OR_Investments

September 7, 2025

[11]:

```
import gurobipy as gp
from gurobipy import GRB
import pandas as pd
```

[12]:

```
# Load data from investment.csv
df = pd.read_csv('/Users/utkarshtyagi/Documents/VSC/Dataset/investment.csv')
```

[13]:

```
# Parameters

s = df['Current Share'].values   # s_i
p = df['Buying Price'].values    # p_i
q = df['Current Price'].values   # q_i
r = df['Expected Future Price'].values  # r_i
```

[14]:

```
# Number of stocks
n = len(s)
K = 9000  # Target net proceeds
```

[15]:

```
# Calculate net proceeds per share c_i for each stock
c = []
for i in range(n):
    if q[i] > p[i]:
        # Capital gain: tax 30% on gain, transaction cost 1%
        c_i = 0.69 * q[i] + 0.30 * p[i]
    else:
        # No capital gain: only transaction cost 1%
        c_i = 0.99 * q[i]
    c.append(c_i)
```

[16]:

```
# Create a new model
model = gp.Model("Investment_Under_Taxation")

# Decision variables: x_i = number of shares to sell for stock i
x = model.addVars(n, lb=0, ub=s, name="x")

# Set objective: minimize sum(r_i * x_i) [equivalent to maximizing expected␣
 ↪portfolio value]
obj = gp.quicksum(r[i] * x[i] for i in range(n))
```

```python
model.setObjective(obj, GRB.MINIMIZE)

# Constraint: total net proceeds >= K
model.addConstr(gp.quicksum(c[i] * x[i] for i in range(n)) >= K, "NetProceeds")
```

[16]:

```
<gurobi.Constr *Awaiting Model Update*>
```

[17]:

```python
# Optimize the model
model.optimize()

# Check if the model is solved optimally
if model.status == GRB.OPTIMAL:
    print("\nOptimal solution found:")
    total_net = sum(c[i] * x[i].x for i in range(n))
    expected_value = sum(r[i] * (s[i] - x[i].x) for i in range(n))
    print(f"Total net proceeds raised: ${total_net:.2f}")
    print(f"Expected portfolio value next year: ${expected_value:.2f}")
    print("\nShares to sell for each stock:")
    for i in range(n):
        stock_name = df['Stock'].iloc[i]
        print(f"{stock_name}: {x[i].x:.2f} shares (out of {s[i]})")
else:
    print("No optimal solution found. Status:", model.status)
```

```
Gurobi Optimizer version 12.0.2 build v12.0.2rc0 (mac64[arm] - Darwin 24.6.0
24G84)

CPU model: Apple M4
Thread count: 10 physical cores, 10 logical processors, using up to 10 threads

Optimize a model with 1 rows, 5 columns and 5 nonzeros
Model fingerprint: 0xcd7e38b5
Coefficient statistics:
  Matrix range     [2e+00, 6e+00]
  Objective range  [2e+00, 9e+00]
  Bounds range     [1e+03, 1e+03]
  RHS range        [9e+03, 9e+03]
Presolve time: 0.00s
Presolved: 1 rows, 5 columns, 5 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    0.0000000e+00   5.625000e+02   0.000000e+00      0s
       1    1.1743209e+04   0.000000e+00   0.000000e+00      0s

Solved in 1 iterations and 0.01 seconds (0.00 work units)
Optimal objective  1.174320910e+04

Optimal solution found:
```

```
Total net proceeds raised: $9000.00
Expected portfolio value next year: $15356.79

Shares to sell for each stock:
S1: 1000.00 shares (out of 1000)
S2: 1000.00 shares (out of 1000)
S3: 1000.00 shares (out of 1000)
S4: 118.76 shares (out of 1000)
S5: 0.00 shares (out of 1000)
```

[ ]: