Marketing Campaign Analysis

Executive summary

Dataset: ~200K marketing campaign records (2021–2022).

Objective: Evaluate ROI, cost efficiency, conversion behavior, channel performance, and identify underperforming high-spend campaigns.

High-level insight: Most campaigns land in Low ROI, but strong performance pockets exist across Website, Email, and Display; especially for Women 35–44 and Mandarin/Website combinations.

Data overview

Key columns: Campaign_ID, Company, Campaign_Type, Channel_Used, Duration (kept as string categories), Conversion_Rate, Acquisition_Cost (int), ROI, Clicks, Impressions, CTR, Cost_per_Conversion, Engagement_Score, ROI_Category.

Methods

- Loaded Unity Catalog table: workspace.default.marketing_campaign_dataset
- Cleaned acquisition cost; standardized categorical columns.
- Performed SQL analysis using Spark SQL magic.
- Created visual analyses: scatterplots, heatmaps, bar charts, and outlier detection.

3

```
#Data extraction
from pyspark.sql.functions import regexp_replace, regexp_extract, col, round, when, sum as spark_sum

df = spark.table("workspace.default.marketing_campaign_dataset")
display(df)
```

df: pyspark.sql.connect.dataframe.DataFrame = [Campaign_ID: long, Company: string ... 14 more fields]

Table

	Campaign_ID	Company	Campaign_Type	Target_Audience	Duration	Cha
1	1	Innovate Industries	Email	Men 18-24	30 days	Google
2	2	NexGen Systems	Email	Women 35-44	60 days	Google
3	3	Alpha Innovations	Influencer	Men 25-34	30 days	YouTube
4	4	DataTech Solutions	Display	All Ages	60 days	YouTube
5	5	NexGen Systems	Email	Men 25-34	15 days	YouTube
6	6	DataTech Solutions	Display	All Ages	15 days	Instagra
7	7	NexGen Systems	Email	Women 35-44	60 days	Website
8	8	DataTech Solutions	Search	Men 18-24	45 days	Google
9	9	Alpha Innovations	Social Media	Women 35-44	15 days	Faceboo
10	10	TechCorp	Email	Women 35-44	15 days	Instagra
11	11	NexGen Systems	Display	Men 25-34	45 days	Email
12	12	Innovate Industries	Influencer	Men 25-34	60 days	Google
13	13	TechCorp	Social Media	Men 25-34	60 days	Faceboo
14	14	TechCorp	Email	Men 25-34	45 days	Instagra
15	15	TechCorp	Display	All Ages	45 days	Website

10,000+ rows | Truncated data

4

```
# Check schema and row count
df.printSchema()
df.count()
```

```
root
 |-- Campaign_ID: long (nullable = true)
 |-- Company: string (nullable = true)
 |-- Campaign_Type: string (nullable = true)
 |-- Target_Audience: string (nullable = true)
 |-- Duration: string (nullable = true)
 |-- Channel_Used: string (nullable = true)
 |-- Conversion_Rate: double (nullable = true)
 |-- Acquisition_Cost: string (nullable = true)
 |-- ROI: double (nullable = true)
 |-- Location: string (nullable = true)
 |-- Language: string (nullable = true)
 |-- Clicks: long (nullable = true)
 |-- Impressions: long (nullable = true)
 |-- Engagement_Score: long (nullable = true)
 |-- Customer_Segment: string (nullable = true)
 |-- Date: date (nullable = true)

200000
```

Data Cleaning

- Removed symbols from Acquisition_Cost → cast to integer.
- Retained Duration as strings (“15 days”, “30 days”, “45 days”, “60 days”).
- Created KPIs (CTR, Cost_per_Conversion) and ROI Categories.
- Confirmed no remaining nulls.

6

```
# Acquisition_Cost: remove $ and , → cast to double → cast to int
df = df.withColumn(
    "Acquisition_Cost",
    regexp_replace(col("Acquisition_Cost"), "[$,]", "").cast("double").cast("int")
)

# Check schema
df.printSchema()

# Count nulls in each column
null_counts = df.select([spark_sum(col(c).isNull().cast("int")).alias(c) for c in df.columns])

display(null_counts)
```

```
df:  pyspark.sql.connect.dataframe.DataFrame = [Campaign_ID: long, Company: string ... 14 more fields]
null_counts:  pyspark.sql.connect.dataframe.DataFrame = [Campaign_ID: long, Company: long ... 14 more fields]
root
 |-- Campaign_ID: long (nullable = true)
 |-- Company: string (nullable = true)
 |-- Campaign_Type: string (nullable = true)
 |-- Target_Audience: string (nullable = true)
 |-- Duration: string (nullable = true)
 |-- Channel_Used: string (nullable = true)
 |-- Conversion_Rate: double (nullable = true)
 |-- Acquisition_Cost: integer (nullable = true)
 |-- ROI: double (nullable = true)
 |-- Location: string (nullable = true)
 |-- Language: string (nullable = true)
 |-- Clicks: long (nullable = true)
 |-- Impressions: long (nullable = true)
 |-- Engagement_Score: long (nullable = true)
 |-- Customer_Segment: string (nullable = true)
 |-- Date: date (nullable = true)
```

Table

	Campaign_ID	Company	Campaign_Type	Target_Audience	Duration	Chan
1	0	0	0	0	0	

1 row

7

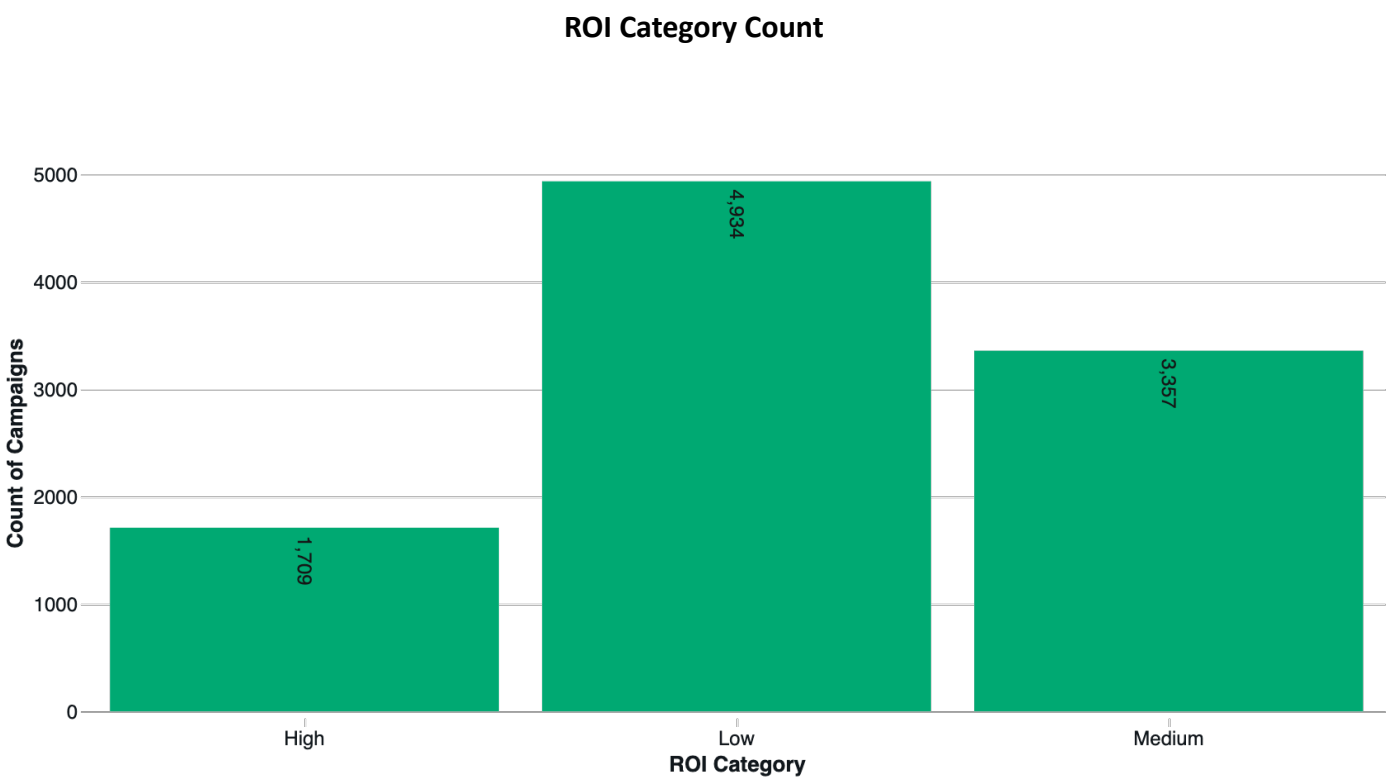
```python
# Cost per Conversion
df = df.withColumn("Cost_per_Conversion", round(col("Acquisition_Cost") / (col("Conversion_Rate") * col("Impressions")),2))

# CTR in percentage
df = df.withColumn("CTR", round((col("Clicks") / col("Impressions"))*100,2))

# ROI Category
df = df.withColumn(
    "ROI_Category",
    when(col("ROI") >= 7, "High")
    .when(col("ROI") >= 5, "Medium")
    .otherwise("Low")
)

# Verify new columns
display(df)
```

```
df:  pyspark.sql.connect.dataframe.DataFrame = [Campaign_ID: long, Company: string ... 17 more fields]
```

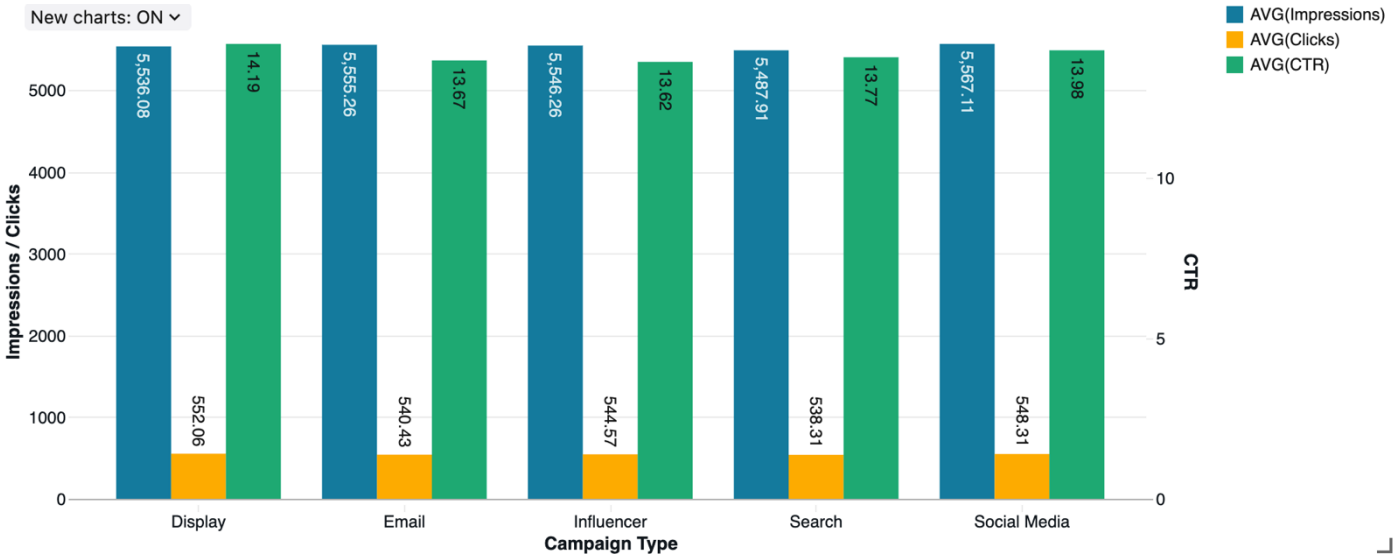
Table | ROI Category Count | Impressions, Clicks and CTR by Campaign Type | Acquisition C

	Campaign_ID	Company	Campaign_Type	Target_Audience	Duration	Char
1	1	Innovate Industries	Email	Men 18-24	30 days	Google
2	2	NexGen Systems	Email	Women 35-44	60 days	Google
3	3	Alpha Innovations	Influencer	Men 25-34	30 days	YouTube
4	4	DataTech Solutions	Display	All Ages	60 days	YouTube
5	5	NexGen Systems	Email	Men 25-34	15 days	YouTube
6	6	DataTech Solutions	Display	All Ages	15 days	Instagra
7	7	NexGen Systems	Email	Women 35-44	60 days	Website
8	8	DataTech Solutions	Search	Men 18-24	45 days	Google
9	9	Alpha Innovations	Social Media	Women 35-44	15 days	Faceboo
10	10	TechCorp	Email	Women 35-44	15 days	Instagra
11	11	NexGen Systems	Display	Men 25-34	45 days	Email
12	12	Innovate Industries	Influencer	Men 25-34	60 days	Google
13	13	TechCorp	Social Media	Men 25-34	60 days	Faceboo
14	14	TechCorp	Email	Men 25-34	45 days	Instagra
15	15	TechCorp	Display	All Ages	45 days	Website

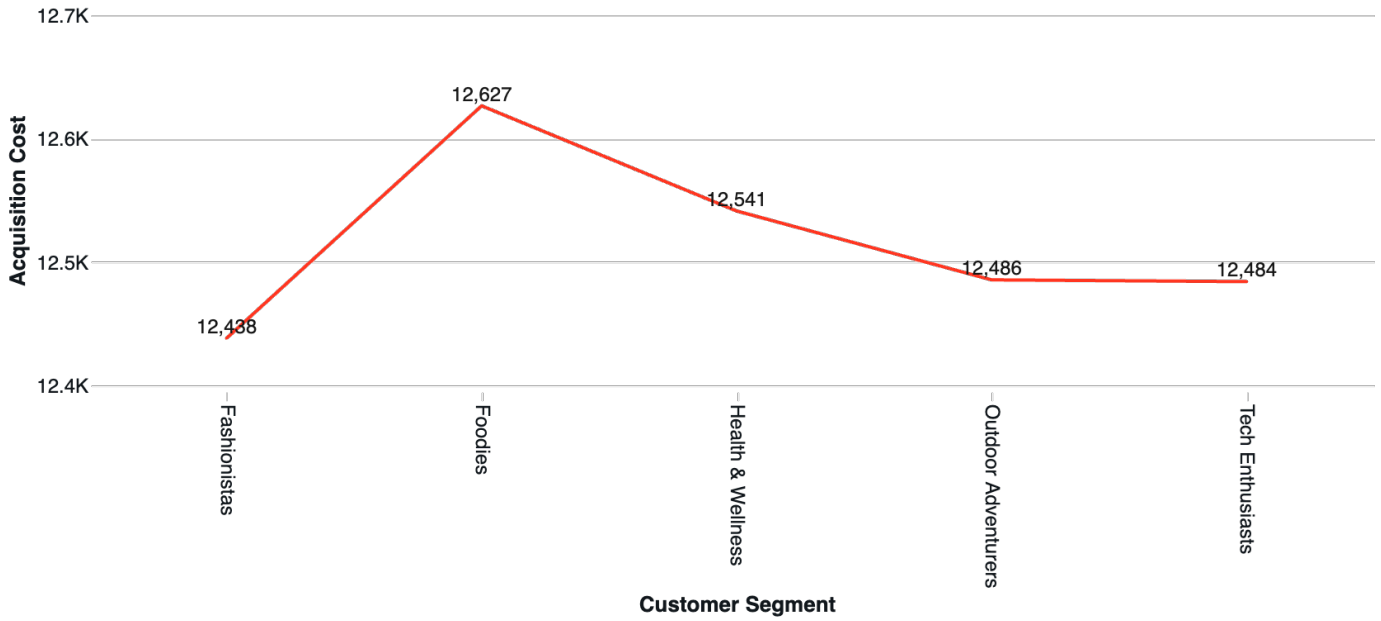
10,000+ rows | Truncated data



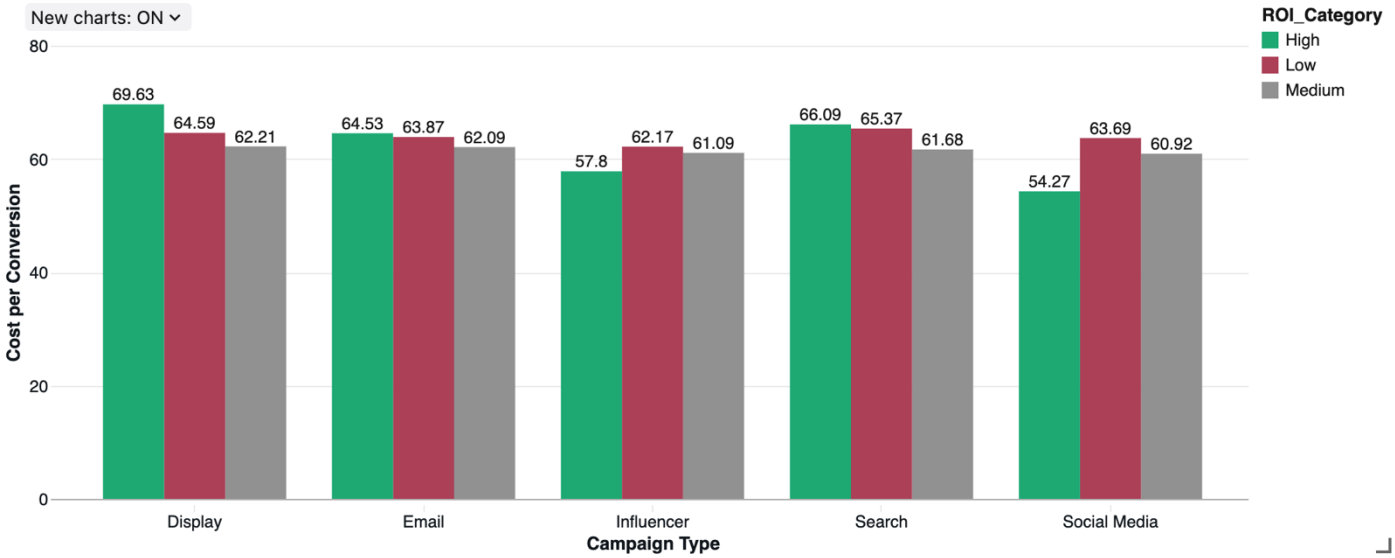
Impressions, Clicks and CTR by Campaign Type



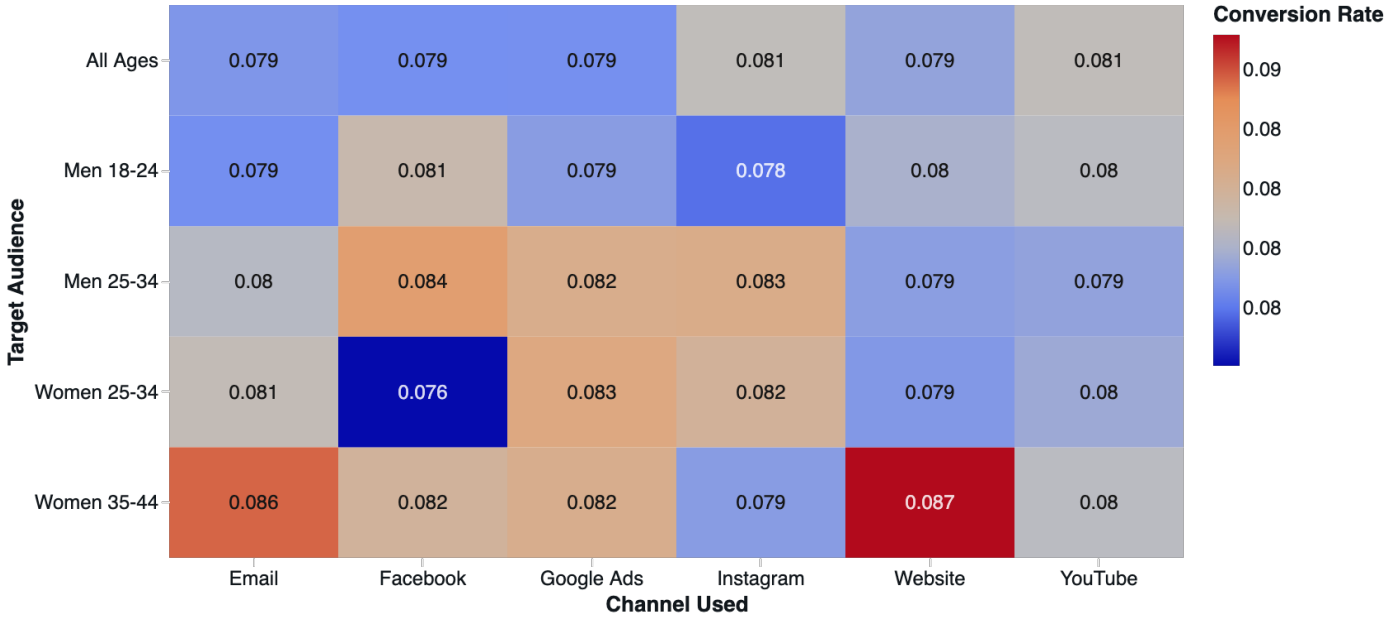
Acquisition Cost by Customer Segment



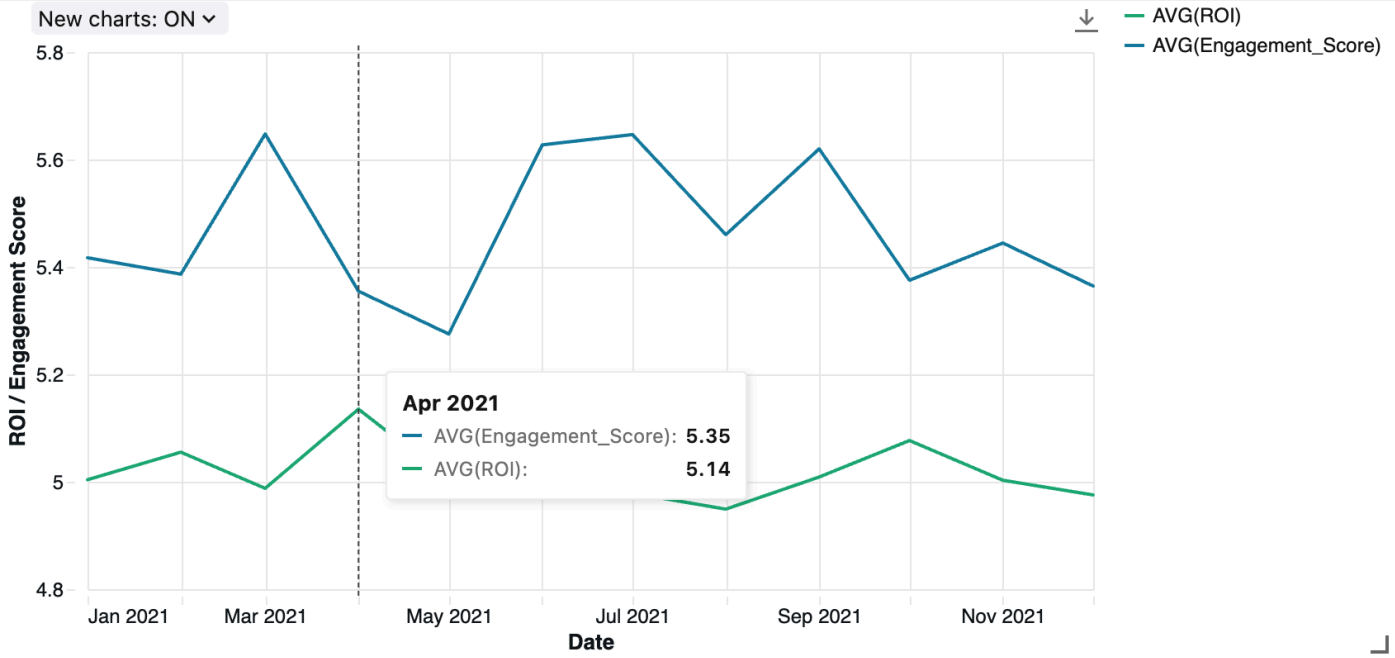
Cost per Conversion for Different ROI Category by Campaign Type



Conversion Rate of Different Target Audience across Different Channels



Monthly ROI and Engagement Score Trends



Key findings

ROI & Categories

- **ROI category counts:** High = **1,709**, Medium = **3,357**, Low = **4,934**.
- **Observation:** Although Low ROI is the majority, the High ROI group is large enough to justify focused scaling experiments on those campaigns.

Reach & Engagement

- **Top impressions by channel:** Social Media (avg ~5,567) > Display > Email > Search (avg ~5,488).
- **Top clicks:** Display campaigns generated the highest average clicks.
- **CTR:** Display and Website-driven campaigns show higher CTR (Display: high CTR; Website for specific audiences is strong).

Conversion & Audience

- **Best converting audience:** Women 35–44 on Website and Email (Conversion Rate ~0.087 and ~0.086 respectively).
- **Lowest observed:** Women 25–34 on Facebook (~0.076).
- **Insight:** Prioritize creative & personalization for Women 35–44; revisit creative/offer for Women 25–34 on Facebook.

Cost Efficiency

- **Acquisition cost by Customer Segment:** Foodies segment shows the highest acquisition cost (12,525); Tech enthusiasts lowest (~12,480).

- **Cost per Conversion by ROI & Campaign Type:**
  - **High ROI:** Display has highest cost per conversion (~69.63).
  - **Medium ROI:** Display again highest (~62.21).
  - **Low ROI:** Search shows highest cost per conversion (~65.37).
- **Interpretation:** Display often drives scale and ROI but at higher acquisition cost; some Display campaigns are high-value but expensive, test optimizing creative or targeting to lower cost-per-conversion.

Time trends

- **Monthly ROI:** Peak in April 2021 (5.14), trough in December (~4.98).
- **Engagement Score:** Peak March 2021 (5.65), lowest in May 2021 (~5.27).
- **Action:** Align major spend increases to months with historical performance lift and run A/B tests in lower-performing months.

Recommendations

1. **Scale winners, test to reduce cost:** For channels/campaigns with High ROI but high Cost per Conversion (e.g., some Display campaigns), run creative and targeting A/B tests focused on lowering acquisition cost while preserving conversion quality.
2. **Prioritize audience: Women 35–44** on Website and Email — increase targeted spend, personalized creatives, and dedicated landing pages.
3. **Optimize Search** for low-ROI / high cost-per-conversion campaigns — refine keywords, landing pages, and bidding strategies.
4. **Seasonality plan:** Increase tests and budget shifts into April-based windows where ROI historically peaks; prepare mitigation campaigns for low months (Dec/May).
5. **Segment-level experiments:** For the Foodies segment (highest acquisition cost), run a micro-campaign testing offer variations and measure ROI lift before full reallocation.

Advanced SQL Analysis & Visual Insights

This section summarizes four advanced SQL analyses conducted on the `marketing_campaigns` dataset to identify outlier behavior, channel–language strengths, segment-level ROI efficiency, and underperforming high-reach campaigns. Each analysis is paired with recommended visualizations and key insights generated from the results.

10

```
# Create a temporary SQL view for SQL queries
df.createOrReplaceTempView("marketing_campaigns")
```

High-Cost, Low-ROI Campaigns (Top 50 Outliers)

12

```
%sql
SELECT
  Campaign_ID, Company, Campaign_Type, Duration, Cost_per_Conversion, ROI, CTR
FROM marketing_campaigns
WHERE Cost_per_Conversion > (SELECT percentile_approx(Cost_per_Conversion, 0.75) FROM
marketing_campaigns)
  AND ROI < (SELECT percentile_approx(ROI, 0.25) FROM marketing_campaigns)
ORDER BY Cost_per_Conversion DESC
LIMIT 50;
```

_sqldf: pyspark.sql.connect.dataframe.DataFrame = [Campaign_ID: long, Company: string ... 5 more fields]

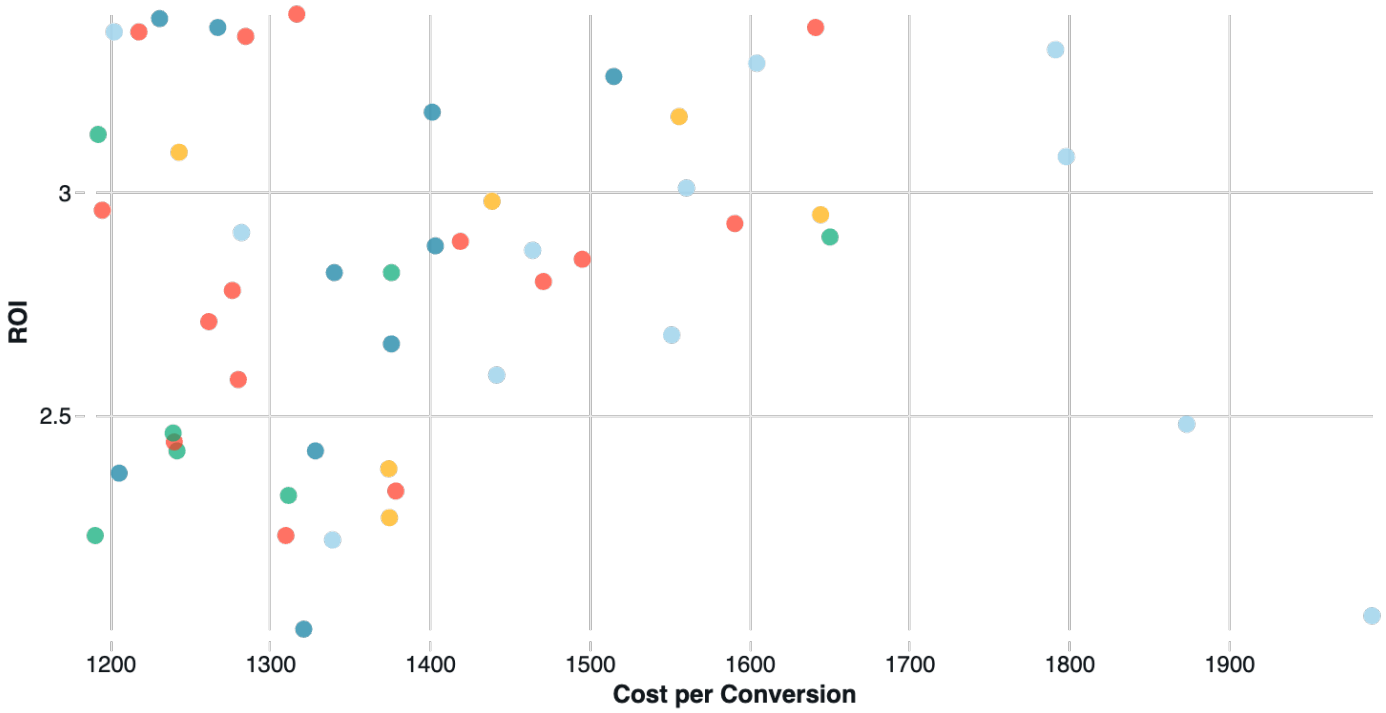
Table | Top 50 High-Cost, Low-ROI Campaigns

	Campaign_ID	Company	Campaign_Type	Duration	Cost_per_Conversion	1.2
1	121860	Alpha Innovations	Social Media	30 days	1989.61	
2	53764	DataTech Solutions	Social Media	45 days	1873.64	
3	196832	Innovate Industries	Social Media	15 days	1798.29	
4	130167	Alpha Innovations	Social Media	45 days	1791.67	
5	8188	Alpha Innovations	Influencer	15 days	1650.5	
6	71642	TechCorp	Email	45 days	1644.66	
7	170826	Innovate Industries	Search	45 days	1641.51	
8	19262	DataTech Solutions	Social Media	45 days	1604.68	
9	128310	Innovate Industries	Search	60 days	1590.97	
10	7200	DataTech Solutions	Social Media	60 days	1560.68	
11	58559	TechCorp	Email	15 days	1556.15	
12	155000	NexGen Systems	Social Media	15 days	1551.42	
13	70792	Innovate Industries	Display	45 days	1515.26	
14	31569	Alpha Innovations	Search	30 days	1495.54	
15	16468	Innovate Industries	Search	15 days	1471.21	

50 rows

This result is stored as `_sqldf` and can be used in other Python and SQL cells.

Top 50 High-Cost, Low-ROI Campaigns



Key Insights

- Social Media campaigns showed **the lowest ROI (~2.05)** while simultaneously having **the highest Cost per Conversion (~1990)** among the outliers.
- Other low-ROI, high-cost patterns were seen across multiple companies, suggesting cross-channel inefficiencies.
- These campaigns represent priority targets for **budget cuts or creative/targeting optimization**.

Languages × Channel performance (ROI & Conversion)

15

```
%sql
SELECT
  Language, Channel_Used,
  ROUND(AVG(ROI),2) AS avg_ROI,
  ROUND(AVG(Conversion_Rate),4) AS avg_conversion,
  COUNT(*) AS campaigns
FROM marketing_campaigns
GROUP BY Language, Channel_Used
HAVING COUNT(*) >= 50
ORDER BY avg_ROI DESC;
```

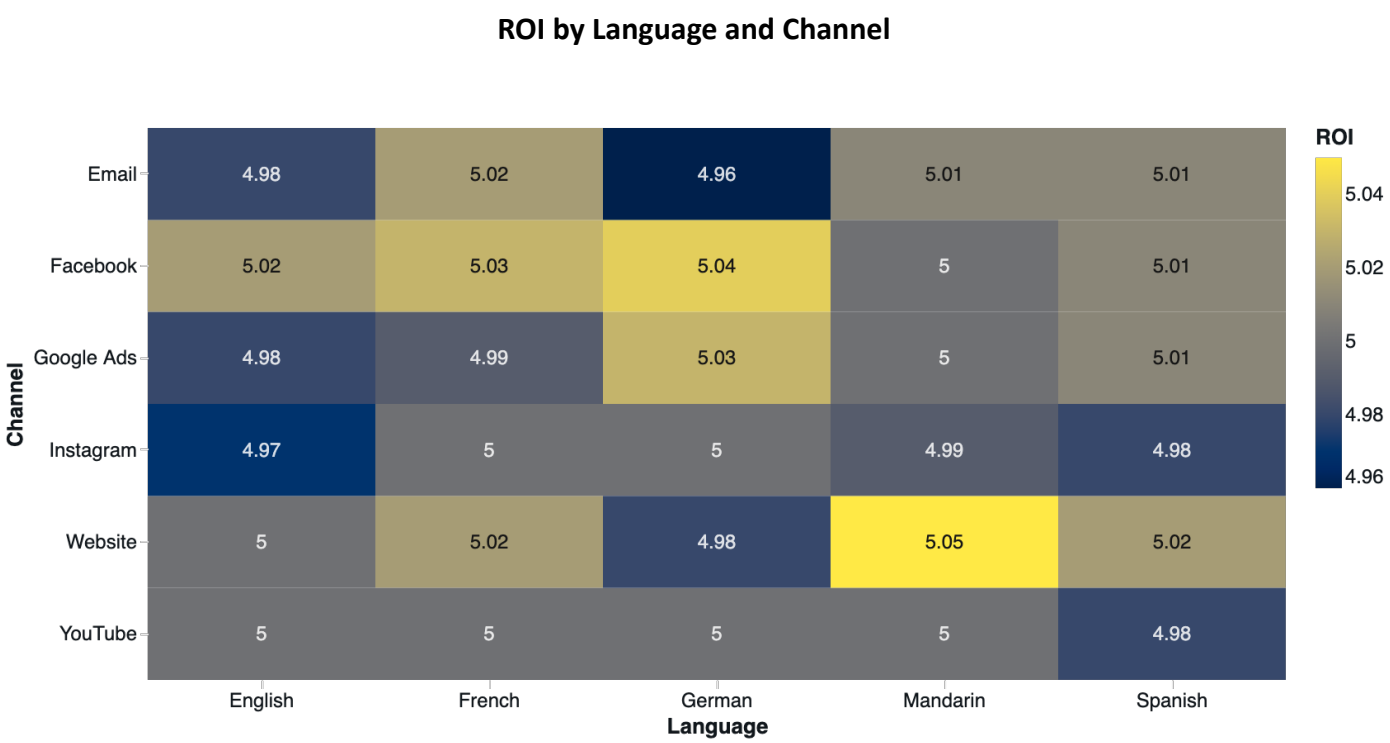
_sqldf: pyspark.sql.connect.dataframe.DataFrame = [Language: string, Channel_Used: string ... 3 more fields]

Table | ROI by Language and Channel

	Language	Channel_Used	avg_ROI	avg_conversion	campaigns
1	Mandarin	Website	5.05	0.0794	6791
2	German	Facebook	5.04	0.0788	6617
3	French	Facebook	5.03	0.0802	6582
4	German	Google Ads	5.03	0.0805	6632
5	French	Website	5.02	0.0806	6601
6	French	Email	5.02	0.0802	6694
7	English	Facebook	5.02	0.0798	6577
8	Spanish	Website	5.02	0.0804	6734
9	Spanish	Email	5.01	0.0798	6558
10	Spanish	Google Ads	5.01	0.0802	6861
11	Spanish	Facebook	5.01	0.0807	6470
12	Mandarin	Email	5.01	0.0805	6827
13	English	Website	5	0.0801	6569
14	French	Instagram	5	0.0802	6619
15	German	Instagram	5	0.0803	6627

30 rows

This result is stored as `_sqldf` and can be used in other Python and SQL cells.



Key Insights

- **Website + Mandarin** delivered the **highest average ROI (5.05)** across all combinations.
- **Facebook + German** also performed strongly with competitive ROI values.
- **Email + German** showed the **lowest ROI (~4.96)**, indicating potential message–audience mismatch.
- This highlights the importance of **language-tailored creative** and **channel–audience alignment**.

ROI Efficiency by Customer Segment (ROI per Dollar Spent)

18

```
%sql
SELECT
  Customer_Segment,
  ROUND(AVG(ROI / NULLIF(Acquisition_Cost,0)), 6) AS roi_per_dollar,
  COUNT(*) AS campaigns
FROM marketing_campaigns
GROUP BY Customer_Segment
ORDER BY roi_per_dollar DESC;
```

_sqldf: pyspark.sql.connect.dataframe.DataFrame = [Customer_Segment: string, roi_per_dollar: double ... 1 more field]

Table

	Customer_Segment	roi_per_dollar	campaigns
1	Tech Enthusiasts	0.000463	40151
2	Fashionistas	0.000462	39742
3	Health & Wellness	0.000462	39888
4	Outdoor Adventurers	0.000462	40011
5	Foodies	0.000461	40208

5 rows

This result is stored as `_sqldf` and can be used in other Python and SQL cells.

Key Insights

- All segments demonstrated **very close ROI-per-dollar efficiency**, with minimal difference between them.
- **Tech Enthusiasts** led marginally as the most cost-efficient audience.
- **Foodies** showed the lowest ROI-per-dollar, indicating potential for offer redesign or targeting refinement.

High-Impression Campaigns Underperforming CTR (Below Channel Median)

21

```
%sql
WITH channel_medians AS (
  SELECT Channel_Used, percentile_approx(CTR, 0.5) AS median_ctr
  FROM marketing_campaigns
  GROUP BY Channel_Used
)
SELECT m.Campaign_ID, m.Company, m.Channel_Used, m.Impressions, m.Clicks, m.CTR, m.Duration
FROM marketing_campaigns m
JOIN channel_medians cm ON m.Channel_Used = cm.Channel_Used
WHERE m.Impressions >= 10000 AND m.CTR < cm.median_ctr
ORDER BY m.Impressions DESC
LIMIT 50;
```

_sqldf: pyspark.sql.connect.dataframe.DataFrame = [Campaign_ID: long, Company: string ... 5 more fields]

Table

	Campaign_ID	Company	Channel_Used	Impressions	Clicks	CTR	
1	26806	DataTech Solutions	Google Ads	10000	958	9.58	30
2	89918	TechCorp	Google Ads	10000	233	2.33	60
3	93720	Innovate Industries	YouTube	10000	988	9.88	30
4	29381	Innovate Industries	Google Ads	10000	586	5.86	60
5	164936	NexGen Systems	Facebook	10000	338	3.38	15
6	26042	DataTech Solutions	Website	10000	414	4.14	60
7	106474	TechCorp	Instagram	10000	792	7.92	45
8	71869	Alpha Innovations	Instagram	10000	871	8.71	30
9	8905	TechCorp	Website	10000	822	8.22	15
10	60573	TechCorp	Instagram	10000	498	4.98	45
11	173029	Alpha Innovations	Google Ads	10000	638	6.38	45
12	159436	TechCorp	YouTube	10000	208	2.08	15
13	183416	NexGen Systems	Facebook	10000	705	7.05	30
14	133042	Innovate Industries	YouTube	10000	801	8.01	15
15	192775	DataTech Solutions	Website	10000	531	5.31	30

23 rows

This result is stored as `_sqldf` and can be used in other Python and SQL cells.

Key Insights

- Large numbers of Google Ads and YouTube campaigns fell below their channel median CTR despite reaching **10,000 impressions**.
- Some campaigns had very low CTR values (e.g., **2.08–2.91** for Google Ads campaigns).
- Duration varied across 15, 30, 45, and 60 days, suggesting that **duration alone is not the cause**.
- These campaigns are ideal for **creative refresh, improved targeting, or landing page optimization**.

Overall Insights from SQL Analysis

- Outlier campaigns (top 50 by cost per conversion) indicate **major budget inefficiencies**, especially on Social Media.
- Website + Mandarin and Facebook + German emerged as **high-performing language–channel pairs**, while Email + German underperformed.
- Customer segments showed **small but meaningful differences** in ROI-per-dollar, with Tech Enthusiasts offering the best value.
- A significant subset of campaigns (especially Google Ads & YouTube) achieved high reach but low CTR — prime candidates for optimization.

Recommendations

These findings support the broader strategy of:

- Reallocating budget away from high-cost/low-ROI outliers
- Expanding investment in high-performing language–channel combinations
- Running targeted creative tests on underperforming high-impression campaigns
- Refining segment-specific messaging where ROI efficiency dips

---