Optimizing Marketing Advertising Budget using Linear Programming

Objective

The problem statement is to recommend the marketing department an optimal budget allocation strategy for different media platforms. Monthly budget available is $10M which can be spent on 10 different media platforms that can have different Return on Investment (ROI). The final allocation to be recommended should maximize the overall return on investment every month. To arrive at the allocation ROI estimates from two different firms are available for each platform as below -

Platform	Print	TV	SEO	AdWords	Facebook	LinkedIn	Instagram	Snapchat	Twitter	Email
First Firm's ROI estimates	0.031	0.049	0.024	0.039	0.016	0.024	0.046	0.026	0.033	0.044
Second Firm's ROI estimate	0.049	0.023	0.024	0.039	0.044	0.046	0.026	0.019	0.037	0.026

The manager, along with the ROI estimates, has added a few constraints on the budget allocation based on experience that needs to honor while recommending budget allocations. The three constraints imposed by the manager are below –

1. The amount invested in print and TV should be no more than the amount spent on Facebook and Email. Surprisingly, email seems to be a great channel for reaching real people.
2. The total amount used in social media (Facebook, LinkedIn, Instagram, Snapchat, and Twitter) should be at least twice that of SEO and AdWords.
3. For each platform, the amount invested should be no more than $3M.

Mathematical Formulation

The business problems will be mathematically formulated as a linear optimization problem with constraints (set by the manager) before programming happens *(code – Appendix A.1)*.

The *fraction of total budget allocated to each platform* will be represented as x_i, with different i's representing different platforms from Print to Email. On top of the three constraints provided by the manager, an additional constraint will be required to consider the fact that the overall budget for allocation is $10M. Since the variables considered in the formulation are fractions of the budget and not actual budget, the constraint applicable will be that the sum of all fractions for all the platforms should be 1. The constraint equations and overall ROI function are below –

Assuming variables x_0 to x_9 to be the fraction of total budget for different platforms - from Print to Email in the above csv

Constraint 1:

$$Total\,Budget * (x_0 + x_1) - Total\,Budget * (x_4 + x_9) <= 0$$

Constraint 2:

$$Total\,Budget * (x_4 + x_5 + x_6 + x_7 + x_8) - 2 * Total\,Budget * (x_2 + x_3) <= 0$$

Constraint 3 (for all platfroms):

$$Total\,Budget * x_i <= 3$$
$$\forall i \in [0, 9]$$

Constrain 4 - sum of fractions to be one:

$$\Sigma x_i = 1$$
$$\forall i \in [0, 9]$$

Assuming the ROI of each platform as c_i for $i \in [0, 9]$ We try to maximise the overall ROI from differnet platforms based on a budget allocation, which can be represented as below -

$$z = Total\,Budget * \Sigma(x_i * c_i)$$

Based on the above-mentioned set of equations - constraint matrix A, bounding vector b, and ROI vector c is defined and solved via Gurobi *(code – Appendix A.2)* for the two distinct set of estimates provided.

Optimal Allocation Results

Using linear optimization, optimal allocations are obtained using both firm's estimates *(code – Appendix A.3)*. The allocation for different platforms varies for the two firms but the overall ROI that can be obtained is the same. The expected total ROI comes to be 4.56% giving a return of $456K for $10M marketing budget. Although the budget allocation of different platforms varies. The two allocations fundamentally don't vary between online platform - one giving more value to Facebook & LinkedIn while the other to Instagram & Email. They also vary between traditional platform TV vs Print. Though the allocation for both firms is the same for AdWords. At a higher level we can say both allocations emphasize Digital Media more than Traditional platform, with $6M going to the former and $3M allocated to latter.

