PROJECT 1 - LINEAR PROGRAMMING

GROUP 12	
Bhuvana Chandrika Kothapalli	Bk24542
Mayank Gupta	mg66426
Anmol Agrawal	aa96674
Anvesh Karangula	ak49956

Aim

We are currently facing the challenge of understanding how the impact of marketing strategies differs from one company to another. What proves successful for one company may not yield the same results for another. To tackle this issue, we are employing linear programming techniques to devise a strategy for allocating our marketing budget effectively.

Methodology

In order to develop an optimal marketing budget allocation strategy, we rely on data concerning the expected return on investment for each marketing channel. This data has been sourced from two consulting firms and serves as the foundation for our analysis.

Speaking of the analysis itself, our primary goal is to maximize revenue, and the insights derived from this analysis will guide our marketing expenditure for the upcoming year. We've conducted this analysis on both an annual and a monthly basis, employing linear programming as the optimization technique. The Gurobi package in Python was utilized to perform this analysis.

Constraints

1. The total Budget allocated for marketing is $10M.
2. The amount invested in print and TV should be no more than the amount spent on Facebook and Email.
3. The Total amount used in social media should be twice that of SEO and AdWords.
4. For each platform, the amount invested should be no more than $3M.

Analysis:

1-3. Budget Allocation based on ROI data from the first consulting firm:

Objective Function:

The objective is to maximise the returns from the marketing expenditure. ROI data for various marketing mediums is given in the "ROI_data.csv" file. Below is the objective function as per the estimated ROI data provided by the first consulting firm.

Maximize: (0.031 * x1 + 0.049 * x2 + 0.024 * x3 + 0.039 * x4 + x5 + 0.016 * x6 + 0.046 * x7 + 0.026 * x8 + 0.033 * x9 + 0.044 * x10)

where below are the variables considered:

x1 = Budget allocation to Print
x2 = Budget allocation to TV
x3 = Budget allocation to SEO
x4 = Budget allocation to AdWords
x5 = Budget allocation to Facebook
x6 = Budget allocation to LinkedIn
x7 = Budget allocation to Instagram
x8 = Budget allocation to Snapchat
x9 = Budget allocation to Twitter
x10 = Budget allocation to Email

Constraints:

Converting the budget restrictions to the constraints to be used in this optimization problem is as follows:

1. x1 + x2 <= x5 + x10 (The budget spent on Print and TV should be no more than the budget spent on Facebook and Email)
2. x5 + x6 + x7 + x8 + x9 >= 2 * (x3 + x4) (The total budget used in social media should be at least twice the budget used in SEO and AdWords)
3. For each platform, the budget allocation should be no more than 3M:
   x1, x2, x3, x4, x5, x6, x7, x8, x9, x10 <= 3
4. The total budget spent should be less than or equal to 10M:
   x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 <= 10

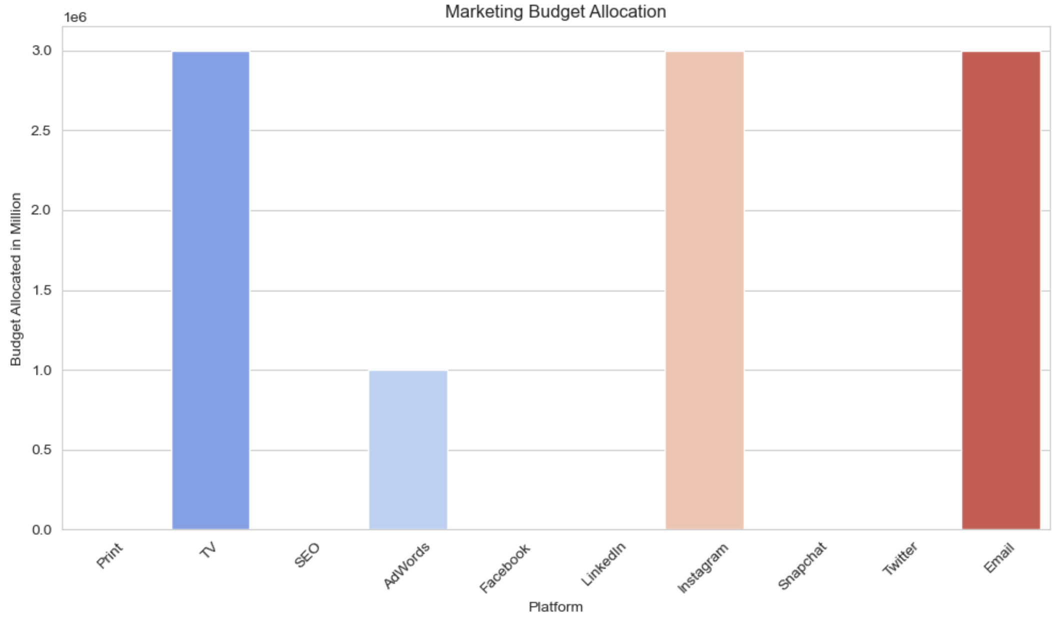
Results:

Solved the above objective function along with the constraints as a linear programming problem using the Gurobi package. Budget allocation restriction for each marketing medium was input as an upper bound while adding the variables to the gurobi model. Below is the optimal allocation obtained by optimizing the objective function using gurobi:

```
Optimal solution found
Total ROI: 456000.0

Allocation
            Allocation
Print              0.0
TV           3000000.0
SEO                0.0
AdWords      1000000.0
Facebook           0.0
LinkedIn           0.0
Instagram    3000000.0
Snapchat           0.0
Twitter            0.0
Email        3000000.0
```

The graphical representation of the optimal allocation is as follows:



Optimal returns obtained using the estimated ROI data from the first consulting firm is $456,000.

4. Budget Allocation based on ROI data from the first consulting firm:

Objective Function:

The objective function remains the same as in the first case but the coefficients differ as per the estimated ROI data provided by the second consulting firm. Below is the objective function as per the estimated ROI data provided by the second consulting firm.

Maximize: (0.049 * x1 + 0.023 * x2 + 0.024 * x3 + 0.039 * x4 + 0.044*x5 + 0.046 * x6 + 0.026 * x7 + 0.019 * x8 + 0.037 * x9 + 0.026 * x10)

where below are the variables considered:

x1 = Budget allocation to Print

x2 = Budget allocation to TV
x3 = Budget allocation to SEO
x4 = Budget allocation to AdWords
x5 = Budget allocation to Facebook
x6 = Budget allocation to LinkedIn
x7 = Budget allocation to Instagram
x8 = Budget allocation to Snapchat
x9 = Budget allocation to Twitter
x10 = Budget allocation to Email

Constraints:
Constraints remain the same as the one's in the first case.

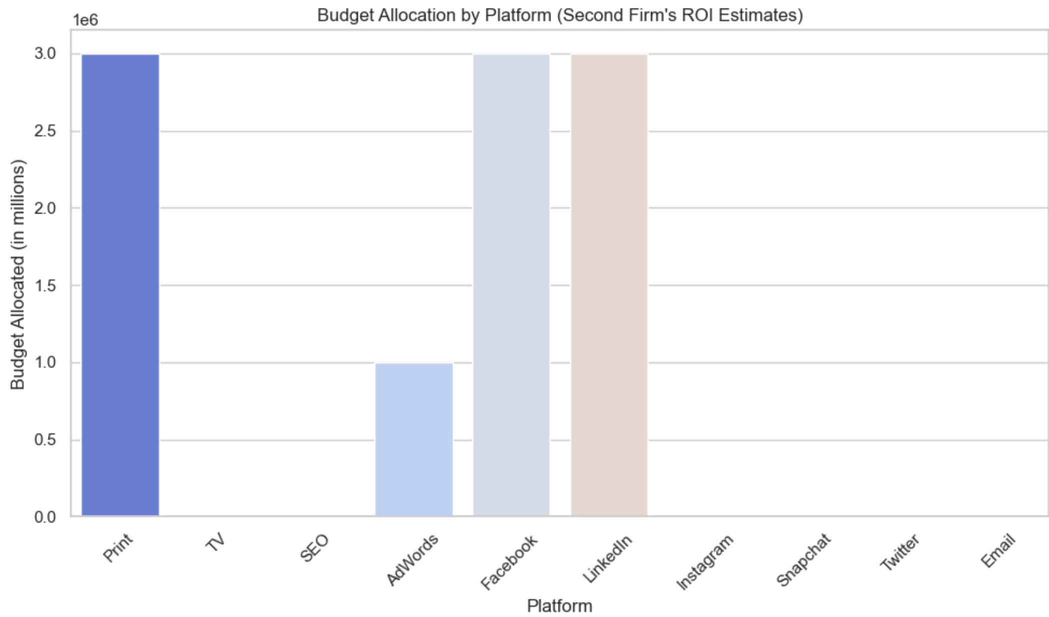
Results:

Below is the optimal allocation obtained by optimizing the objective function using the estimated ROI data from the second consulting firm:

```
Optimal solution found for the second firm
Total ROI (Second Firm): 456000.0

Allocation
           Allocation
Print       3000000.0
TV                0.0
SEO               0.0
AdWords     1000000.0
Facebook    3000000.0
LinkedIn    3000000.0
Instagram         0.0
Snapchat          0.0
Twitter           0.0
Email             0.0
```

The graphical representation of the optimal allocation is as follows:



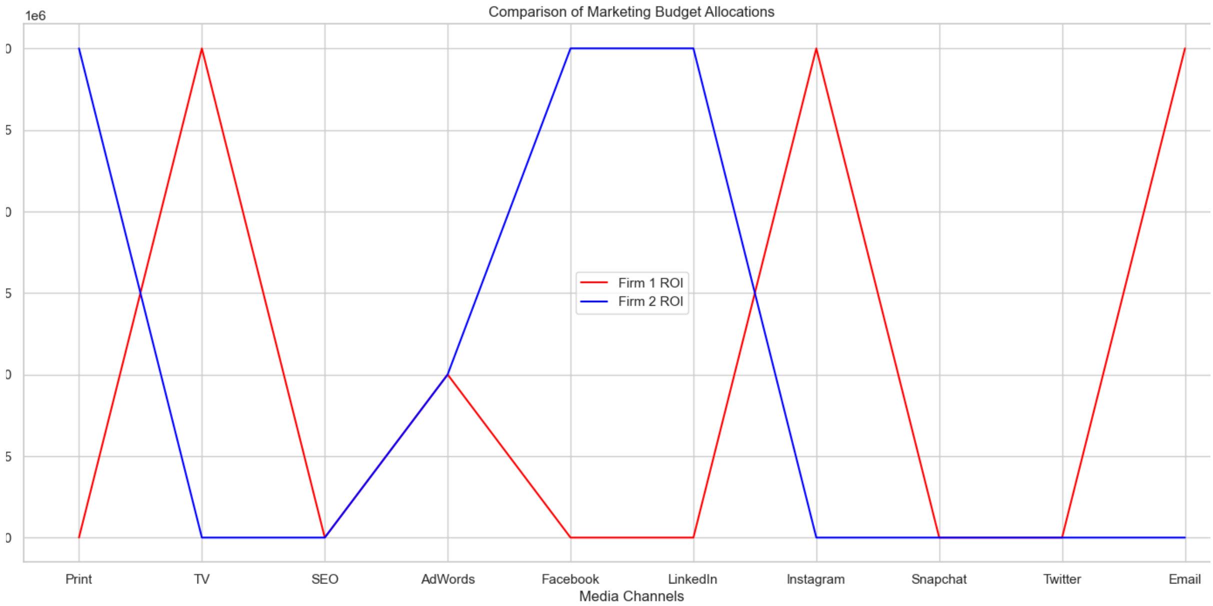
Optimal returns obtained using the estimated ROI data from the second consulting firm is $456,000.

5. Comparison of the Budget Allocations based on ROI data from both consulting firms:

a) Are the allocations the same?

The allocations are different for the two scenarios. In the first allocation, more budget is allocated to Print, Facebook, and LinkedIn, while in the second allocation, more budget is allocated to TV, Instagram, and Email.

