**Code**

Code used for this project was adapted by following this provided example linked from the Gurobi MIP tutorial page. https://colab.research.google.com/github/Gurobi/modeling-examples/blob/master/facility\_location/facility\_location.ipynb#scrollTo=DvJ4hZzMoL-q

1. Setting Up Data
2. import gurobipy as gp
3. from gurobipy import GRB, quicksum
4. *# --------------------*
5. *# Data*
6. *# --------------------*
7. plants = ['Pontiac', 'Cincinnati', 'Dayton', 'Atlanta']
8. dcs = ['Milwaukee', 'Dayton', 'Cincinnati', 'Buffalo', 'Atlanta']
9. *# Weekly demand at each distribution center*
10. demand = {
11. 'Milwaukee': 10000,
12. 'Dayton': 15000,
13. 'Cincinnati': 16000,
14. 'Buffalo': 19000,
15. 'Atlanta': 12000
16. }
17. *# Plant capacity*
18. capacity = {
19. 'Pontiac': 27000,
20. 'Cincinnati': 40000,
21. 'Dayton': 40000,
22. 'Atlanta': 40000
23. }
24. *# Production cost per pair of shoes*
25. production\_cost = {
26. 'Pontiac': 2.70,
27. 'Cincinnati': 2.64,
28. 'Dayton': 2.69,
29. 'Atlanta': 2.62
30. }
31. *# Fixed cost to operate each plant*
32. fixed\_cost = {
33. 'Pontiac': 7000,
34. 'Cincinnati': 4000,
35. 'Dayton': 6000,
36. 'Atlanta': 7000
37. }
38. *# Distribution costs per pair (from plant to DC)*
39. distribution\_cost = {
40. ('Milwaukee', 'Pontiac'): 0.42,
41. ('Milwaukee', 'Cincinnati'): 0.46,
42. ('Milwaukee', 'Dayton'): 0.44,
43. ('Milwaukee', 'Atlanta'): 0.48,
44. ('Dayton', 'Pontiac'): 0.36,
45. ('Dayton', 'Cincinnati'): 0.37,
46. ('Dayton', 'Dayton'): 0.30,
47. ('Dayton', 'Atlanta'): 0.45,
48. ('Cincinnati', 'Pontiac'): 0.41,
49. ('Cincinnati', 'Cincinnati'): 0.30,
50. ('Cincinnati', 'Dayton'): 0.37,
51. ('Cincinnati', 'Atlanta'): 0.43,
52. ('Buffalo', 'Pontiac'): 0.39,
53. ('Buffalo', 'Cincinnati'): 0.42,
54. ('Buffalo', 'Dayton'): 0.38,
55. ('Buffalo', 'Atlanta'): 0.46,
56. ('Atlanta', 'Pontiac'): 0.50,
57. ('Atlanta', 'Cincinnati'): 0.43,
58. ('Atlanta', 'Dayton'): 0.45,
59. ('Atlanta', 'Atlanta'): 0.27,
60. }

2. Setting up the model

*# Defining the model*

def solve\_spencer\_plant\_location(require\_pontiac\_open=True):

    model = gp.Model("SpencerShoePlantLocation")

*# Decision vars*

    open\_plant = model.addVars(plants, vtype=GRB.BINARY, name="Open")

    ship = model.addVars(dcs, plants, vtype=GRB.CONTINUOUS, name="Ship")

*# Objective: Total cost (shipping + production + fixed)*

    model.setObjective(

        quicksum(

            ship[i, j] \* (distribution\_cost[i, j] + production\_cost[j])

*for* i *in* dcs *for* j *in* plants

        ) +

        quicksum(fixed\_cost[j] \* open\_plant[j] *for* j *in* plants),

        GRB.MINIMIZE

    )

*# Each DC's demand must be met*

*for* i *in* dcs:

        model.addConstr(quicksum(ship[i, j] *for* j *in* plants) == demand[i], name=f"demand\_{i}")

*# Plant capacity respected*

*for* j *in* plants:

        model.addConstr(quicksum(ship[i, j] *for* i *in* dcs) <= capacity[j] \* open\_plant[j], name=f"capacity\_{j}")

*# At least one new plant must be opened (not counting Pontiac)*

    model.addConstr(quicksum(open\_plant[j] *for* j *in* plants *if* j != 'Pontiac') >= 1, name="at\_least\_one\_new")

*# Optionally require Pontiac to remain open*

*if* require\_pontiac\_open:

        model.addConstr(open\_plant['Pontiac'] == 1, name="force\_Pontiac\_open")

*# Solve*

    model.optimize()

*# Output*

*if* model.Status == GRB.OPTIMAL:

        print("\nOptimal solution found")

        print(f"Total cost: ${model.ObjVal:,.2f}")

        print("\nPlants to Open:")

*for* j *in* plants:

*if* open\_plant[j].X > 0.5:

                print(f"  - {j}")

        print("\nShipments:")

*for* i *in* dcs:

*for* j *in* plants:

*if* ship[i, j].X > 1e-2:

                    print(f"  {int(ship[i,j].X):5d} pairs from {j} to {i}")

*else*:

        print("No optimal solution found.")