Budget allocation and optimal ROI for First firm's vs Second firm's estimates

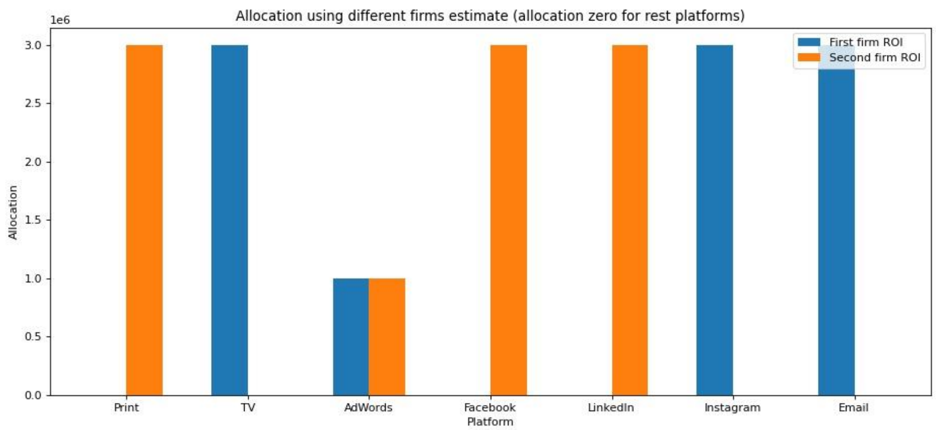
The following questions were posed to understand the allocation and overall ROI differences when using the two different platform's ROI estimates –

1. Are the allocations the same?

   No allocations are not the same. Non-zero budget is allocated to only 4 platforms in both cases.

   Only AdWords allocation is the same in both cases - $1M monthly budget. It can be noted that both firms estimate AdWords' ROI is 3.9%, identical allocation is not happening due to this. As the ROI of one platform can be the same for both firms, it may be possible that in one firm's estimates that platform is ranked different in the order of ROI. But in the shared estimate for both firms, AdWords ROI is not just same, it is also ranked the same i.e., 4^{th} in decreasing order.

While for the rest of the three platforms with non-zero budget, there is no similarity in the two allocations as can be seen in the graph below. While first firm values TV, Instagram, and Email more, the second firm expects to get more ROI from Print, Facebook & LinkedIn.



2. 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)?

   Optimal ROI is reduced by $204,000 when allocation happens using second firm's estimate, but the observed monthly ROI is same as first ROI Data

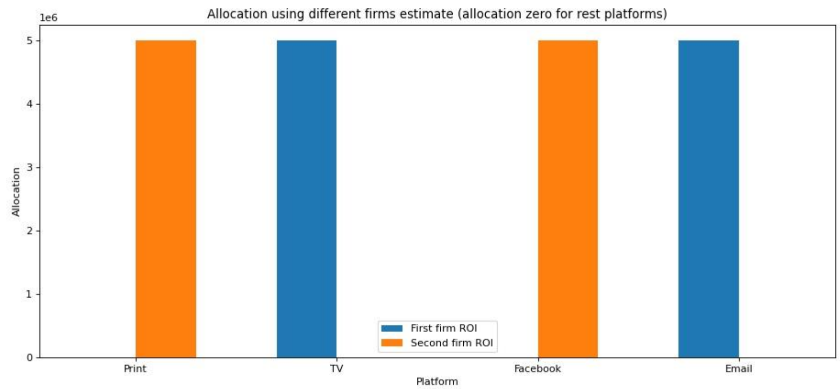
3. Assuming the second ROI data is correct, if you used the first allocation how much lower would the objective be relative to the optimal objective?

   Optimal ROI is reduced by $192,000 when allocation happens using second firm's estimate, but the observed monthly ROI is same as first ROI Data

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

   To understand the effects of third constraint on overall allocation and ROI, an experiment is run. In the experiment the third constraint – individual budget for each platform should not increase $3M, will be removed. Allocation and ROI thus obtained will be compared with the results with keeping the third constraint intact *(code – Appendix A.4)*.

   The overall ROI obtained after removing the constraint is *4.65% giving a return of $465K* for both the estimates, giving $9K more than the allocation with the constraint intact. Platform wise allocations are below –



Given the uncertainty in ROI estimates and the objective of maintaining presence over various platforms it makes sense to put the third constraints imposed by the boss. If that constraint is removed, all the budgets can be allocated to the platform which gives the highest ROI. If we remove that constraint, for first ROI we get TV & Email as 50-50%, and for second its Print & Facebook. This kind of allocation reduces the presence over different platforms that is ensured by the constraint imposed by the manager. Also, the incremental value obtained after removing the constraint is not significant enough to diminish the presence over multiple platforms.

Sensitivity Analysis

It is impossible to accurately predict the ROI from investment made in each platform beforehand hence the marketing department has employed two different firms to obtain ROI estimates. Since the estimates can vary for each platform, it is essential to understand the stability of overall ROI if there are any deviations from the estimates.

Sensitivity analysis will be done in two ways – 1. Manual approach, 2. Via Gurobi

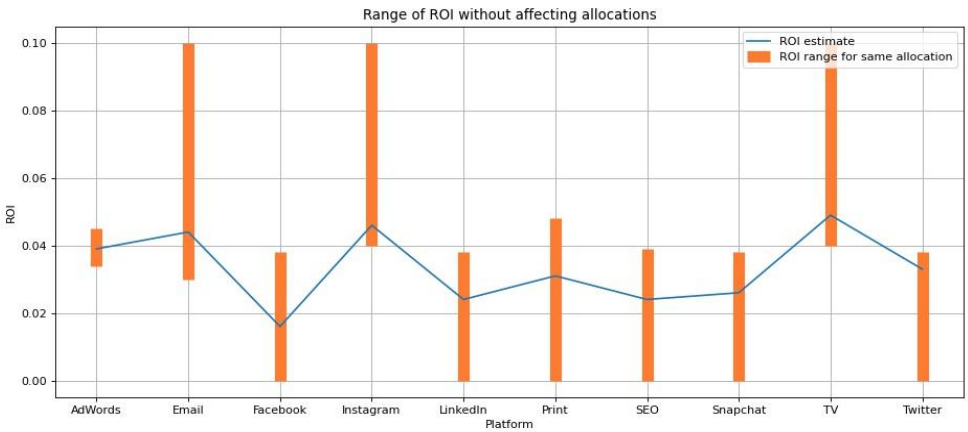
Manual Approach

How to perform sensitivity analysis *(code – Appendix A.5)* -

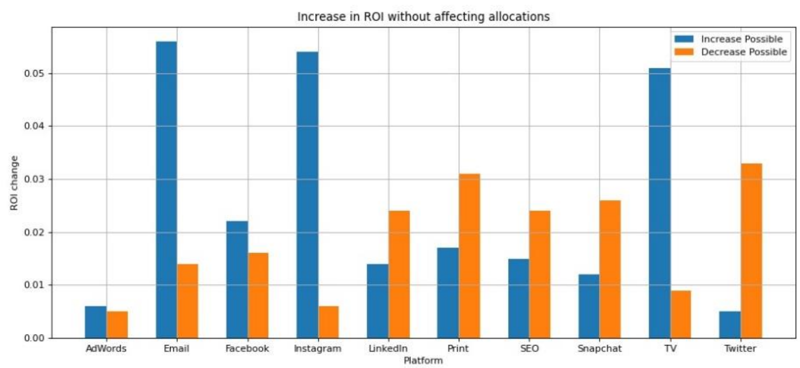
1. ROI will be *increase for one platform by 0.1%* without changing it for the other platforms
2. Optimal Allocation will be recorded to ensure that there is no change in overall allocation
3. This iterative process will be put to a halt once the allocation changes
4. Once done for one platform repeat the same for other platforms keeping the ROI of the rest the same
5. Once steps 1-4 are done, same steps are repeating except the ROI now will be *decreased for the platform by 0.1%*

Both increase and decrease in the ROI are limited to 0% and 10% respectively. These constraints are put in place due to two reasons. Firstly, we have assumed that we cannot have a negative ROI, as marketing budgets can be considered as sunk cost and the objective is to get the return on that investment. Secondly, since the manual approach is iterative, without putting any bounds on the expected ROI the iterations can continue to the extreme ends of positive and negative.

Below is the figure to compare the acceptable range of ROI for each platform without changing the overall allocation. The figure illustrates the acceptable range of ROI with respect to the first firm's estimates.



The increase and decrease possible from ROI without changing the allocation can be observed below -



The change in ROI possible (decrease/increase) plotted in the above plot is restricted based on the possible ROI assumed – 0% to 10%.

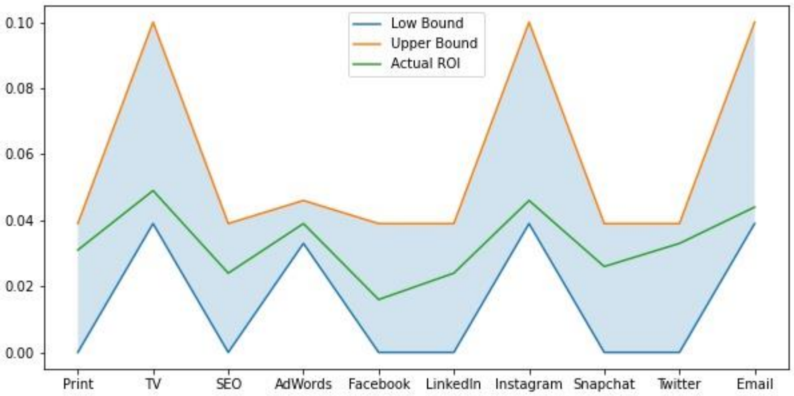
Conclusion -

1. For Email, Instagram & TV no matter how much we increase the ROI, allocation can be the same (though ROI limit has been set to 10%, upon increasing the limit we still see no change)
2. For Facebook, LinkedIn, Print, SEO, Snapchat & Twitter no matter how much we decrease the ROI, allocation can still be the same (assuming ROI cannot be less than 0%)
3. Only AdWords is sensitive to any increase or decrease in ROI estimate. In both cases if the change is significant the budget allocation can change

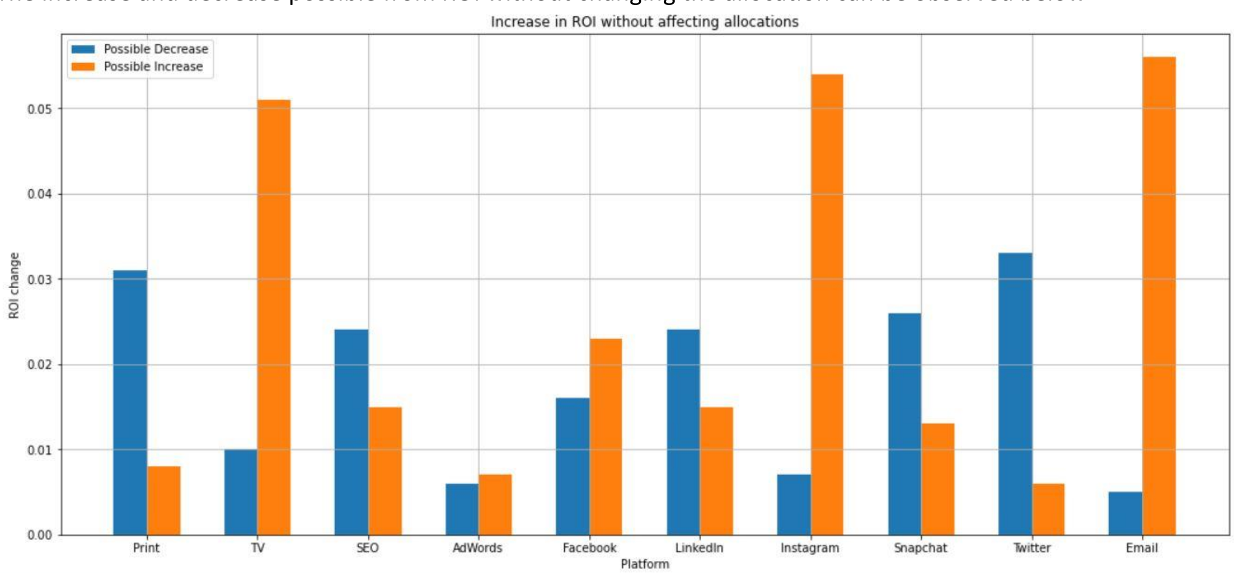
Via Gurobi

Gurobi provides a functionality to derive Upper and Lower bound for coefficients keeping the optimal value same. Through Gurobi, we get the sensitivity analysis for each platform's ROI while keeping other ROI's constant.

The below graph shows the Upper and Lower bound for the given ROI value while keeping the optimal value same.



The increase and decrease possible from ROI without changing the allocation can be observed below



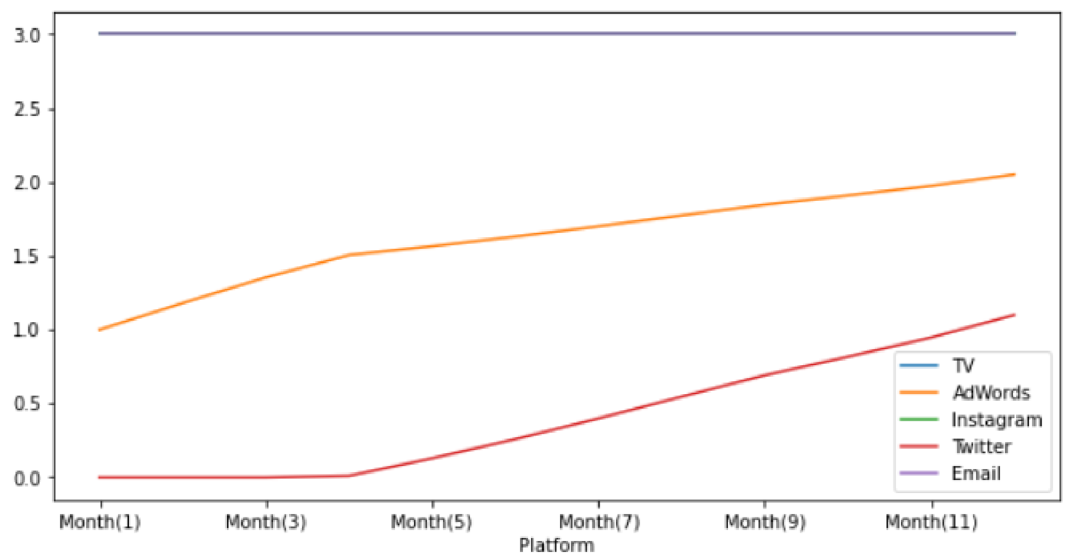
The Manual and via Gurobi sensitivity analysis give almost the same results. In Gurobi, we have considered 0 as the min value for ROI and 0.5 as the maximum for visual representation

Stable Budget

Assume that our company is deciding the optimal monthly budget allocation based on the first ROI data. And now we have actual monthly ROI for next year and the permission to reinvest half of monthly return earned. Thus, next year our marketing budget would start with \$10M and increase by ($Budget_{last\_month}$ * ROI_{last_month} * 50%) per month *(code – Appendix A.7)*. Based on our analysis, the optimal allocation for each month is shown as below.

Unnamed: 0	Print	TV	SEO	AdWords	Facebook	LinkedIn	Instagram	Snapchat	Twitter	Email
January	0.0	3.0	0.0	1.000000	0.0	0.0	3.0	0.0	0.000000	3.0
February	0.0	3.0	0.0	1.180000	0.0	0.0	3.0	0.0	0.000000	3.0
March	0.0	3.0	0.0	1.353920	0.0	0.0	3.0	0.0	0.000000	3.0
April	0.0	3.0	0.0	1.505381	0.0	0.0	3.0	0.0	0.010763	3.0
May	0.0	3.0	0.0	1.564986	0.0	0.0	3.0	0.0	0.129971	3.0
June	0.0	3.0	0.0	1.629829	0.0	0.0	3.0	0.0	0.259657	3.0
July	0.0	3.0	0.0	1.699709	0.0	0.0	3.0	0.0	0.399418	3.0
August	0.0	3.0	0.0	1.773336	0.0	0.0	3.0	0.0	0.546672	3.0
September	0.0	3.0	0.0	1.845758	0.0	0.0	3.0	0.0	0.691517	3.0
October	0.0	3.0	0.0	1.908289	0.0	0.0	3.0	0.0	0.816578	3.0
November	0.0	3.0	0.0	1.973321	0.0	0.0	3.0	0.0	0.946643	3.0
December	0.0	3.0	0.0	2.049697	0.0	0.0	3.0	0.0	1.099393	3.0

A stable budget is defined as a monthly allocation such that for each platform the monthly change in spend is no more than $1M. Based on our analysis, next year our company would invest on TV, AdWords, Instagram, Twitter, and Email. Regarding TV, Instagram, and Email, the amount invested on those platforms would be $3M throughout the year. Regarding AdWords and Twitter, the amount invested on those platforms would be increased but no more than $1M per month. Therefore, the optimal allocation for each month next year is a stable budget.



On the other hand, in case we might get an unstable budget, we can consider adding additional constraints in our model. For each platform, the amount invested should not be $1M more than that of last month, and $1M less than that of last month.

Appendix

A.1 Building the matrix for constraints

```
# initializing constraint matrix and constraint limit and directions
A = np.zeros((13, 10))
b = np.zeros(13)
sense = ['']*13

# Constraint 1:
# total_budget*(x_0 + x_1) - total_budget*(x_4 + x_9) <= 0
A[0, 0] = 1
A[0, 1] = 1
A[0, 4] = -1
A[0, 9] = -1
b[0] = 0
sense[0] = '<'

# Constraint 2:
# total_budget*(x_4 + x_5 + x_6 + x_7 + x_8) - 2*total_budget*(x_2 + x_3) >= 0
A[1, 4:9] = 1
A[1, 2:4] = -2
b[1] = 0
sense[1] = '>'
```

```
# Constraint 3 to 12:
# x_i <= 0.3
for i in range(10):
    A[i+2, i] = 1
    b[i+2] = 0.3
    sense[i+2] = '<'

# Constraint 13:
# sum of all x_i <= 1
A[12, :] = 1
b[12] = 1
sense[12] = '<'

A, sense, b
```

A.2 Function for creating the optimization problem

```
def create_model(A, sense, b, obj, opt=gp.GRB.MAXIMIZE):
    # creating model
    model = gp.Model()

    # creating variable an setting the constraints
    modx = model.addMVar(A.shape[1])
    mod_con = model.addMConstrs(A, modx, sense, b)

    # setting the objective function
    model.setMObjective(None, obj, 0, sense=opt)

    # restricting gurobi logs
    model.Params.OutputFlag = 0

    # optimizing the function
    model.optimize()

    return model
```

A.3 Getting results for both firm's estimates

A) First ROI

```
# creating an output df with different firms ROI for Platform allocation
output_df = roi_df.T.reset_index()
output_df.columns = output_df.iloc[0]
output_df = output_df.iloc[1:]

# creating new column in output df for budget allocation
output_df['Allocation'] = adv_model.x

first_firms_roi_pct = round(adv_model.objVal*100,3)
first_firms_roi = round(adv_model.objVal*total_budget)
print('Total ROI from ${} marketing budget is {} dollars which is {}%'.format(
    total_budget, first_firms_roi, first_firms_roi_pct))
```

```
Total ROI from $10000000 marketing budget is 456000 dollars which is 4.56%
```

