Exercises - Section 4: Lecture 12 – Types of Optimization Models - Questions

1. For each of the mathematical optimization models below, specify which of the following types of optimization models it is. In each model, x are the variables, all other letters refer to known data.

TYPES OF MODELS

i. Linear program
ii. Integer program
iii. Quadratic program
iv. Quadratic integer program
v. Quadratically-constrained program
vi. Quadratically-constrained integer program
vii. Quadratic, quadratically-constrained program
viii. Quadratic, quadratically-constrained integer program
ix. None of the above

a. minimize $\sum_i c_i x_i$

subject to $\sum_i a_{ij} x_i \leq b_j$ for all j
all $x_i \geq 0$

b. minimize $\sum_i c_i x_i$

subject to $\sum_i a_{ij} x_i \leq b_j$ for all j
all $x_i \in \{0,1\}$

c. minimize $\sum_i c_i x_i^2$

subject to $\sum_i a_{ij} x_i \leq b_j$ for all j
all $x_i \geq 0$

d. minimize $\sum_i c_i \, x_i$

subject to $\sum_i a_{ij} x_i^2 \leq b_j$ for all j
all $x_i \geq 0$

e. minimize $\sum_i c_i^2 x_i$

subject to $\sum_i a_{ij} x_i \leq b_j$ for all j

all $x_i \geq 0$

f. minimize $\sum_i c_i x_i^2$

subject to $\sum_i a_{ij} x_i^2 \leq b_j$ for all j
all $x_i \geq 0$

g. minimize $\sum_i c_i x_i$

subject to $\sum_i \sum_j a_{ij} x_i x_j \leq b$
all $x_i \in \{0,1\}$

h. minimize $\sum_i c_i x_i^2$

subject to $\sum_i a_{ij} x_i \leq b_j$ for all j
all $x_i \geq 0$, integer

i. minimize $\sum_i c_i x_i^2$

subject to $\sum_i a_{ij} x_i^2 \leq b_j$ for all j
all $x_i \in \{0,1\}$

j. minimize $\sum_i c_i^2 x_i$

subject to $\sum_i a_{ij}^{1.5} x_i \leq b_j^3$ for all j
all $x_i \in \{0,1\}$

k. minimize $\sum_i c_i x_i^3$

subject to $\sum_i a_{ij} x_i \leq b_j$ for all j
all $x_i \geq 0$

l. minimize $\sum_i c_i^2 x_i$

subject to $\sum_i \sum_k a_{ik} x_i x_k \leq b_j$ for all j
all $x_i \geq 0$, integer

2. For each of the gurobipy optimization models below, specify which of the following types of optimization models it is. In each model, x are the variables, all other letters refer to known data.

TYPES OF MODELS

i. Linear program
ii. Integer program
iii. Quadratic program
iv. Quadratic integer program
v. Quadratically-constrained program
vi. Quadratically-constrained integer program
vii. Quadratic, quadratically-constrained program
viii. Quadratic, quadratically-constrained integer program
ix. None of the above

a.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB.CONTINUOUS, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

b.

Variables	`x = model.addVars(range(1,N+1), vtype=gp.GRB.BINARY, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

c.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB. CONTINUOUS, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

d.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB. CONTINUOUS, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

e.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB.CONTINUOUS, name="x")`
Objective	`model.setObjective(sum((c[i]**2)*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

f.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB. CONTINUOUS, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

g.

Variables	`x = model.addVars(range(1,N+1), vtype=gp.GRB.BINARY, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstr(sum(sum(a[i,j]*x[i]*x[j] for i in range(1,N+1) for j in range(1,M+1)) <= b[j])`

h.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB.INTEGER, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

i.

Variables	`x = model.addVars(range(1,N+1), vtype=gp.GRB.BINARY, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

j.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB.BINARY, name="x")`
Objective	`model.setObjective(sum((c[i]**2)*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*math.sqrt(a[i,j])*x[i] for i in range(1,N+1)) <= b[j]*b[j]*b[j]) for j in range(1,M+1))`

k.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB.CONTINUOUS, name="x")`
Objective	`model.setObjective(sum(c[i]*x[i]*x[i]*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,j]*x[i] for i in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

l.

Variables	`x = model.addVars(range(1,N+1), lb=0, vtype=gp.GRB.INTEGER, name="x")`
Objective	`model.setObjective(sum((c[i]**2)*x[i] for i in range(1,N+1)), GRB.MINIMIZE)`
Constraints	`model.addConstrs((sum(a[i,k]*x[i]*x[k] for i in range(1,N+1) for k in range(1,N+1)) <= b[j]) for j in range(1,M+1))`

NOTES: