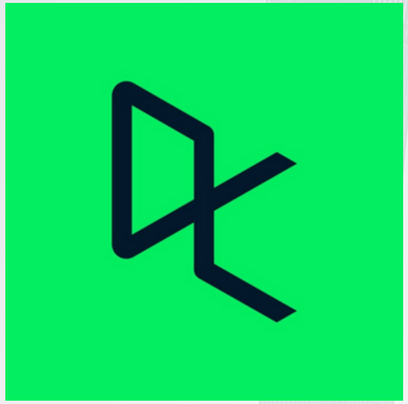
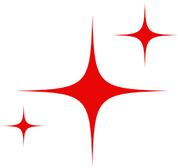
DATA CAMP



PROFESSIONAL DATA ANALYSIS CERTIFICATE PRESENTATION



Presented By: LE TUAN ANH

PRODUCT_SALES

CONTENT

PART 1: DATA VALIDATION AND CLEANING

01 Check data validation

```
# Checking for missing values and finding each column data type
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15000 entries, 0 to 14999
Data columns (total 8 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   week               15000 non-null  int64
 1   sales_method       15000 non-null  object
 2   customer_id        15000 non-null  object
 3   nb_sold            15000 non-null  int64
 4   revenue            13926 non-null  float64
 5   years_as_customer  15000 non-null  int64
 6   nb_site_visits     15000 non-null  int64
 7   state              15000 non-null  object
dtypes: float64(1), int64(4), object(3)
memory usage: 937.6+ KB
```

```
# 1. Check missing value
df.isna().sum()
```
[205]

	0
week	0
sales_method	0
customer_id	0
nb_sold	0
revenue	1074
years_as_customer	0
nb_site_visits	0
state	0

Table Chart 8 rows

```
# 2. Data validation: Check the unique values to identify any unexpected or erroneous data.
df.nunique()
```
[206]

	0
week	6
sales_method	5
customer_id	15000
nb_sold	10
revenue	6743
years_as_customer	42
nb_site_visits	27
state	50

Table Chart 8 rows

```
# 3. Check outlier: Using summary statistics such as mean, median,standard deviation, and quartiles, we can identify outliers
numeric columns.
df.describe()
```
[207]

	week	nb_sold	revenue	years_as_customer	nb_site_visits
count	15000	15000	13926	15000	15000
mean	3.0982666667	10.0846666667	93.9349425535	4.9659333333	24.9908666667
std	1.6564198071	1.8122133327	47.4353122457	5.0449515589	3.5009142152
min	1	7	32.54	0	12
25%	2	9	52.47	1	23
50%	3	10	89.5	3	25
75%	5	11	107.3275	7	27
max	6	16	238.32	63	41

Table Chart 8 rows

Checking for data validation shows there are **1074 missing values in the revenue** column. In the **sales_method** column we see **5 different methods which is wrong** (we should have 3 type of methods), and **there are outliers in years_as_customer**, because The company was founded in 1984 and is currently operating in 2024 and the maximum number of years customers have purchased from us is 40 years. So, we can't have a customers since 63 years ago!.

PART 1: DATA VALIDATION AND CLEANING

01 Handle unique and outlier

```
# We should have 3 unique values in sales_method column.
df['sales_method'].unique()
```

[208]

```
array(['Email', 'Email + Call', 'Call', 'em + call', 'email'],
      dtype=object)
```

```
# Standardize the values in the 'sales_method' column by replacing 'em + call' with 'Email + Call' and 'email' with 'Email'
df.sales_method = df.sales_method.replace({'em + call':'Email + Call','email':'Email'})
print(f'Unique values in the sales_method column after replacement:{df.sales_method.unique()}')
```

[209]

```
Unique values in the sales_method column after replacement:['Email' 'Email + Call' 'Call']
```

Instead of three unique sales method we have five as follow: **['Email' 'Email + Call' 'Call' 'em + call' 'email']**.
After correcting typos (replacing with correct ones) in the sales_method column, unique values are as follows: **['Email' 'Email + Call' 'Call']**.

b. Handle outlier: year_as_customer

```
# Values greater than 40 in years_as_customer are wrong and we should consider them as outliers
df[df['years_as_customer']>40]
```

[210]

	week	sales_method	customer_id	nb_sold	revenue	years_as_customer	nb_site_visits	state
13741	2	Email	18919515-a618-430c-9a05-2c7d8fe...	10	97.22	63	24	California
13800	4	Call	2ea97d34-571d-4e1b-95be-fea1c4...	10	50.47	47	27	California

Table Chart 2 rows

=> There are only two records as years_as_customer outliers, so we can easily drop them.

```
df = df[df.years_as_customer <= 40]
```

[211]

```
# Check the number of records after filtering
print("Number of records after filtering:", len(df))
```

[212]

```
Number of records after filtering: 14998
```

The business started 40 years ago; therefore, if the values in years_as_customer show numbers greater than 40, they are incorrect.

Fortunately, there are only two incorrect values, and because there are 15,000 records, dropping them doesn't make our analysis biased.

PART 1: DATA VALIDATION AND CLEANING

01 Handle missing value

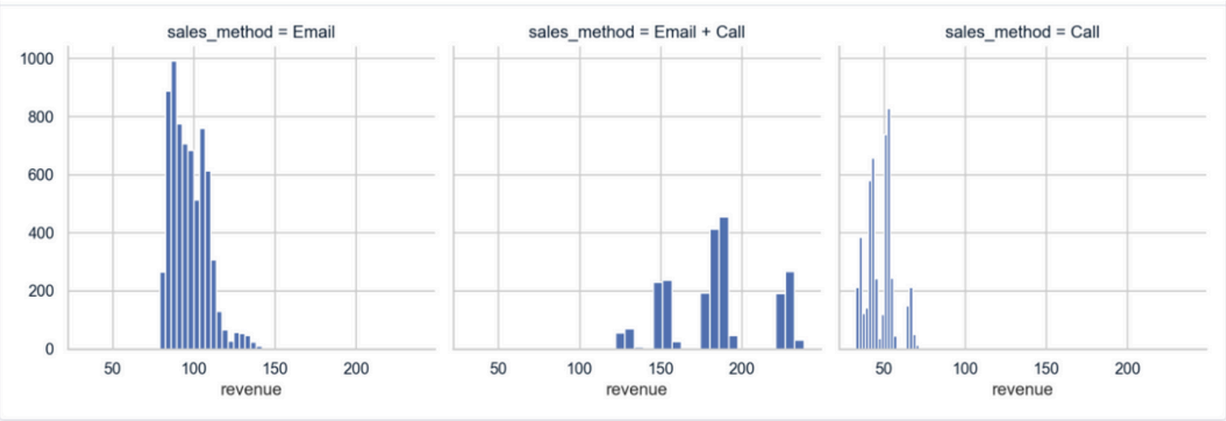
c. Handle missing value: revenue column

```
# Proportion of missing values relative to the total entries in the revenue column
(df.revenue.isna().sum()/df.revenue.count())*100
```

```
7.713300775639184
```

sales_method	Not_Null	Null
Call	4780	181
Email	6921	544
Email + Call	2223	349

```
Chi-square statistic: 251.11783987671868
p-value: 2.954295391738752e-55
```



There are **more than 5 percent missing values in revenue column (7.71 %)**, so we can not easily drop them unless we examine randomness of missing values. If they are missing completely at random (MCAR) or missing at random (MAR), dropping them may be less problematic. However, if they are missing not at random (MNAR), dropping them could introduce bias.

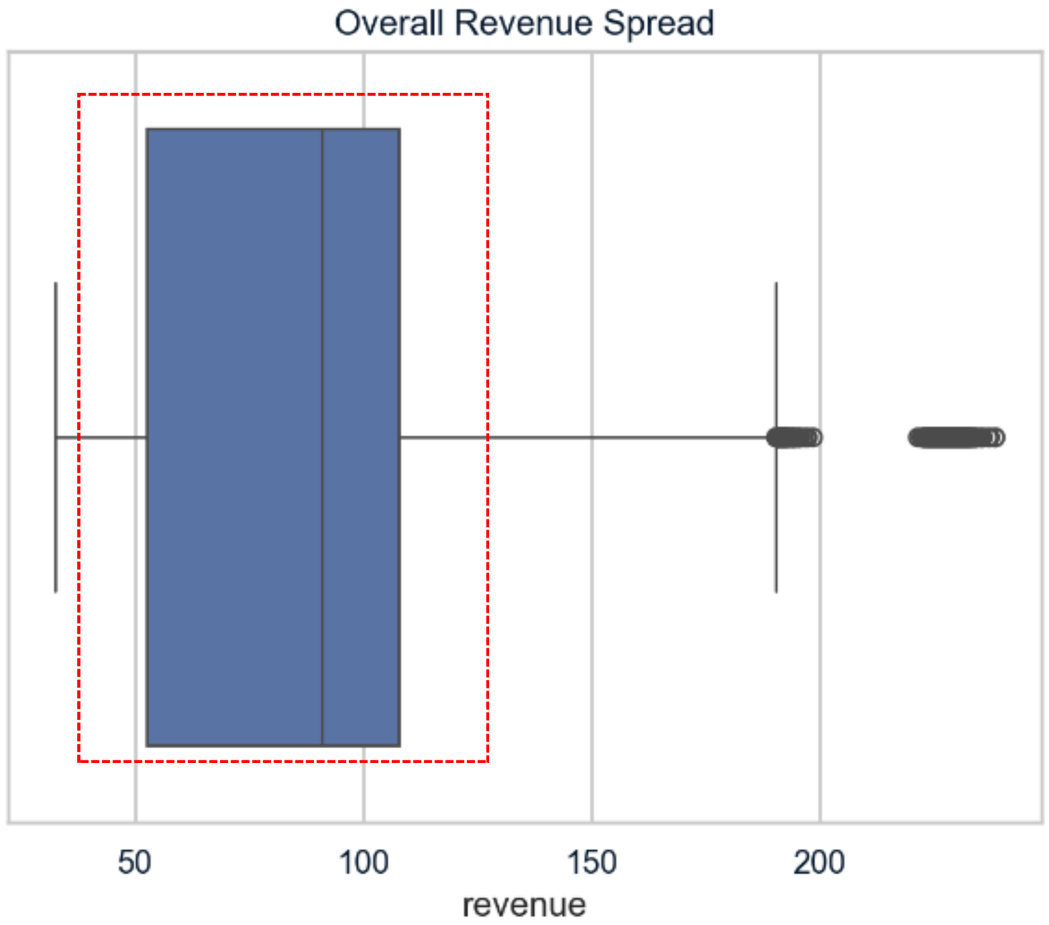
To conduct a statistical test to determine if there is a relationship between missing values in the 'revenue' column and the 'sales_method' column, we can use a chi-square test of independence. This test is appropriate when both variables are categorical, so we first **create a categorical revenue column**, then we **create a contingency table** representing the frequencies of each combination of categories. At the end we can use the **chi2_contingency function** from the **scipy.stats module** to perform the test.

AS you can see the p_value is **less than 5 percents**, **So there is a significant relationship between 'sales_method' and 'revenue_category',** therefore we can't easily drop missing values. To avoid bias, let's look at the distribution of revenue for each sales method.

When the distribution of revenue for each sales method is not normal, it is generally advisable to **use the median** for imputation rather than the mean.

PART 2: EXPLORE ANALYSIS

Now, our data is ready for exploratory data analysis. Let's see the overall revenue spread and the revenue spread for each sales method.

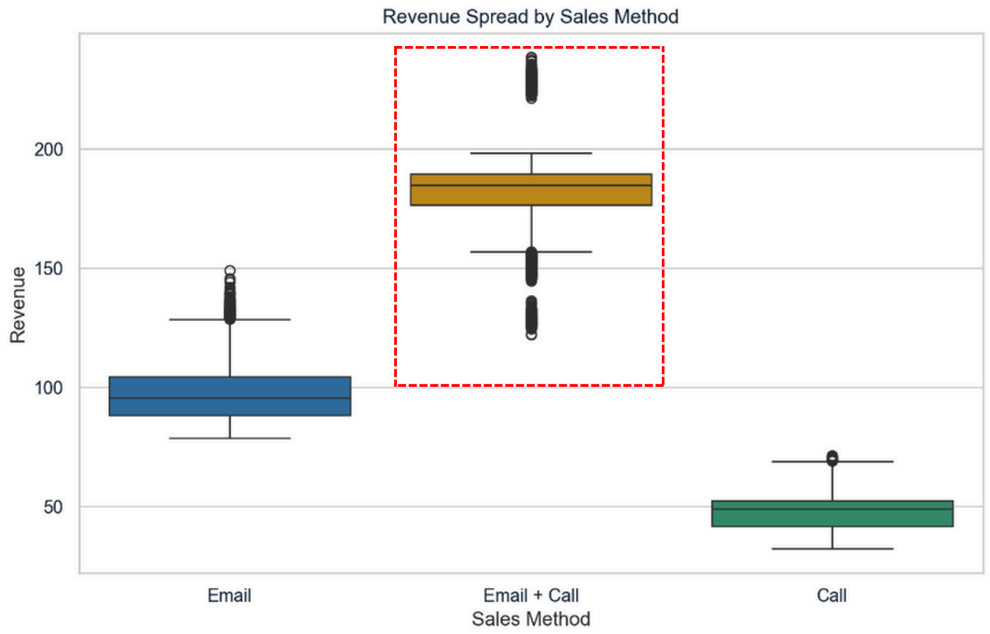


Sales revenue is often concentrated in the range of 50 to 100 units of currency, while higher revenues occur less frequently.

PART 2: EXPLORE ANALYSIS

Median revenue for each sales method is as follows:

sales_method	Median Value
Call	49.05
Email	95.58
Email + Call	184.74



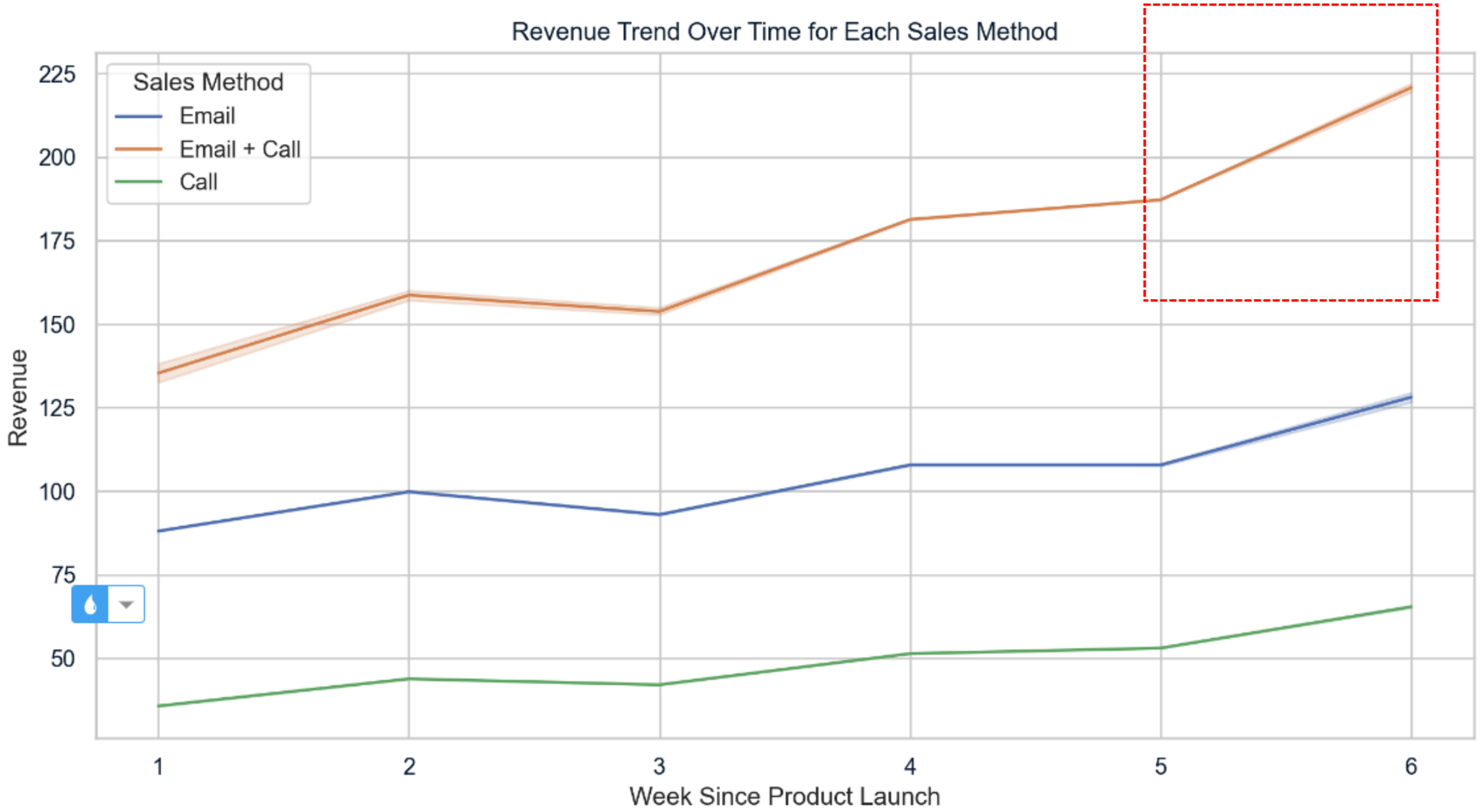
Observations:
Using Email to reach customers appears to be more effective than only using Calls.
The combination of Email and Call yields the highest effectiveness, indicating that combining multiple methods to approach customers can significantly increase revenue.

This could be due to various reasons, such as:
Email provides an open communication channel, offers more detailed information, and can be referred to later.
Calls provide direct and more personal interaction.
The combination of both methods leverages the strengths of each, creating a more comprehensive approach to customer engagement.

DATA CAMP

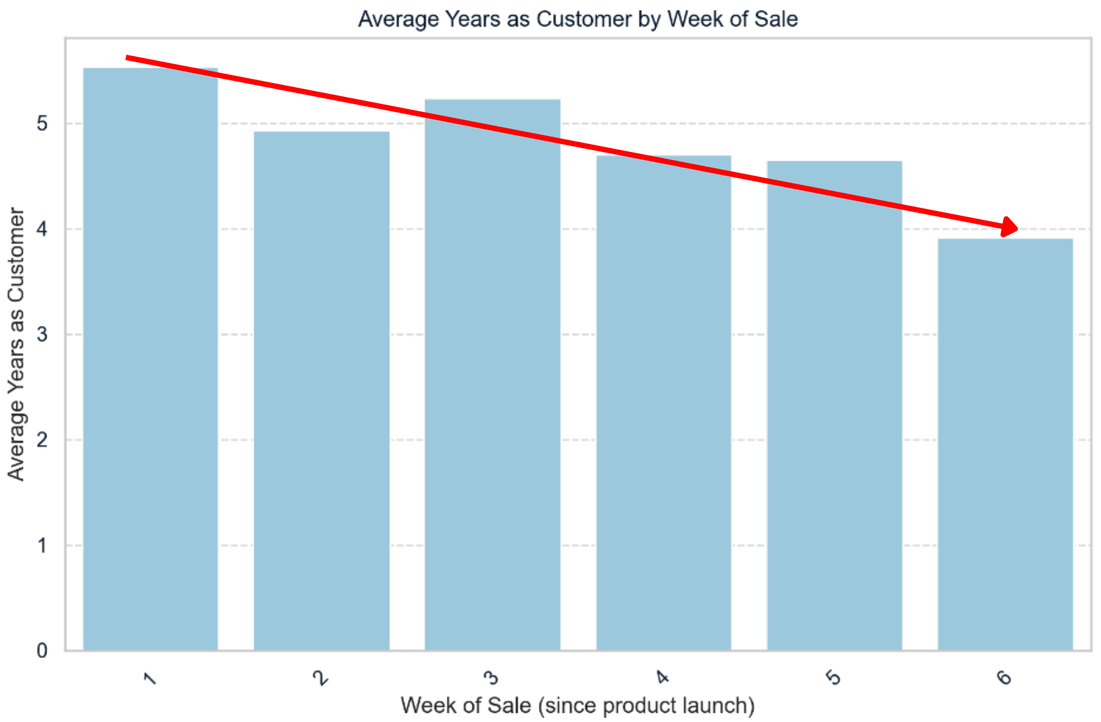
PART 2: EXPLORE ANALYSIS

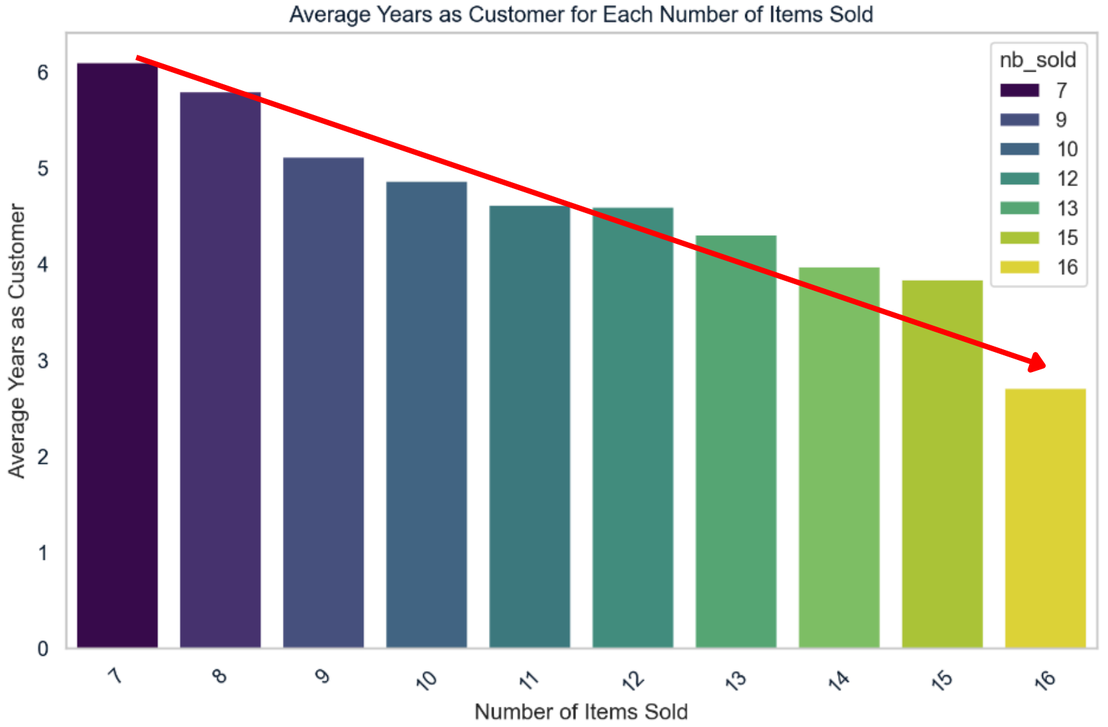
To analyze the difference in median revenue over time for each sales method, we can create line plots showing the trend of revenue over the weeks since the product launch.