B) Second ROI

```
# creating advertising model for second estimates
adv_model2 = create_model(A, sense, b, obj=second_roi, opt=gp.GRB.MAXIMIZE)
# creating new column in output df for budget allocation
output_df['Second Allocation'] = adv_model2.x

second_firms_roi_pct = round(adv_model2.objVal*100,3)
second_firms_roi = round(adv_model2.objVal*total_budget)
print('Total ROI from ${} marketing budget is {} dollars which is {}%'.format(
    total_budget, second_firms_roi, second_firms_roi_pct))
```

```
Total ROI from $10000000 marketing budget is 456000 dollars which is 4.56%
```

A.4 Removing third constraint

```
# updating constraint limit vector
b_unconstrained = b.copy()
b_unconstrained[2:13] = 1

# creating the two models
adv_model_unconstrained1 = create_model(A, sense, b_unconstrained, obj=roi, opt=gp.GRB.MAXIMIZE)
adv_model_unconstrained2 = create_model(A, sense, b_unconstrained, obj=second_roi, opt=gp.GRB.MAXIMIZE)

# creating an output df with different firms ROI for Platform allocation
output_unconstrained_df = roi_df.T.reset_index()
output_unconstrained_df.columns = output_unconstrained_df.iloc[0]
output_unconstrained_df = output_unconstrained_df.iloc[1:]

# creating new column in output df for budget allocation
output_unconstrained_df['Allocation'] = adv_model_unconstrained1.x
output_unconstrained_df['Second Allocation'] = adv_model_unconstrained2.x

# overall ROI after removing the third constraint
first_firms_unconstrained_roi_pct = round(adv_model_unconstrained1.objVal*100,3)
first_firms_unconstrained_roi = round(adv_model_unconstrained1.objVal*total_budget)
print('Total ROI from ${} marketing budget is {} dollars which is {}%'.format(
    total_budget, first_firms_unconstrained_roi, first_firms_unconstrained_roi_pct))

second_firms_unconstrained_roi_pct = round(adv_model_unconstrained2.objVal*100,3)
second_firms_unconstrained_roi = round(adv_model_unconstrained2.objVal*total_budget)
print('Total ROI from ${} marketing budget is {} dollars which is {}%'.format(
    total_budget, second_firms_unconstrained_roi, second_firms_unconstrained_roi_pct))
```

```
Total ROI from $10000000 marketing budget is 465000 dollars which is 4.65%
Total ROI from $10000000 marketing budget is 465000 dollars which is 4.65%
```

A.5 Sensitivity Analysis – Manual Approach

```python
# getting first allocation vector for comparision
allocation1 = output_df['Allocation'].values
# creating senstivity df
sensitivity1 = pd.DataFrame(data=None, columns=['Platform', 'ROI', 'Total ROI', 'Allocation'])
# looping over for getting optimal allocation
for i in range(1, roi_df.shape[1]):
    # platform name
    platform = roi_df.columns[i]
    platform_roi = roi_df[platform].values[0]
    # looping over for range in ROI
    # print(platform, platform_roi)
    s_allocation = allocation1.copy()
    new_roi = roi.copy()
    # for decreasing the platform ROI
    while (np.array_equal(allocation1, s_allocation) & (new_roi[i-1] > 0)):
        new_roi[i-1] -= 0.001
        s_model = create_model(A, sense, b, obj=new_roi, opt=gp.GRB.MAXIMIZE)
        s_allocation = s_model.x
        if np.array_equal(allocation1, s_allocation):
            row_len = len(sensitivity1)
            sensitivity1.loc[row_len, 'Platform'] = platform
            sensitivity1.loc[row_len, 'ROI'] = new_roi[i-1]
            sensitivity1.loc[row_len, 'Total ROI'] = s_model.objVal
            sensitivity1.loc[row_len, 'Allocation'] = str(s_allocation)
    # for increasing the platform ROI
    s_allocation = allocation1.copy()
    new_roi = roi.copy()
    while (np.array_equal(allocation1, s_allocation) & (new_roi[i-1] < 0.1)):
        new_roi[i-1] += 0.001
        s_model = create_model(A, sense, b, obj=new_roi, opt=gp.GRB.MAXIMIZE)
        s_allocation = s_model.x
        if np.array_equal(allocation1, s_allocation):
            row_len = len(sensitivity1)
            sensitivity1.loc[row_len, 'Platform'] = platform
            sensitivity1.loc[row_len, 'ROI'] = new_roi[i-1]
            sensitivity1.loc[row_len, 'Total ROI'] = s_model.objVal
            sensitivity1.loc[row_len, 'Allocation'] = str(s_allocation)
```

A.6 Sensitivity Analysis – via Gurobi

```python
sg_model = create_model(A, sense, b, obj=new_roi, opt=gp.GRB.MAXIMIZE)

low_bound = sg_model.SAObjLow
up_bound = sg_model.SAObjUp

labels = ['Print', 'TV', 'SEO', 'AdWords', 'Facebook', 'LinkedIn',
       'Instagram', 'Snapchat', 'Twitter', 'Email']

l1 = [0 if (x==-np.inf) else x for x in low_bound]
l2 = [0.1 if (x==np.inf) else x for x in up_bound]

plt.figure(figsize=(10,5))
plt.plot(labels,l1,label='Low Bound')
plt.plot(labels,l2,label='Upper Bound')
plt.plot(labels,roi,label='Actual ROI')
plt.fill_between(labels,l1,l2,alpha=0.2)
plt.legend()
plt.savefig('figure/sensitivity_range_using_gurobi.jpeg')
plt.show()
```

A.7 Stable Budget & Monthly allocation

```python
# creating monthly budget var to be updated every month
monthly_budget = [0]*13
monthly_budget[0] = total_budget
months = np.arange(1, 13)
month_name = [calendar.month_name[m] + ' ' + 'Allocation' for m in months]

# reading actual roi monthly data
actual_roi = pd.read_csv('roi_mat.csv')

# creating output df
monthly_allocation_df = pd.DataFrame(data=None, columns=(['Platform'] + month_name))
monthly_allocation_df['Platform'] = output_df['Platform']
```

```python
# calculating allocation for every month
for m in months:

    # updating constraint for varying monthly budget
    b_monthly = b.copy()
    for i in range(10):
        b_monthly[i+2] = 0.3*total_budget/monthly_budget[m-1]

    # creating the model and storing in output dataframe
    adv_month_model = create_model(A, sense, b_monthly, obj=roi, opt=gp.GRB.MAXIMIZE)
    monthly_allocation_df.iloc[:, m] = adv_month_model.x
    monthly_allocation_df.iloc[:, m] =  monthly_allocation_df.iloc[:, m] * monthly_budget[m-1]

    # updating budget
    additional_budget = sum(monthly_allocation_df.iloc[:, m] * actual_roi.T.iloc[1:, m-1].values)/100 * 0.5
    monthly_budget[m] = monthly_budget[m-1] + additional_budget
```

```python
# plot size
plt.figure(figsize=(14, 6), dpi=80)

# plotting each months allocation
for i in months:
    plt.plot(monthly_allocation_df.iloc[:, 0].values, monthly_allocation_df.iloc[:, i].values,
             label=monthly_allocation_df.columns[i])

# adding title and axis names
plt.xlabel('Platform')
plt.ylabel('Monthly ROI')
plt.title('M-o-M change in budget allocation')
plt.legend()
plt.savefig('figure/mom_allocation_change.jpeg')
plt.show()
```