The graphical representation of the comparison of the market medium allocations based on 2 ROI data is as below:



b) Assuming the first ROI data is correct, if you were to use the second allocation (the allocation that assumed the second ROI data was correct) how much lower would the objective be relative to the optimal objective (the one that uses the first ROI data and the first allocation)?

Assuming the first ROI data is correct, using the second allocation (the allocation that assumed the second ROI data was correct) will lower the ROI value to $252000 which is $204000 lower than the optimal ROI value obtained using first ROI data.

c) Assuming the second ROI data is correct, if you used the first allocation how much lower would the objective be relative to the optimal objective?

Similarly, assuming the second ROI data is correct, using the first allocation (the allocation that assumed the first ROI data was correct) will lower the ROI value to $264000 which is $192000 lower than the optimal ROI value obtained using second ROI data.

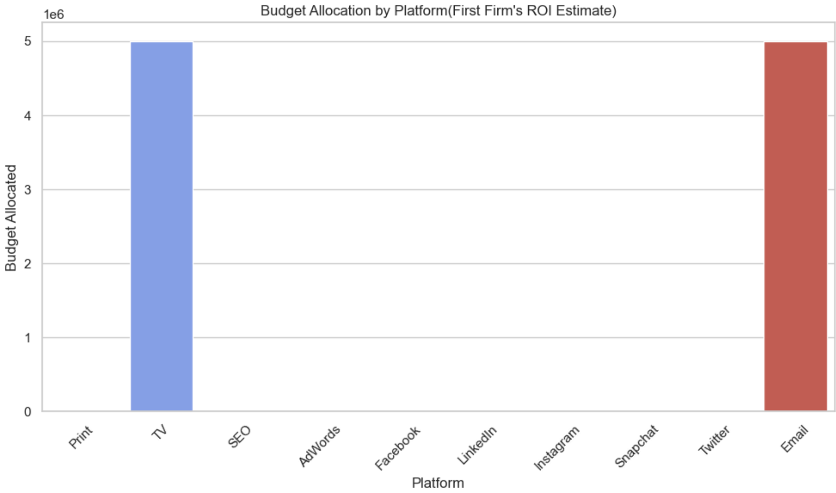
d) Do you think the third constraint above, based on your boss' experience, is useful?

As suggested by the Chief Marketing Officer, a constraint was added to limit the budget for an individual marketing medium. But removing this constraint led to higher returns of $465,000 and the allocation is as follows:

First Firm's optimal budget allocations:

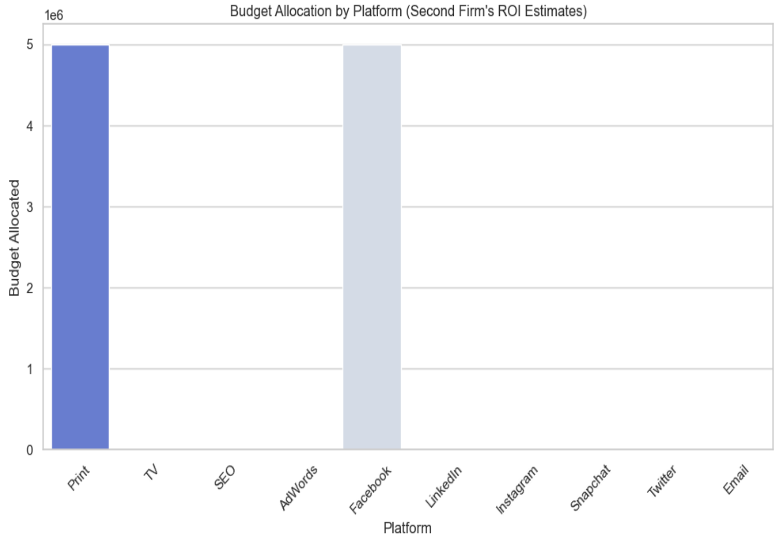
```
Optimal solution found
Total ROI: 465000.0

Allocation
           Allocation
Print             0.0
TV          5000000.0
SEO               0.0
AdWords           0.0
Facebook          0.0
LinkedIn          0.0
Instagram         0.0
Snapchat          0.0
Twitter           0.0
Email       5000000.0
```



Second Firm's optimal budget allocations:

```
Allocation
           Allocation
Print       5000000.0
TV                0.0
SEO               0.0
AdWords           0.0
Facebook    5000000.0
LinkedIn          0.0
Instagram         0.0
Snapchat          0.0
Twitter           0.0
Email             0.0
```



Removing the third constraint leads to a higher profit and doesn't influence optimal solution but it's important to consider the practical aspects of budget allocation. In the current optimal solution, only two media types are included in the budget allocation for media spending. This may not align with our overall marketing strategy, which likely involves a more diversified approach across various platforms. Allocating the budget with limitations ensures a more balanced and diversified allocation, reducing the risk of overconcentration on a small set of platforms.

6. To explore this further perform some analysis of how your optimal allocation would change based on changes in the ROI data. Use the first ROI data as your starting point. By how much could each advertising medium's ROI increase or decrease and still result in the same optimal allocation you found in step (3)?

Following are the specified upper and lower bounds for the return on investments (ROIs) for each advertising medium using which the optimal allocation remains unchanged.

	Lower Bounds	Upper Bounds
Print	-inf	0.049
TV	0.039	0.062
SEO	-inf	0.039
Adwords	0.033	0.046
Facebook	-inf	0.029
LinkedIn	-inf	0.039
Instagram	0.039	inf
Snapchat	-inf	0.039
Twitter	-inf	0.039
Email	0.029	inf

7. Your boss has gained permission to reinvest half of the return. For example, if the marketing obtains a 4% return in January, the budget of February will be 10M +10M × 4% × 50% = $10.2M. The monthly ROI for next year is given in an attached csv file. The three constraints given by your boss are still in place for each month. What is the optimal allocation for each month?

```
Monthly Budget Allocations and Returns:
```

Month	Print	TV	SEO	AdWords	Facebook	LinkedIn	Instagram	Snapchat	Twitter	Email	Returns	Budgets
January	3.000000	0.000000	0.0	1.333333	0.000000	0.0	2.666667	0.0	0.000000	3.000000	0.373000	10.000000
February	3.000000	0.000000	0.0	2.395500	3.000000	0.0	0.000000	0.0	1.791000	0.000000	0.406296	10.186500
March	0.000000	0.000000	0.0	3.000000	0.000000	3.0	1.203148	0.0	3.000000	0.000000	0.407516	10.203148
April	0.000000	0.000000	0.0	3.000000	0.000000	3.0	3.000000	0.0	1.203758	0.000000	0.400335	10.203758
May	1.200168	0.000000	0.0	0.000000	0.000000	0.0	3.000000	0.0	3.000000	3.000000	0.411006	10.200168
June	3.000000	0.000000	0.0	0.000000	0.000000	0.0	3.000000	0.0	1.205503	3.000000	0.423809	10.205503
July	0.000000	0.000000	0.0	3.000000	1.211905	0.0	3.000000	0.0	3.000000	0.000000	0.428264	10.211905
August	2.714132	0.000000	0.0	1.500000	0.000000	0.0	0.000000	0.0	3.000000	3.000000	0.437994	10.214132
September	0.609498	0.000000	0.0	3.000000	0.000000	3.0	0.000000	0.0	3.000000	0.609498	0.402712	10.218997
October	0.000000	0.000000	0.0	3.000000	0.000000	3.0	3.000000	0.0	0.000000	1.201356	0.371443	10.201356
November	3.000000	0.000000	0.0	1.185722	0.000000	0.0	3.000000	0.0	0.000000	3.000000	0.441615	10.185722
December	3.000000	2.110404	0.0	0.000000	3.000000	0.0	0.000000	0.0	0.000000	2.110404	0.432501	10.220807

8. A stable budget is defined as a monthly allocation such that for each platform the monthly change in spend is no more than $1M. Is the allocation you found stable? If it isn't, you do not need to solve a new optimization model. Describe how my might model this?

<u>MoM change in spend for each platform:</u>

Month	Print	TV	SEO	AdWords	Facebook	LinkedIn	Instagram	Snapchat	Twitter	Email
February	0.000000	0.000000	0.000000	1.062167	3.000000	0.000000	-2.666667	0.000000	1.791000	-3.000000
March	-3.000000	0.000000	0.000000	0.604500	-3.000000	3.000000	1.203148	0.000000	1.209000	0.000000
April	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.796852	0.000000	-1.796242	0.000000
May	1.200168	0.000000	0.000000	-3.000000	0.000000	-3.000000	0.000000	0.000000	1.796242	3.000000
June	1.799832	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	-1.794497	0.000000
July	-3.000000	0.000000	0.000000	3.000000	1.211905	0.000000	0.000000	0.000000	1.794497	-3.000000
August	2.714132	0.000000	0.000000	-1.500000	-1.211905	0.000000	-3.000000	0.000000	0.000000	3.000000
September	-2.104634	0.000000	0.000000	1.500000	0.000000	3.000000	0.000000	0.000000	0.000000	-2.390502
October	-0.609498	0.000000	0.000000	0.000000	0.000000	0.000000	3.000000	0.000000	-3.000000	0.591858
November	3.000000	0.000000	0.000000	-1.814278	0.000000	-3.000000	0.000000	0.000000	0.000000	1.798644
December	0.000000	2.110404	0.000000	-1.185722	3.000000	0.000000	-3.000000	0.000000	0.000000	-0.889596

The budget allocation for various marketing platforms exhibits fluctuations over the months, with some months experiencing significant changes exceeding $1 million in spending for specific platforms. To address this issue, it may be advisable to enhance the optimization model by incorporating constraints that limit monthly spending fluctuations, ensuring a more stable and predictable budget allocation over time. Hence, we can add 1 additional constraint for each variable(in total 10 more constraints to the model) where the MOM difference is <= $1M.

Appendix

1. Code chunk for case 3

```
marketing_platforms = data.columns[0:]

# Create a Gurobi model
m = gp.Model()

# Decision Variables: Budget allocation for each marketing platform
x = m.addVars(marketing_platforms, name="budget_allocation", lb=0, ub=3000000)

# Define the objective function (maximize total ROI)
obj = gp.quicksum(data.iloc[0][platform] * x[platform] for platform in marketing_platforms)
m.setObjective(obj, sense=GRB.MAXIMIZE)

# Constraint 1: Total budget constraint
budget = 10000000
m.addConstr(gp.quicksum(x[platform] for platform in marketing_platforms) <= budget)

# Constraint 2: Budget limit for each platform
for platform in marketing_platforms:
    m.addConstr(x[platform] <= 3000000)

# Constraint 3: Print and TV constraint (update column names as per your data)
m.addConstr(x['TV'] + x['Print'] <= x['Facebook'] + x['Email'])

# Constraint 4: Social media constraint (update column names as per your data)
m.addConstr(x['Facebook'] + x['LinkedIn'] + x['Instagram'] + x['Snapchat'] + x['Twitter']
             >= 2 * (x['SEO'] + x['AdWords']))
m.Params.OutputFlag = 0
# Optimize the model
m.optimize()

# Display the results
if m.status == GRB.OPTIMAL:
    print("Optimal solution found")
    print(f"Total ROI: {m.objVal}")
    print("\nAllocation")
    allocation_df = pd.DataFrame({'Allocation': [x[platform].x for platform in marketing_platforms]},
                                 index=marketing_platforms)
    print(allocation_df)
else:
    print("No optimal solution found")
```