Email + Call method has garnered more customer attention and higher spending, resulting in significant revenue. And it has the trend to increase growth more than other methods.

PART 2: EXPLORE ANALYSIS

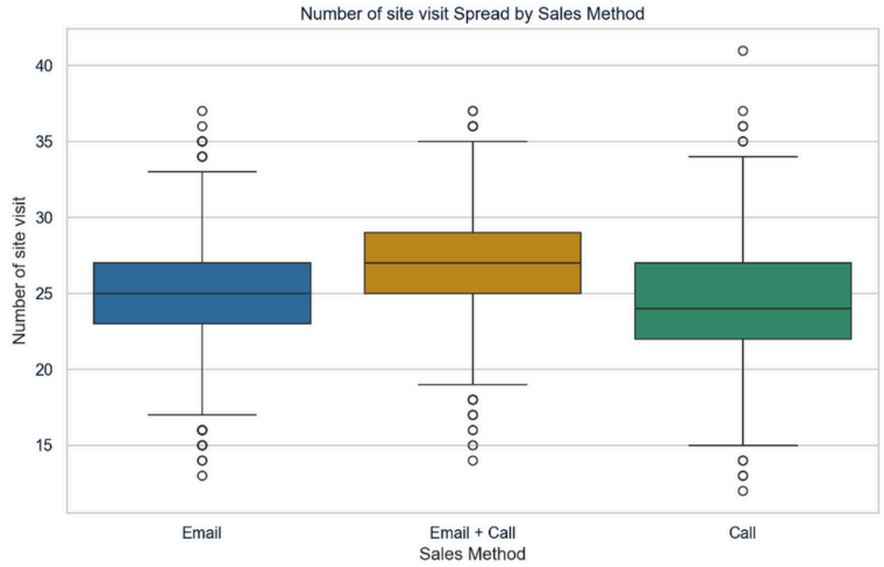




Analysis of the number of items sold and who bought earlier shows that older customers tend to buy earlier and more than new customers, which makes sense and shows that our long-term relationships with customers are healthy and that they trust us.

PART 3: METRIC

Average Revenue per Customer Visit is a metric that measures the average amount of revenue generated by each visit to the company's website. It helps in understanding the effectiveness of the company's online sales strategy and the value derived from each interaction with customers on the website.



Sales Method	Initial Average Revenue per Visit
Call	1.951001
Email	3.919088
Email + Call	6.865501

Average Revenue per Customer Visit is a metric that measures the average amount of revenue generated by each visit to the company's website. It helps in understanding the effectiveness of the company's online sales strategy and the value derived from each interaction with customers on the website.

How to Use the Metric:

Monitor Trends: Track the average revenue per customer visit over time to identify trends and patterns. Increasing trends may indicate improvements in online sales strategies or customer engagement, while decreasing trends may signal areas for improvement.

Benchmarking: Compare the average revenue per visit against industry benchmarks or competitors' performance to assess the company's competitiveness and identify areas for improvement.

Performance Evaluation: Use the metric to evaluate the effectiveness of marketing campaigns, website optimizations, or changes in sales strategies. Identify high-performing periods and factors contributing to success.

It provides insights into the effectiveness of each sales method in converting website visits into revenue and helps in identifying which sales methods are most successful in driving revenue per visit.

Conclusion

Based on these findings, it appears that the **"Email + Call" method** is the most effective in generating revenue per visit. However, we need to monitor metrics for all sales methods for a longer period to make informed decisions and optimize sales strategies.

There are other metrics that can be used like Conversion Rate, Customer Acquisition Cost (CAC), Customer Lifetime Value (CLV), Churn Rate, Return on Investment (ROI) and Average Order Value (AOV). In future analyses, using more detailed data, we can provide more specific metrics, which are crucial for decision-making.

THANK YOU

FOR YOUR LISTENING