the demand for  $D_2$  by a small amount and see what happens to the slack variable for  $D_4$ ,  $D_7$  and  $D_8$ .

Your explanation for Problem 10.2: ...