2. Code chunk for case 4

```
# Load the ROI data from the CSV file
data = pd.read_csv("ROI_data.csv").set_index("Platform")

# Define the marketing platforms
marketing_platforms = data.columns

# Create a Gurobi model for the second firm's ROI estimates
m_second_firm = gp.Model()

# Decision Variables: Budget allocation for each marketing platform
x_second_firm = m_second_firm.addVars(marketing_platforms, name="budget_allocation", lb=0, ub=3000000)

# Load the second firm's ROI estimates from the second row of the DataFrame
second_firm_roi = data.loc["Second Firms ROI Estimate"]

# Define the objective function (maximize total ROI) using the second firm's ROI estimates
obj_second_firm = gp.quicksum(second_firm_roi[platform] * x_second_firm[platform] for platform in marketing_platforms)
m_second_firm.setObjective(obj_second_firm, sense=GRB.MAXIMIZE)

# Constraint 1: Total budget constraint
budget = 10000000
m_second_firm.addConstr(gp.quicksum(x_second_firm[platform] for platform in marketing_platforms) <= budget)

# Constraint 2: Budget limit for each platform
for platform in marketing_platforms:
    m_second_firm.addConstr(x_second_firm[platform] <= 3000000)

# Constraint 3: Print and TV constraint (update column names as per your data)
m_second_firm.addConstr(x_second_firm['TV'] + x_second_firm['Print'] <= x_second_firm['Facebook'] + x_second_firm['Email'])

# Constraint 4: Social media constraint (update column names as per your data)
m_second_firm.addConstr(x_second_firm['Facebook'] + x_second_firm['LinkedIn'] + x_second_firm['Instagram'] + x_second_firm['Snapchat'] + x_second_firm['Twitter']
             >= 2 * (x_second_firm['SEO'] + x_second_firm['AdWords']))
m_second_firm.Params.OutputFlag = 0
# Optimize the model with the second firm's ROI estimates
m_second_firm.optimize()

# Display the results for the second firm
if m_second_firm.status == GRB.OPTIMAL:
    print("Optimal solution found for the second firm")
    print(f"Total ROI (Second Firm): {m_second_firm.objVal}")
    print("\nAllocation")
    allocation_df_second_firm = pd.DataFrame({'Allocation': [x_second_firm[platform].x for platform in marketing_platforms]},
                                 index=marketing_platforms)
    print(allocation_df_second_firm)
else:
    print("No optimal solution found for the second firm")
```

3. Code chunks for case 5

Removing 3M budget constraint for 1st one

```python
marketing_platforms = data.columns[0:]

# Create a Gurobi model
m1 = gp.Model()

# Decision Variables: Budget allocation for each marketing platform
x1 = m1.addVars(marketing_platforms, name="budget_allocation", lb=0)

# Define the objective function (maximize total ROI)
obj = gp.quicksum(data.iloc[0][platform] * x1[platform] for platform in marketing_platforms)
m1.setObjective(obj, sense=GRB.MAXIMIZE)

# Constraint 1: Total budget constraint
budget = 10000000
m1.addConstr(gp.quicksum(x1[platform] for platform in marketing_platforms) <= budget)

# Constraint 2: Print and TV constraint (update column names as per your data)
m1.addConstr(x1['TV'] + x1['Print'] <= x1['Facebook'] + x1['Email'])

# Constraint 3: Social media constraint (update column names as per your data)
m1.addConstr(x1['Facebook'] + x1['LinkedIn'] + x1['Instagram'] + x1['Snapchat'] + x1['Twitter']
             >= 2 * (x1['SEO'] + x1['AdWords']))
m1.Params.OutputFlag = 0
# Optimize the model
m1.optimize()

# Display the results
if m1.status == GRB.OPTIMAL:
    print("Optimal solution found")
    print(f"Total ROI: {m1.objVal}")
    print("\nAllocation")
    allocation_df = pd.DataFrame({'Allocation': [x1[platform].x for platform in marketing_platforms]},
                                 index=marketing_platforms)
    print(allocation_df)
else:
    print("No optimal solution found")
```

