ALLO HEALTH ASSIGNMENT

INSIGHT GENERATION

Dissertation submitted in fulfilment of the requirements for the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

K. Shyam

12111001

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab

DECLARATION STATEMENT

I hereby declare that the research work reported in the dissertation/dissertation proposal entitled "**ALLO HEALTH ASSIGNMENT"** in partial fulfilment of the requirement for the award of Degree for Master of Technology in Computer Science and Engineering at Lovely Professional University, Phagwara, Punjab. I have not submitted this work elsewhere for any degree or diploma.

I understand that the work presented herewith is in direct compliance with Lovely Professional University's Policy on plagiarism, intellectual property rights, and highest standards of moral and ethical conduct. Therefore, to the best of my knowledge, the content of this dissertation represents authentic and honest research effort conducted, in its entirety, by me. I am fully responsible for the contents of my dissertation work.

Signature of Candidate

K. Shyam

R.No 12111001

INTRODUCTION:

Insight generation is a critical phase in data analysis, where we translate raw data into actionable recommendations. This project delves into the performance of various advertising campaigns and ad sets on Google and Facebook, aiming to uncover opportunities for scaling, identify underperforming areas, suggest rationalization strategies, and assess predictability. By thoroughly analyzing key performance metrics and benchmark comparisons, we can provide strategic insights that help optimize marketing efforts, improve ROI, and drive business growth.

METHODOLOGY:

1.Opportunities for scaling:

Get the campaigns and AD sets where the performance meets/exceeds benchmarks.

```
# Filter rows where any evaluation for Google or Facebook is 'Meets/Exceeds'
scaling_opportunities_google = google_performance[(google_performance[[col for col in google_performance.columns if 'evaluation' in col]].apply(lambda x: any(x == 'Meets/Exceeds'), axis=1))]
scaling_opportunities_facebook = facebook_performance[(facebook_performance[[col for col in facebook_performance.columns if 'evaluation' in col]].apply(lambda x: any(x == 'Meets/Exceeds'), axis=1))]

# Concatenate the scaling opportunities for Google and Facebook
scaling_opportunities = pd.concat([scaling_opportunities_google, scaling_opportunities_facebook])

# Extract only the columns where performance meets or exceeds benchmarks
meets_exceeds_columns = scaling_opportunities.filter(regex='_evaluation').apply(lambda x: x.str.contains('Meets/Exceeds'))

# Display only the meets/exceeds columns
print("Columns where Performance Meets/Exceeds Benchmarks:")
print(meets_exceeds_columns)
```

Output:

```
Columns where Performance Meets/Exceeds Benchmarks:
   CTR_google_evaluation Traffic to Lead_google_evaluation  \
0                  False                             False
1                  False                             False
2                  False                             False
3                  False                             False
4                  False                             False
5                  False                             False
6                    NaN                               NaN
7                    NaN                               NaN
8                    NaN                               NaN
9                    NaN                               NaN
10                   NaN                               NaN
11                   NaN                               NaN
12                   NaN                               NaN
13                   NaN                               NaN
14                   NaN                               NaN

   Lead to Call_google_evaluation CTR_facebook_evaluation  \
0                            True                     NaN
1                            True                     NaN
2                            True                     NaN
3                            True                     NaN
4                            True                     NaN
5                            True                     NaN
6                             NaN                    True
7                             NaN                    True
8                             NaN                    True
9                             NaN                    True
10                            NaN                    True
11                            NaN                    True
12                            NaN                    True
13                            NaN                    True
14                            NaN                    True
```

```
   Traffic to Lead_facebook_evaluation Lead to Call_facebook_evaluation
0                                  NaN                              NaN
1                                  NaN                              NaN
2                                  NaN                              NaN
3                                  NaN                              NaN
4                                  NaN                              NaN
5                                  NaN                              NaN
6                                False                             True
7                                False                             True
8                                False                             True
9                                False                             True
10                               False                             True
11                               False                             True
12                               False                             True
13                               False                             True
14                               False                             True
```

2.Issues to solve:

Get the campaigns and AD sets where the performance Short Falls benchmarks

```
# Filter columns where performance falls short of benchmarks
falls_short_columns = scaling_opportunities.filter(regex='_evaluation').apply(lambda x: x.str.contains('Falls Short'))


# Display only the falls short columns
print("Columns where Performance Falls Short of Benchmarks:")
print(falls_short_columns)
```

Output:

```
Columns where Performance Falls Short of Benchmarks:
   CTR_google_evaluation Traffic to Lead_google_evaluation  \
0                   True                              True
1                   True                              True
2                   True                              True
3                   True                              True
4                   True                              True
5                   True                              True
6                    NaN                               NaN
7                    NaN                               NaN
8                    NaN                               NaN
9                    NaN                               NaN
10                   NaN                               NaN
11                   NaN                               NaN
12                   NaN                               NaN
13                   NaN                               NaN
14                   NaN                               NaN

   Lead to Call_google_evaluation CTR_facebook_evaluation  \
0                           False                     NaN
1                           False                     NaN
2                           False                     NaN
3                           False                     NaN
4                           False                     NaN
5                           False                     NaN
6                             NaN                   False
7                             NaN                   False
8                             NaN                   False
9                             NaN                   False
10                            NaN                   False
11                            NaN                   False
12                            NaN                   False
13                            NaN                   False
14                            NaN                   False
```

	Traffic to Lead_facebook_evaluation	Lead to Call_facebook_evaluation
0	NaN	NaN
1	NaN	NaN
2	NaN	NaN
3	NaN	NaN
4	NaN	NaN
5	NaN	NaN
6	True	False
7	True	False
8	True	False
9	True	False
10	True	False
11	True	False
12	True	False
13	True	False
14	True	False

potential reasons:

1.CTR_google:

-This column not reaching the threshold value because

* AD is not attractive to the customer

* May be crews in the AD is not attractive to the customer

2.Traffic to Lead_google:

-This column not reaching the threshold value because

*May be login interface is too long to fill.

*The benefits of website is not clear to the customer

3.Traffic to Lead_facebook:

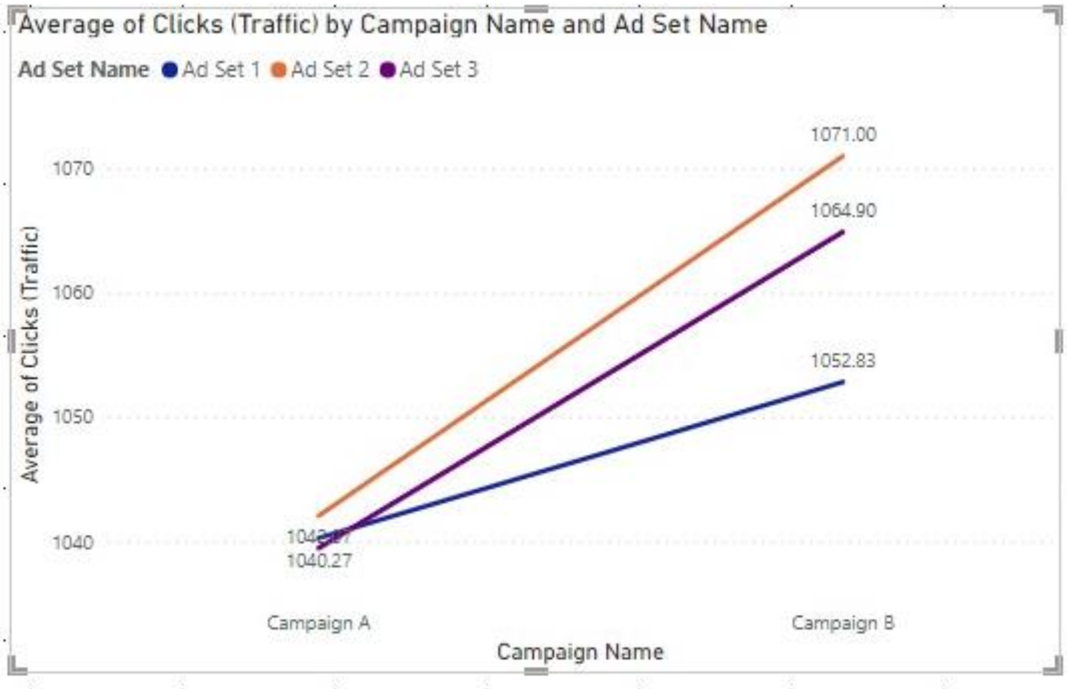
-This also having the same reasons in Traffic to Lead_google for not reaching the threshold value

3.Strategies to rationalise:

The decisions for any campaigns/ad sets that should be reduced or discontinued is taken carefully by visualizing the data .

Google data:

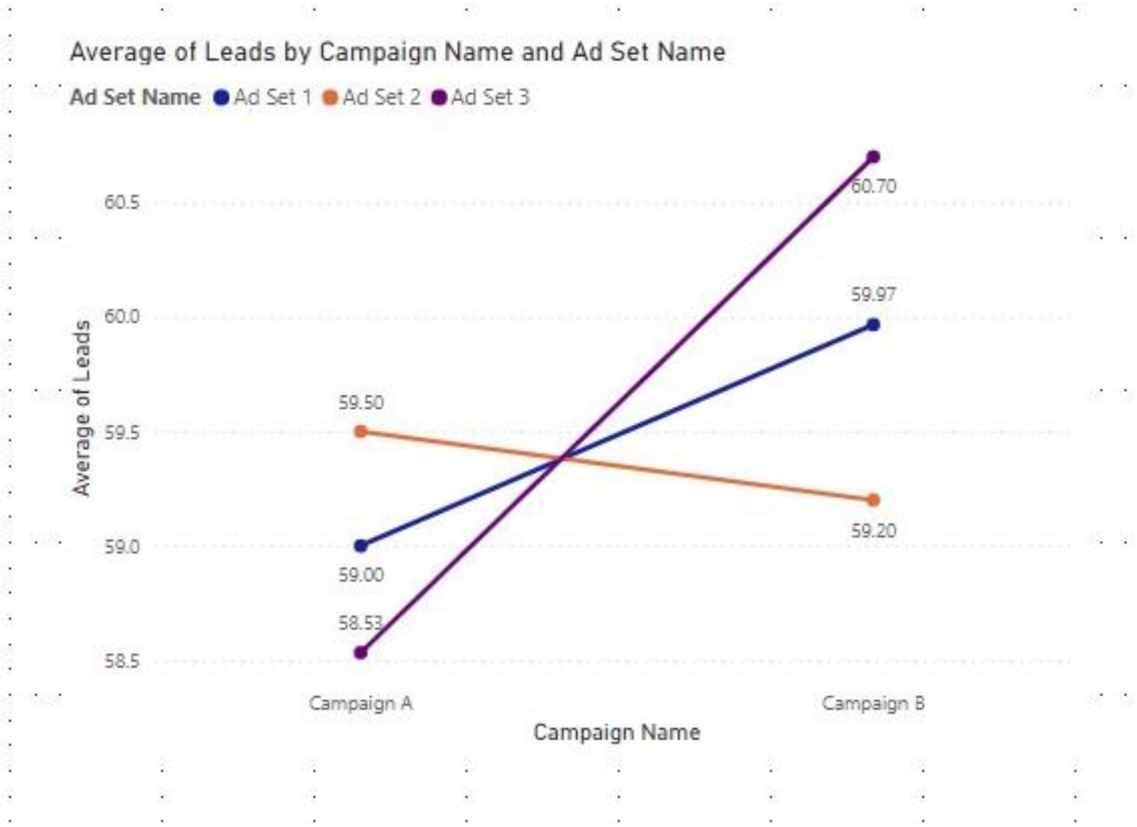
I done some visualizations in power BI . To understand the google data which campaign/AD set is best



-The first step after seeing the AD is to click the AD and go to the website

From the above graph we find that the campaign B is more attractive to the customers .In campaign B the set 2 is more good compare to the remaining

-After click the AD then leads is very important to analyze

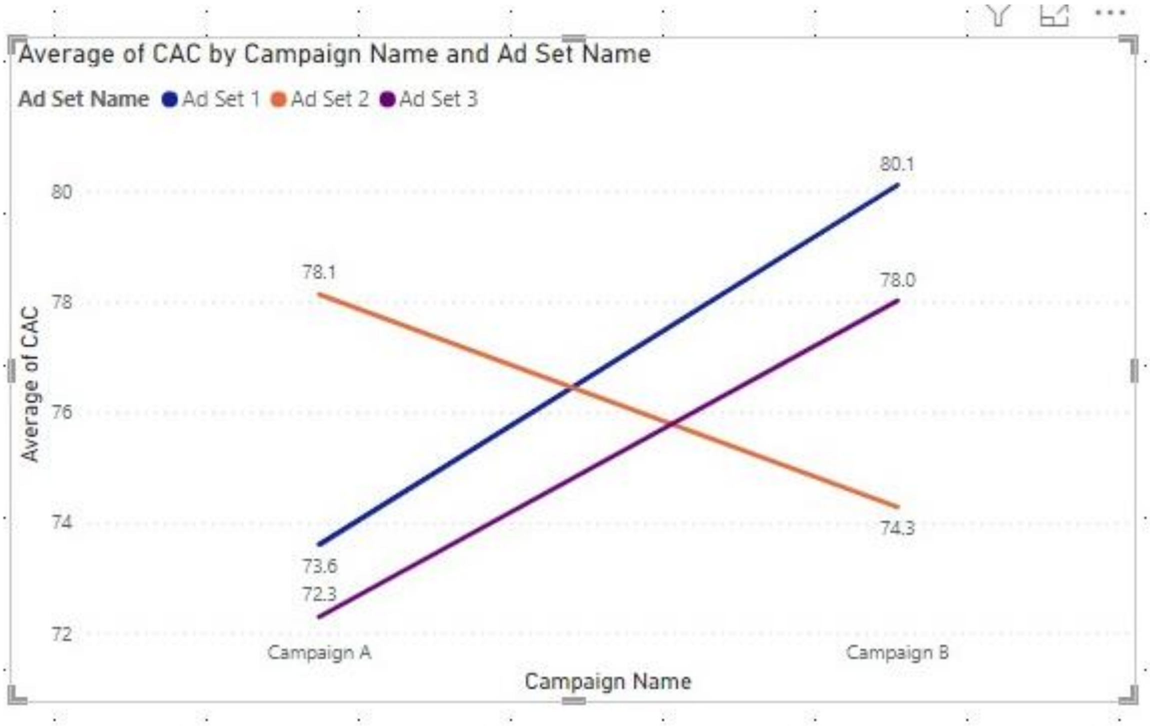


-From this graph we find the Ad set3 and Ad set1is performing well in campaign B

-From the fist graph the set2 is performing well and coming to this graph Ad set2 is good compared to the set3 and set1

-May be the ad is not fully transparent and the interface may not be liked by customer

-final conclude the set1 and set3 in campaign A is not good in both graphs



-From this graph we find how cost effectively invested

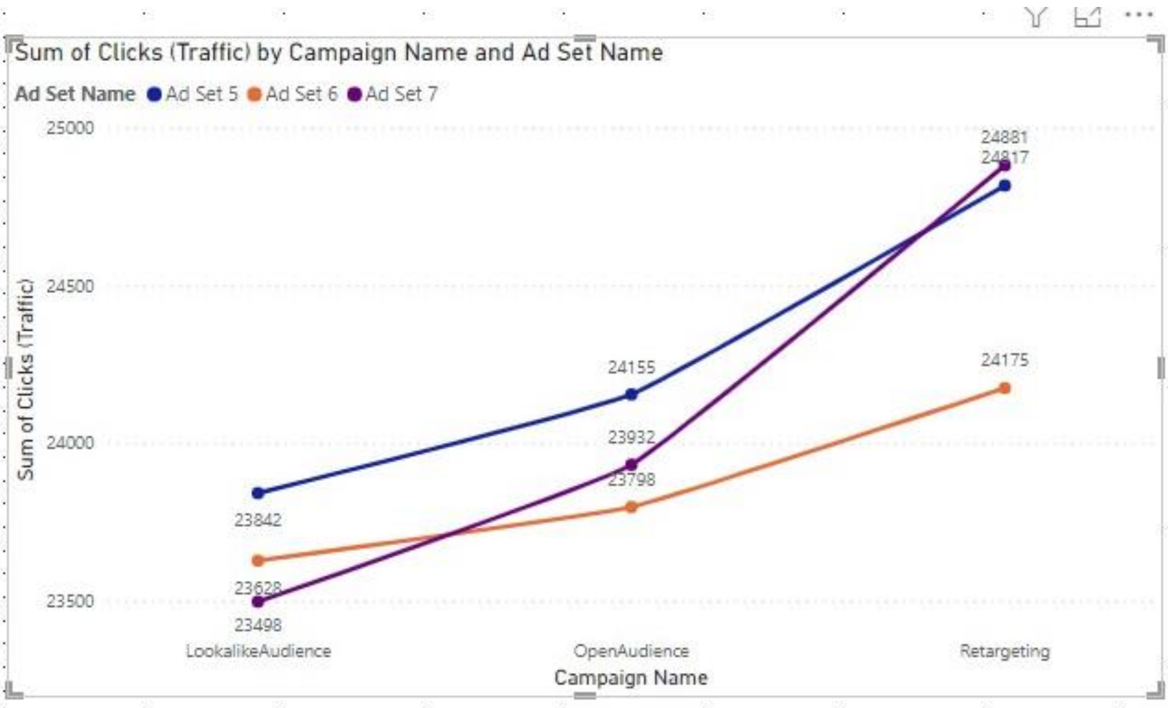
-From this graph also the set1 and set3 are not performing well in campaign A

<u>Final conclude for google data:</u>

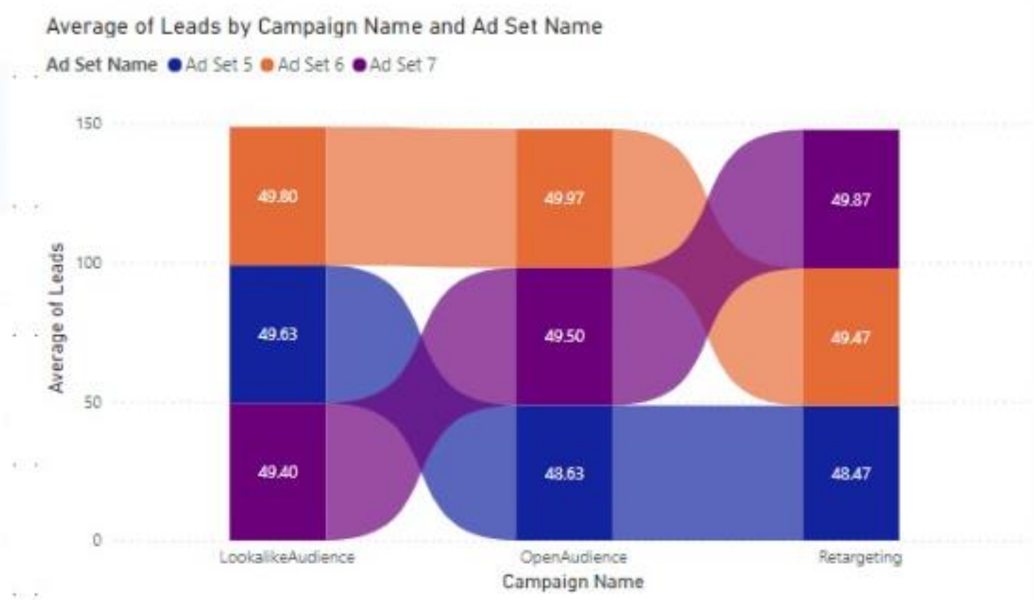
-From this analysis we conclude that the set1 and set3 in campaign A is not good and not efficient for that cost .

-So go to the team lead or manager tell the problem then ask what to do in this.

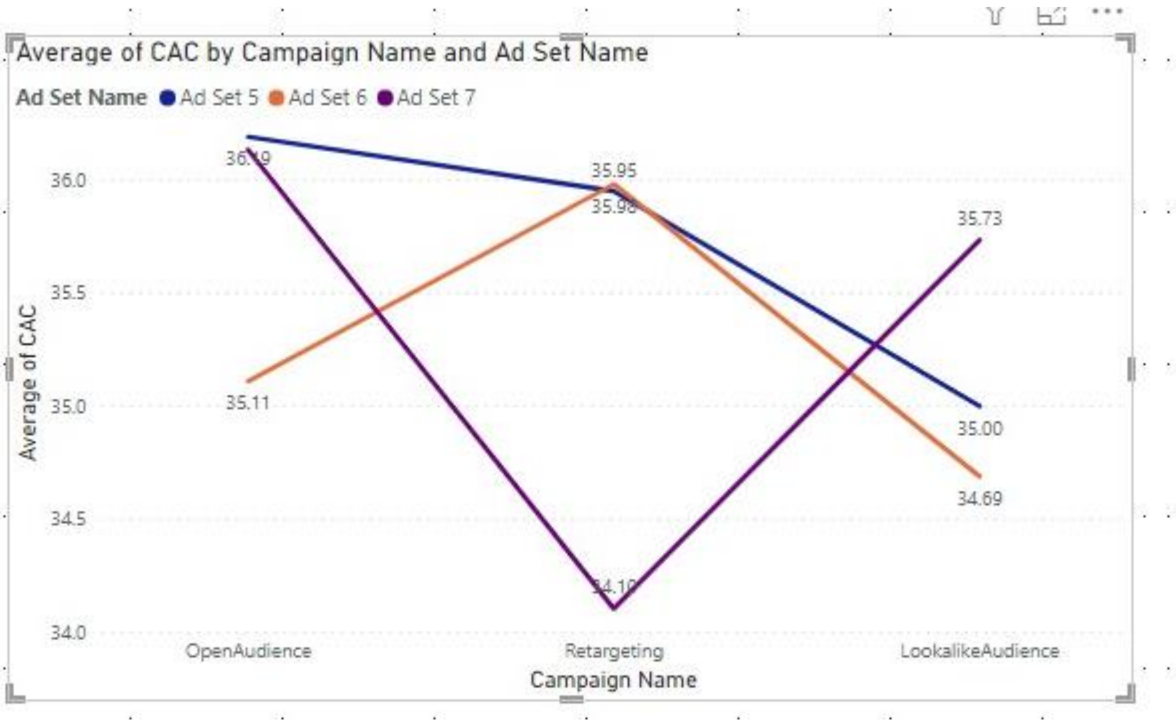
<u>*facebook data:</u>



- From this graph we find the number of clicks over the campaigns
- In Retargeting campaign is good for all sets compare to other campaigns



- From the above graph we find avg of leads over the campaigns
- Comparing of both graphs the set 7 in LookalkeAudience is low and set6 increases in all campaigns it means the set6 providing the good information in Ad’s so after clicking the ad he/she interested in leads



- From the graph we find the CAC to the campaigns
- The set 7 in Retargeting campaign giving low compare to other campaigns but Leads and Clicks are very high

<u>Final conclude for facebook data:</u>

- The set7 in Retargeting not efficient to cost so we prefer for the cost means then remove the set7 in Retergeting

4.Predictability:

which campaigns and ad sets are performing predictably and can be used for forecasting