3. Running the model

print("=== Part (a): Pontiac must remain open ===")

solve\_spencer\_plant\_location(require\_pontiac\_open=True)

print("\n=== Part (b): Pontiac may be closed ===")

solve\_spencer\_plant\_location(require\_pontiac\_open=False)

**Output**

=== Part (a): Pontiac must remain open ===

Gurobi Optimizer version 12.0.2 build v12.0.2rc0 (win64 - Windows 11.0 (26100.2))

CPU model: AMD Ryzen 5 5600X 6-Core Processor, instruction set [SSE2|AVX|AVX2]

Thread count: 6 physical cores, 12 logical processors, using up to 12 threads

Optimize a model with 11 rows, 24 columns and 48 nonzeros

Model fingerprint: 0x44bceab4

Variable types: 20 continuous, 4 integer (4 binary)

Coefficient statistics:

Matrix range [1e+00, 4e+04]

Objective range [3e+00, 7e+03]

Bounds range [1e+00, 1e+00]

RHS range [1e+00, 2e+04]

Presolve removed 1 rows and 1 columns

Presolve time: 0.00s

Presolved: 10 rows, 23 columns, 46 nonzeros

Variable types: 20 continuous, 3 integer (3 binary)

Found heuristic solution: objective 239710.00000

Root relaxation: objective 2.291200e+05, 6 iterations, 0.00 seconds (0.00 work units)

Nodes | Current Node | Objective Bounds | Work

Expl Unexpl | Obj Depth IntInf | Incumbent BestBd Gap | It/Node Time

0 0 229120.000 0 2 239710.000 229120.000 4.42% - 0s

H 0 0 234720.00000 229120.000 2.39% - 0s

0 0 231580.000 0 2 234720.000 231580.000 1.34% - 0s

H 0 0 234210.00000 231580.000 1.12% - 0s

Cutting planes:

Implied bound: 3

Flow cover: 2

Explored 1 nodes (10 simplex iterations) in 0.01 seconds (0.00 work units)

Thread count was 12 (of 12 available processors)

Solution count 2: 234210 239710

Optimal solution found (tolerance 1.00e-04)

Best objective 2.342100000000e+05, best bound 2.342100000000e+05, gap 0.0000%

Optimal solution found

Total cost: $234,210.00

Plants to Open:

- Pontiac

- Cincinnati

- Atlanta

Shipments:

10000 pairs from Atlanta to Milwaukee

15000 pairs from Cincinnati to Dayton

16000 pairs from Cincinnati to Cincinnati

9000 pairs from Cincinnati to Buffalo

10000 pairs from Atlanta to Buffalo

12000 pairs from Atlanta to Atlanta

=== Part (b): Pontiac may be closed ===

Gurobi Optimizer version 12.0.2 build v12.0.2rc0 (win64 - Windows 11.0 (26100.2))

CPU model: AMD Ryzen 5 5600X 6-Core Processor, instruction set [SSE2|AVX|AVX2]

Thread count: 6 physical cores, 12 logical processors, using up to 12 threads

Optimize a model with 10 rows, 24 columns and 47 nonzeros

Model fingerprint: 0xca338eaf

Variable types: 20 continuous, 4 integer (4 binary)

Coefficient statistics:

Matrix range [1e+00, 4e+04]

Objective range [3e+00, 7e+03]

Bounds range [1e+00, 1e+00]

RHS range [1e+00, 2e+04]

Presolve time: 0.00s

Presolved: 10 rows, 24 columns, 47 nonzeros

Variable types: 20 continuous, 4 integer (4 binary)

Found heuristic solution: objective 239710.00000

Root relaxation: objective 2.248600e+05, 8 iterations, 0.00 seconds (0.00 work units)

Nodes | Current Node | Objective Bounds | Work

Expl Unexpl | Obj Depth IntInf | Incumbent BestBd Gap | It/Node Time

0 0 224860.000 0 2 239710.000 224860.000 6.19% - 0s

H 0 0 232760.00000 224860.000 3.39% - 0s

H 0 0 228040.00000 224860.000 1.39% - 0s

H 0 0 227370.00000 226290.000 0.47% - 0s

0 0 226650.000 0 1 227370.000 226650.000 0.32% - 0s

H 0 0 227210.00000 226650.000 0.25% - 0s

Cutting planes:

Implied bound: 3

Flow cover: 2

Explored 1 nodes (12 simplex iterations) in 0.01 seconds (0.00 work units)

Thread count was 12 (of 12 available processors)

Solution count 4: 227210 228040 232760 239710

Optimal solution found (tolerance 1.00e-04)

Best objective 2.272100000000e+05, best bound 2.272100000000e+05, gap 0.0000%

Optimal solution found

Total cost: $227,210.00

Plants to Open:

- Cincinnati

- Atlanta

Shipments:

10000 pairs from Atlanta to Milwaukee

15000 pairs from Cincinnati to Dayton

16000 pairs from Cincinnati to Cincinnati

9000 pairs from Cincinnati to Buffalo

10000 pairs from Atlanta to Buffalo

12000 pairs from Atlanta to Atlanta

**Answers**

1. Using the outputs of your code, answer the following questions**a**and **b;** be sure to label your responses as "a" and "b."**Assume that the Pontiac plant:**
   1. has no resale value and must remain open. What are the plant locations that will minimize total costs, including production, distribution, and fixed costs? What is the optimal total cost?

If the Pontiac plant has no resale value and must remain open, the plant locations that will be most optimal are Pontiac, Cincinnati, and Atlanta because they meet the demand of all distribution centers and also minimize all necessary costs. The total optimal cost is $234,210.00

* 1. could be closed at zero net cost. What are the optimal locations? What is the optimal total cost?

If the Pontiac plant can be closed at zero net cost, the optimal locations are only the Cincinnati and Atlanta plants. This is because these two plants alone can meet all the demand requirements will minimizing necessary costs. The total optimal cost is $227,210.00