Removing 3M budget constraint for 2nd one

```python
# Define the marketing platforms
marketing_platforms = data.columns

# Create a Gurobi model for the second firm's ROI estimates
m_second_firm_2 = gp.Model()

# Decision Variables: Budget allocation for each marketing platform
x_second_firm_2 = m_second_firm_2.addVars(marketing_platforms, name="budget_allocation", lb=0)

# Load the second firm's ROI estimates from the second row of the DataFrame
second_firm_roi = data.loc["Second Firms ROI Estimate"]

# Define the objective function (maximize total ROI) using the second firm's ROI estimates
obj_second_firm = gp.quicksum(second_firm_roi[platform] * x_second_firm_2[platform] for platform in marketing_platforms)
m_second_firm_2.setObjective(obj_second_firm, sense=GRB.MAXIMIZE)

# Constraint 1: Total budget constraint
budget = 10000000
m_second_firm_2.addConstr(gp.quicksum(x_second_firm_2[platform] for platform in marketing_platforms) <= budget)

# Constraint 2: Print and TV constraint
m_second_firm_2.addConstr(x_second_firm_2['TV'] + x_second_firm_2['Print'] <= x_second_firm_2['Facebook'] + x_second_firm_2['Email'])

# Constraint 3: Social media constraint
m_second_firm_2.addConstr(x_second_firm_2['Facebook'] + x_second_firm_2['LinkedIn'] + x_second_firm_2['Instagram'] + x_second_firm_2['Snapchat'] + x_second_firm_2['Twitter']
             >= 2 * (x_second_firm_2['SEO'] + x_second_firm_2['AdWords']))
m_second_firm_2.Params.OutputFlag = 0
# Optimize the model with the second firm's ROI estimates
m_second_firm_2.optimize()

# Display the results for the second firm
if m_second_firm_2.status == GRB.OPTIMAL:
    print("Optimal solution found for the second firm")
    print(f"Total ROI (Second Firm): {m_second_firm_2.objVal}")
    print("\nAllocation")
    allocation_df_second_firm = pd.DataFrame({'Allocation': [x_second_firm_2[platform].x for platform in marketing_platforms]},
                              index=marketing_platforms)
    print(allocation_df_second_firm)
else:
    print("No optimal solution found for the second firm")
```

4. Code chunk for case 6

```
upper_bound=m.SAObjUp
lower_bound=m.SAObjLow
index_channels=['Print','TV','SEO','Adwords','Facebook','LinkedIn','Instagram','Snapchat','Twitter','Email']
bounds=pd.DataFrame({'Lower Bounds':lower_bound,'Upper Bounds':upper_bound},index=index_channels)
display(bounds)
```

5. Code chunk for case 7

```
budget = 10
monthly = []
budgets_monthly = []
returns_monthly = []
for i in range(12):
    ojMod=gp.Model()
    budgets_monthly.append(budget)
    ojModX = ojMod.addMVar(10,ub=3)
    objcoeff = list(df.iloc[i,1:])
    ojMod.setObjective(gp.quicksum(ojModX[j]*(objcoeff[j]/100) for j in range(10)),sense=gp.GRB.MAXIMIZE)
    conlist=[0]*3
    conlist[0] = ojMod.addConstr(gp.quicksum(ojModX[i] for i in range(10)) <= budget)
    conlist[1] = ojMod.addConstr(ojModX[0]+ojModX[1]-ojModX[4]-ojModX[9] <= 0)
    conlist[2] = ojMod.addConstr(ojModX[4]+ojModX[5]+ojModX[6]+ojModX[7]+ojModX[8]-2*ojModX[2]-2*ojModX[3] >= 0)
    ojMod.Params.OutputFlag = 0
    ojMod.optimize()
    # Calculate the returns for this month
    returns = sum(objcoeff[j] / 100 * ojModX[j].x for j in range(10))
    monthly.append(list(ojModX.x)+ [returns])

    budget = 10
    budget += 0.5 * returns

    # Append the returns for this month to the list of monthly returns
    returns_monthly.append(returns)

columns = ['Print', 'TV', 'SEO', 'AdWords', 'Facebook', 'LinkedIn', 'Instagram', 'Snapchat', 'Twitter', 'Email', 'Returns']
df_results = pd.DataFrame(monthly, columns=columns)
df_results['Budgets'] = budgets_monthly
df_results['Month'] = df.iloc[:, 0]
df_results.set_index('Month', inplace=True)


# Display the results
print("Monthly Budget Allocations and Returns:")
display(df_results)
```

6. Code chunk for case 7 - calculating differences

```
# Calculate the differences in budget allocation between consecutive months
differences = [list(-df_results.iloc[i - 1, :-2] + df_results.iloc[i, :-2]) for i in range(1, 12)]

# Create a DataFrame to store the differences
df_differences = pd.DataFrame(differences, columns=df_results.columns[:-2])
```