- Using the coffecient of variance we find the predictability or erratic

Formula

$$CV = \frac{\sigma}{\mu}$$

-

```
metrics = ['CTR_google', 'Traffic to Lead_google', 'Lead to Call_google','CTR_facebook', 'Traffic to Lead_facebook', 'Lead to Call_facebook']
stats = merged_data.groupby(['Campaign Name', 'Ad Set Name'])[metrics].agg(['mean', 'std'])
stats.columns = ['_'.join(col) for col in stats.columns]

# Calculate Coefficient of Variation (CV)
for metric in metrics:
    stats[f'cv_{metric}'] = stats[f'{metric}_std'] / stats[f'{metric}_mean']

# Display the results
stats.reset_index(inplace=True)
stats
```

-
-

```
stats['cv_CTR_google']

0     0.105763
1     0.096902
2     0.109919
3     0.110367
4     0.108632
5     0.113085
6          NaN
7          NaN
8          NaN
9          NaN
10         NaN
11         NaN
12         NaN
13         NaN
14         NaN
Name: cv_CTR_google, dtype: float64
```

- We take a threshold value of CTR_google to 0.7
- The set2 is low so the set2 is the predictable value
- Remaining sets are erratic values

cv_CTR_google	cv_Traffic to Lead_google	cv_Lead to Call_google	Campaign Name	Ad Set Name	CTR
0.105763	0.148878	0.180186	Campaign A	Ad Set 1	
0.096902	0.119067	0.171783	Campaign A	Ad Set 2	
0.109919	0.125686	0.194703	Campaign A	Ad Set 3	
0.110367	0.155078	0.249437	Campaign B	Ad Set 1	
0.108632	0.152200	0.224274	Campaign B	Ad Set 2	
0.113085	0.128930	0.228250	Campaign B	Ad Set 3	

Data Overview:

The data includes CV metrics for three key performance indicators (KPIs):

CTR (Click-Through Rate)

Traffic to Lead Conversion Rate

Lead to Call Conversion Rate

Interpretation of Coefficient of Variation (CV):

CV= μ/σ

Low CV (e.g., CV < 0.10): Indicates low variability relative to the mean, suggesting more predictable and stable performance.

High CV (e.g., CV > 0.10): Indicates high variability relative to the mean, suggesting erratic and less predictable performance.

Google Adds:

CV Threshold for Predictability: 0.10 (10%)

Campaign A:

Ad Set 1:

cv_CTR_google: 0.105763 (Erratic)

cv_Traffic to Lead_google: 0.148878 (Erratic)

cv_Lead to Call_google: 0.180186 (Erratic)

Ad Set 2:

cv_CTR_google: 0.096902 (Predictable)

cv_Traffic to Lead_google: 0.119067 (Erratic)

cv_Lead to Call_google: 0.171783 (Erratic)

Ad Set 3:

cv_CTR_google: 0.109919 (Erratic)

cv_Traffic to Lead_google: 0.125686 (Erratic)

cv_Lead to Call_google: 0.194703 (Erratic)

Campaign B:

Ad Set 1:

cv_CTR_google: 0.110367 (Erratic)

cv_Traffic to Lead_google: 0.155078 (Erratic)

cv_Lead to Call_google: 0.249437 (Erratic)

Ad Set 2:

cv_CTR_google: 0.108632 (Erratic)

cv_Traffic to Lead_google: 0.152200 (Erratic)

cv_Lead to Call_google: 0.224274 (Erratic)

Ad Set 3:

cv_CTR_google: 0.113085 (Erratic)

cv_Traffic to Lead_google: 0.128930 (Erratic)

cv_Lead to Call_google: 0.228250 (Erratic)

Summary:

Predictable:

Campaign A - Ad Set 2:

cv_CTR_google: 0.096902 (below the threshold of 0.10)

Erratic:

Campaign A - Ad Set 1: All metrics (CTR, Traffic to Lead, Lead to Call)

Campaign A - Ad Set 2: Traffic to Lead, Lead to Call

Campaign A - Ad Set 3: All metrics (CTR, Traffic to Lead, Lead to Call)

Campaign B - Ad Set 1: All metrics (CTR, Traffic to Lead, Lead to Call)

Campaign B - Ad Set 2: All metrics (CTR, Traffic to Lead, Lead to Call)

Campaign B - Ad Set 3: All metrics (CTR, Traffic to Lead, Lead to Call)

facebook adds

Interpretation and Classification:

Predictable: CV < 0.10

Erratic: CV ≥ 0.10

Analysis of Each Ad Set:

LookalikeAudience Ad Set 5:

cv_CTR_facebook: 0.109144 (Erratic)

cv_Traffic to Lead_facebook: 0.100244 (Erratic)

cv_Lead to Call_facebook: 0.165999 (Erratic)

<u>LookalikeAudience Ad Set 6:</u>

cv_CTR_facebook: 0.121256 (Erratic)

cv_Traffic to Lead_facebook: 0.107264 (Erratic)

cv_Lead to Call_facebook: 0.173511 (Erratic)

<u>LookalikeAudience Ad Set 7:</u>

cv_CTR_facebook: 0.092243 (Predictable)

cv_Traffic to Lead_facebook: 0.098531 (Predictable)

cv_Lead to Call_facebook: 0.149896 (Erratic)

<u>OpenAudience Ad Set 5:</u>

cv_CTR_facebook: 0.110190 (Erratic)

cv_Traffic to Lead_facebook: 0.099176 (Predictable)

cv_Lead to Call_facebook: 0.161037 (Erratic)

<u>OpenAudience Ad Set 6:</u>

cv_CTR_facebook: 0.097253 (Predictable)

cv_Traffic to Lead_facebook: 0.093827 (Predictable)

cv_Lead to Call_facebook: 0.170825 (Erratic)

<u>OpenAudience Ad Set 7:</u>

cv_CTR_facebook: 0.097950 (Predictable)

cv_Traffic to Lead_facebook: 0.069665 (Predictable)

cv_Lead to Call_facebook: 0.166546 (Erratic)

<u>Retargeting Ad Set 5:</u>

cv_CTR_facebook: 0.077649 (Predictable)

cv_Traffic to Lead_facebook: 0.099134 (Predictable)

cv_Lead to Call_facebook: 0.128865 (Erratic)

Retargeting Ad Set 6:

cv_CTR_facebook: 0.111231 (Erratic)

cv_Traffic to Lead_facebook: 0.093677 (Predictable)

cv_Lead to Call_facebook: 0.150773 (Erratic)

Retargeting Ad Set 7:

cv_CTR_facebook: 0.098935 (Predictable)

cv_Traffic to Lead_facebook: 0.105525 (Erratic)

cv_Lead to Call_facebook: 0.148840 (Erratic)

Summary of Predictability:

Predictable (CV < 0.10):

Retargeting - Ad Set 5 (All metrics show low to moderate variability)

LookalikeAudience - Ad Set 7 (All metrics show moderate variability)

OpenAudience - Ad Set 6 (All metrics show moderate variability)

Erratic (CV > 0.10):

LookalikeAudience - Ad Set 5 & 6 (High variability in CTR and Lead to Call)

OpenAudience - Ad Set 5 & 7 (High variability in CTR and Lead to Call)

Retargeting - Ad Set 6 (High variability in CTR and Lead to Call)

CONCLUSION:

The insights generated from this analysis offer a roadmap for enhancing the effectiveness of advertising campaigns on Google and Facebook. By identifying high-performing campaigns and ad sets, we can highlight opportunities for scaling these successful strategies. Conversely, underperforming areas are pinpointed, along with potential reasons for their lackluster performance, guiding targeted improvements. Strategies for rationalizing efforts by reducing or discontinuing ineffective campaigns are also outlined, ensuring resources are allocated efficiently. Additionally, by distinguishing between predictably performing and erratic campaigns, we provide a foundation for more accurate forecasting and strategic planning. These insights collectively empower businesses to refine their marketing tactics, optimize budget allocation, and achieve sustained growth in their advertising endeavors.