CSCE - 5320

Project Increment - 1

Project Title:

Visualising the Social Media Sales Prediction & Ad-Campaign Analysis

Team Members:

Name	Email_Id	Student_Id
Nagasai Gummadi	nagasaigummadi@my.unt.edu	11540905
Divya Sri Vakkala	divyasrivakkala@my.unt.edu	11590101
Harsha Buddana	harshavardhanabuddana@my.unt.edu	11521994
Eswara Reddy Thimmapuram	eswarareddythimmapuram@my.unt.edu	11506566

• Domain

Our Project comes under the domain of **Digital Marketing.** It is a broad domain and deals with use of digital channels such as Search engines, Social Media, Email marketing and more. This domain majorly relies on a 3-step process. The steps are

I. **Targeting** : Where we first look who is the customer base we want to target for the product. This could be done by a variety of factors such as age, interest, demographics, gender and more.

II. **Reaching Out** : Process of reaching the Target audience and the tools required or means of communication we use to reach the defined Target audience

III. **Conversion Optimization** : Once we are done with reaching out, Now we need to optimise the user experience of the user in such a way that the ad we sent should convert into a Sale.

By following the above defined process, Digital marketers can turn ads into sales and drive real valuable results to their business.

• Data Abstraction

Dataset (Type and Attributes):

For this project, we have downloaded the dataset from Kaggle. This dataset is taken from Facebook's social media ad campaign.

This dataset is a CSV file. A table format with 11 columns and 1143 data points for each attribute.

The dataset consists of the following columns,

1.) ad_id: an unique ID for each ad.

2.) xyzcampaignid: an ID associated with each ad campaign of XYZ company.

3.) fbcampaignid: an ID associated with how Facebook tracks each campaign.

4.) age: age of the person to whom the ad is shown.

5.) gender: gender of the person to whom the ad is shown

6.) interest: a code specifying the category to which the person's interest belongs (interests are as mentioned in the person's Facebook public profile).

7.) Impressions: the number of times the ad was shown.

8.) Clicks: number of clicks on for that ad.

9.) Spent: Amount paid by company xyz to Facebook, to show that ad.

10.) Total conversion: Total number of people who enquired about the product after seeing the ad.

11.) Approved conversion: Total number of people who bought the product after seeing the ad.

We can also look at a part of the dataset below.

Loading Data

```
[ ] df=pd.read_csv("/content/KAG_conversion_data.csv")
```

```
[ ] df.head()
```

	ad_id	xyz_campaign_id	fb_campaign_id	age	gender	interest	Impressions	Clicks	Spent	Total_Conversion	Approved_Conversion
0	708746	916	103916	30-34	M	15	7350	1	1.43	2	1
1	708749	916	103917	30-34	M	16	17861	2	1.82	2	0
2	708771	916	103920	30-34	M	20	693	0	0.00	1	0
3	708815	916	103928	30-34	M	28	4259	1	1.25	1	0
4	708818	916	103928	30-34	M	28	4133	1	1.29	1	1

We have also made sure there are no null values in the dataset.

Checking for null values

```
[ ] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1143 entries, 0 to 1142
Data columns (total 11 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   ad_id                1143 non-null   int64
 1   xyz_campaign_id      1143 non-null   int64
 2   fb_campaign_id       1143 non-null   int64
 3   age                  1143 non-null   object
 4   gender               1143 non-null   object
 5   interest             1143 non-null   int64
 6   Impressions          1143 non-null   int64
 7   Clicks               1143 non-null   int64
 8   Spent                1143 non-null   float64
 9   Total_Conversion     1143 non-null   int64
 10  Approved_Conversion  1143 non-null   int64
dtypes: float64(1), int64(8), object(2)
memory usage: 98.4+ KB
```

We can have a look at the shape and description of the dataset.

Exploratory Data Analysis

```
[ ] df.shape
```

```
(1143, 11)
```

```
[ ] df.describe()
```

	ad_id	xyz_campaign_id	fb_campaign_id	interest	Impressions	Clicks	Spent	Total_Conversion	Approved_Conversion
count	1.143000e+03	1143.000000	1143.000000	1143.000000	1.143000e+03	1143.000000	1143.000000	1143.000000	1143.000000
mean	9.872611e+05	1067.382327	133783.989501	32.766404	1.867321e+05	33.390201	51.360656	2.855643	0.944007
std	1.939928e+05	121.629393	20500.308622	26.952131	3.127622e+05	56.892438	86.908418	4.483593	1.737708
min	7.087460e+05	916.000000	103916.000000	2.000000	8.700000e+01	0.000000	0.000000	0.000000	0.000000
25%	7.776325e+05	936.000000	115716.000000	16.000000	6.503500e+03	1.000000	1.480000	1.000000	0.000000
50%	1.121185e+06	1178.000000	144549.000000	25.000000	5.150900e+04	8.000000	12.370000	1.000000	1.000000
75%	1.121804e+06	1178.000000	144657.500000	31.000000	2.217690e+05	37.500000	60.025000	3.000000	1.000000
max	1.314415e+06	1178.000000	179982.000000	114.000000	3.052003e+06	421.000000	639.949998	60.000000	21.000000

• Task Abstraction:

The main task of this project is to perform analysis of Facebook ad campaign data and optimise sales conversion. We then use these campaign results to predict future sales. The objective is to analyse the factors that influence sales such as ad impressions, number of clicks and cost for each click etc and create a predictive model that can be applied to optimise future ad campaigns. The project involves tasks like loading the dataset, performing preprocessing, exploratory data analysis and developing machine learning models for sales prediction.

Task (Target and Actions):

The target of this project is to perform sales prediction based on the results of analysis of Facebook ad campaign dataset.

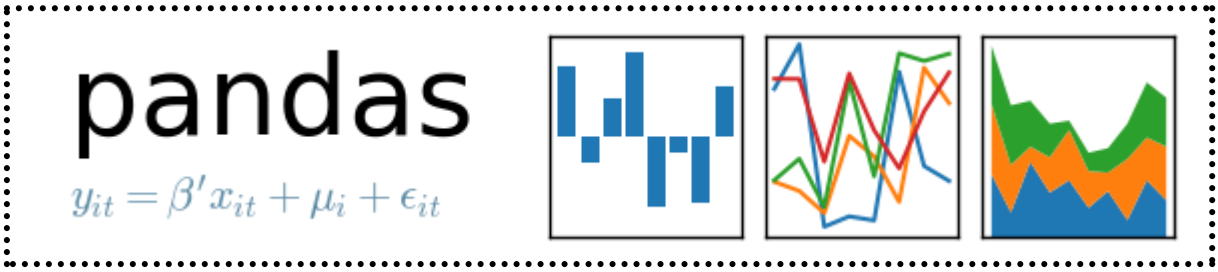
This task involves the following actions:

- Loading and Preprocessing the dataset
- Performing exploratory data analysis
- Training machine learning models for predicting sales
- Performance evaluation of models using different metrics
- Enhance the model based on the obtained results and gain insights that lead to sales prediction

• Implementation using tools

Loading dataset and Exploratory data analysis - Pandas library

- We have used the pandas library for loading the dataset and performing exploratory data analysis. Pandas is a popular python library which is commonly used for data analysis and manipulation. It is also used for tasks like data cleaning and data preprocessing. There are many tools and functions offered by pandas for data manipulation such as dataframes and series. It is used by many users due to its high performance and productivity.



Data visualisation - Matplotlib and Seaborn

- We have used Matplotlib and Seaborn libraries for the purpose of data visualisation. Matplotlib is a popular python library used for creating different kinds of charts, plots and graphs. It offers various visualisation tools making it easier to present the data to the users.

- Seaborn is also a popular data visualisation library built on top of matplotlib. It offers a high level interface for producing attractive and advanced visualisations. It can also be utilised for data exploration. Seaborn offers various functions for visualising different types of data like distribution plots, matrix plots and regression plots.

Machine learning modelling - Scikit-learn

- We have used the scikit-learn library for training the machine learning models to predict sales. Scikit-learn is a machine learning library which offers various tools for machine learning tasks such as clustering, classification and regression. It is built on top of Scipy, Numpy and Matplotlib. It also provides tools for data preprocessing, model selection and feature selection.

• Preliminary Results for Analysis

In this part of the project we majorly focus on performing Exploratory Data Analysis on the taken dataset. We have used the above mentioned tools to get the insights from the data. We plan to convert this project into a Web application using Streamlit.

Using the Pandas library and Matplotlib we have generated the following maps. Firstly we analysed the no. of campaigns.

Campaigns

```
[ ]  df["xyz_campaign_id"].unique()

     array([ 916,  936, 1178])
```

We can observe that the xyz corporation has three separate advertising campaigns here. For better visualisation, we'll now change their names to campaign a, campaign b, and campaign c.

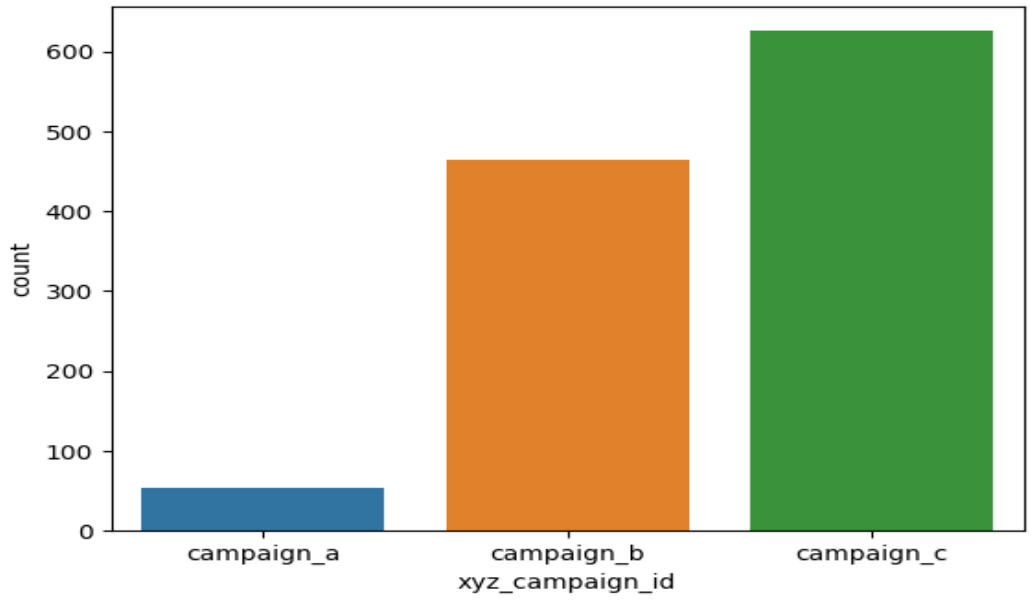
```
[ ]  df["xyz_campaign_id"].replace({916:"campaign_a",936:"campaign_b",1178:"campaign_c"}, inplace=True)
```

```
[ ]  df.head()
```

	ad_id	xyz_campaign_id	fb_campaign_id	age	gender	interest	Impressions	Clicks	Spent	Total_Conversion	Approved_Conversion
0	708746	campaign_a	103916	30-34	M	15	7350	1	1.43	2	1
1	708749	campaign_a	103917	30-34	M	16	17861	2	1.82	2	0
2	708771	campaign_a	103920	30-34	M	20	693	0	0.00	1	0
3	708815	campaign_a	103928	30-34	M	28	4259	1	1.25	1	0
4	708818	campaign_a	103928	30-34	M	28	4133	1	1.29	1	1

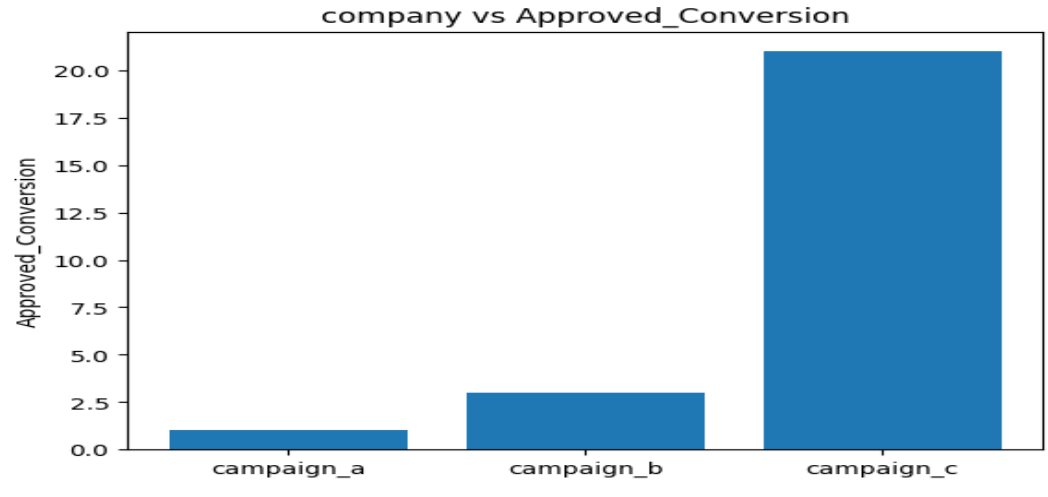
This is how the rendered dataframe looks after we replace the campaign names.

```
[ ]  # count plot on single categorical variable
     sns.countplot(x ='xyz_campaign_id', data = df)
     # Show the plot
     plt.show()
```



A simple count plot to plot the no of ad_campaings per each campaign. From the above plot we can infer that Campaign_c has most no_of ads.

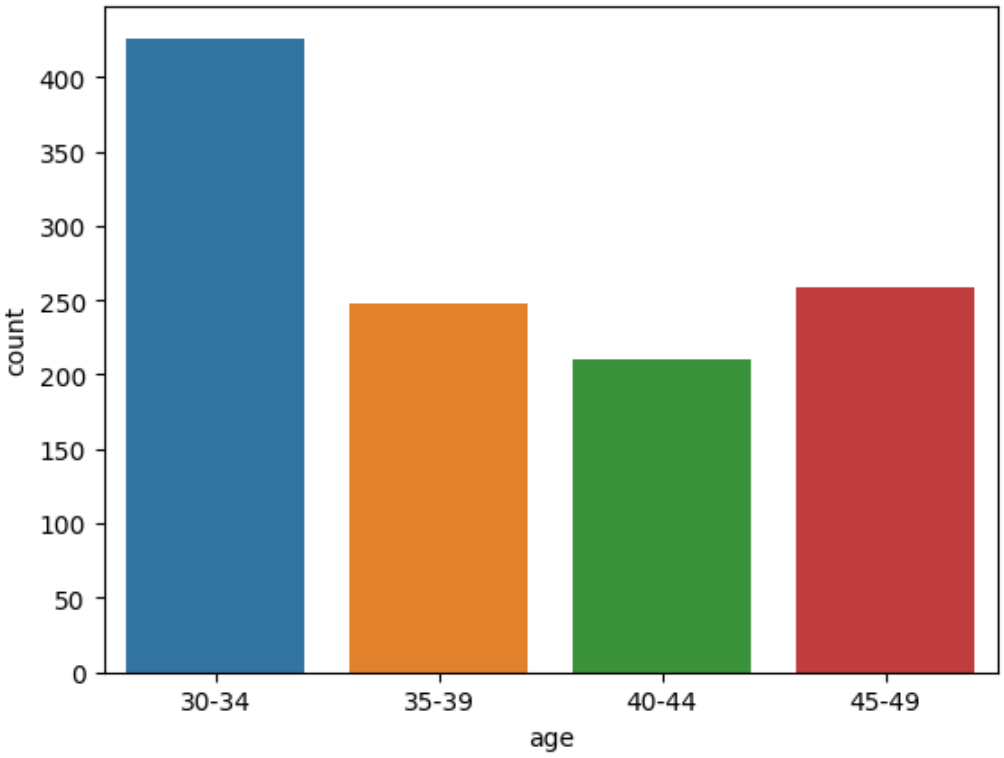
```
[ ]  #Approved_Conversion
     # Creating our bar plot
     plt.bar(df["xyz_campaign_id"], df["Approved_Conversion"])
     plt.ylabel("Approved_Conversion")
     plt.title("company vs Approved_Conversion")
     plt.show()
```



We tried to plot a countplot for no. Approved_conversion in each campaign. From the above plot we can deduce that Campaign_chas best conversion rate i.e, most people bought products in campaign_c.

Age

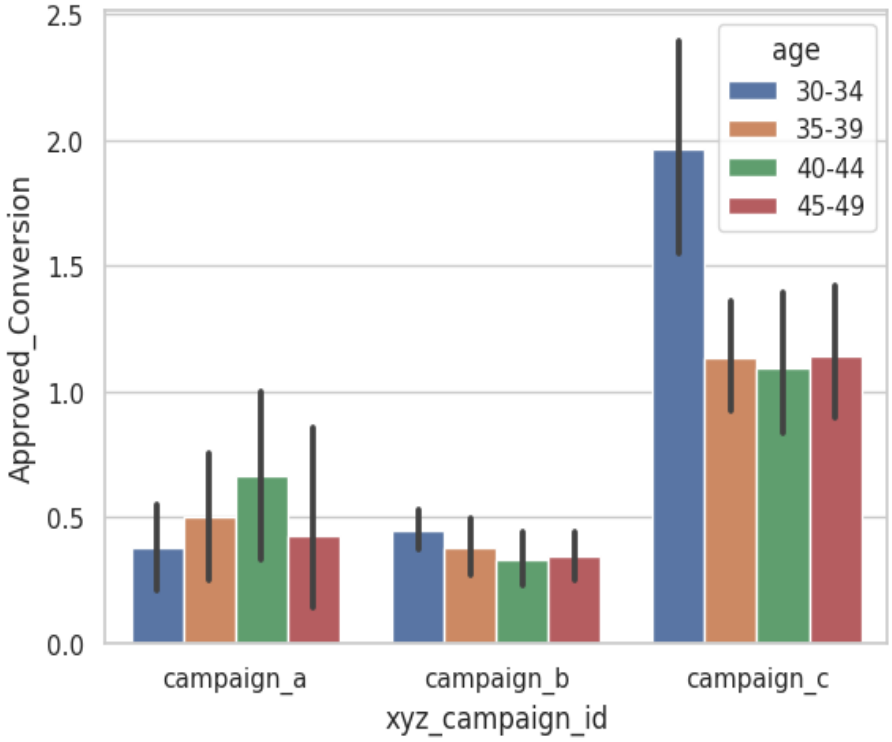
```
[ ]  # count plot on single categorical variable
     sns.countplot(x ='age', data = df)
     # Show the plot
     plt.show()
```



We tried to plot the most targeted age-group from all the campaigns. We can conclude from the above countplot that the age group 30-34 is the most target age-group.

```
import seaborn as sns
sns.set(style="whitegrid")
tips = sns.load_dataset("tips")
sns.barplot(x=df["xyz_campaign_id"], y=df["Approved_Conversion"], hue=df["age"], data=tips)
```

```
<Axes: xlabel='xyz_campaign_id', ylabel='Approved_Conversion'>
```

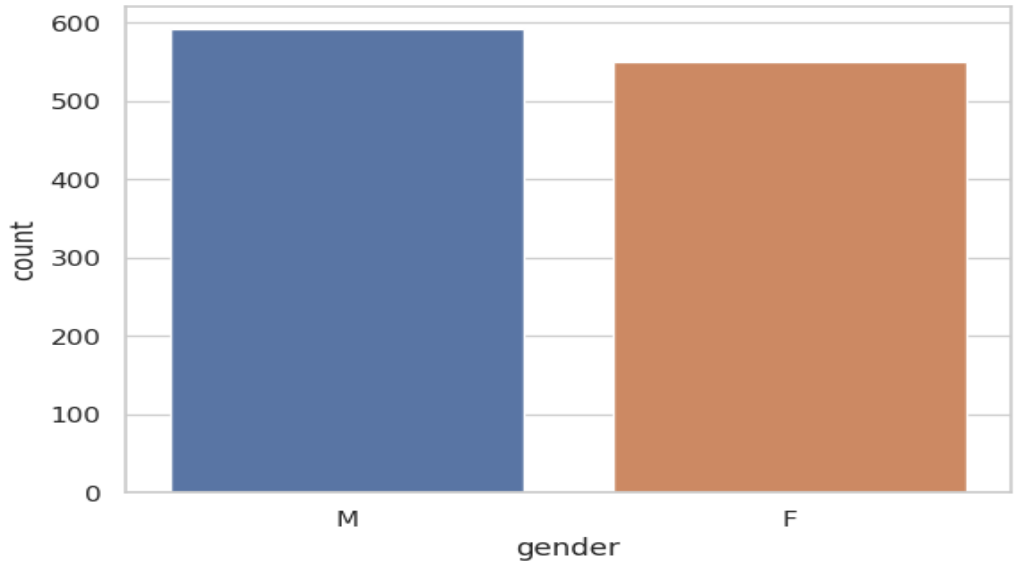


We used seaborn's barplot function to create a grouped bar plot that shows the relationship between campaign ID, approved conversions, and age.

It's interesting to note that in campaign_c and campaign_b, the age group of 30-34 shows more interest, whereas in campaign_a the age group of 40-44 shows more interest.

Gender

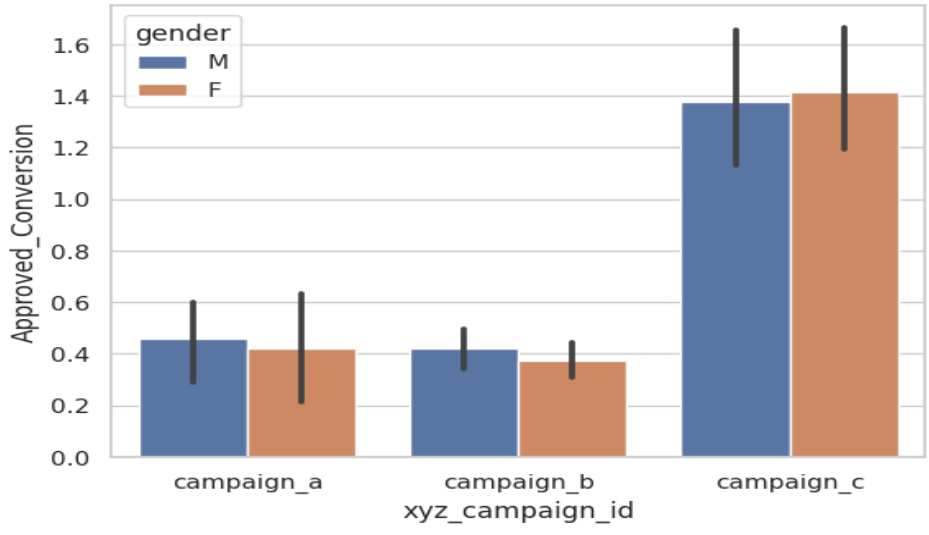
```
[ ]  # count plot on single categorical variable
     sns.countplot(x ='gender', data = df)
     # Show the plot
     plt.show()
```



We tried to create a countplot between the gender and we can infer from the graph that there are more male audiences than the female audience.

```
[ ]  import seaborn as sns
     sns.set(style="whitegrid")
     tips = sns.load_dataset("tips")
     sns.barplot(x=df["xyz_campaign_id"], y=df["Approved_Conversion"], hue=df["gender"], data=tips)
```

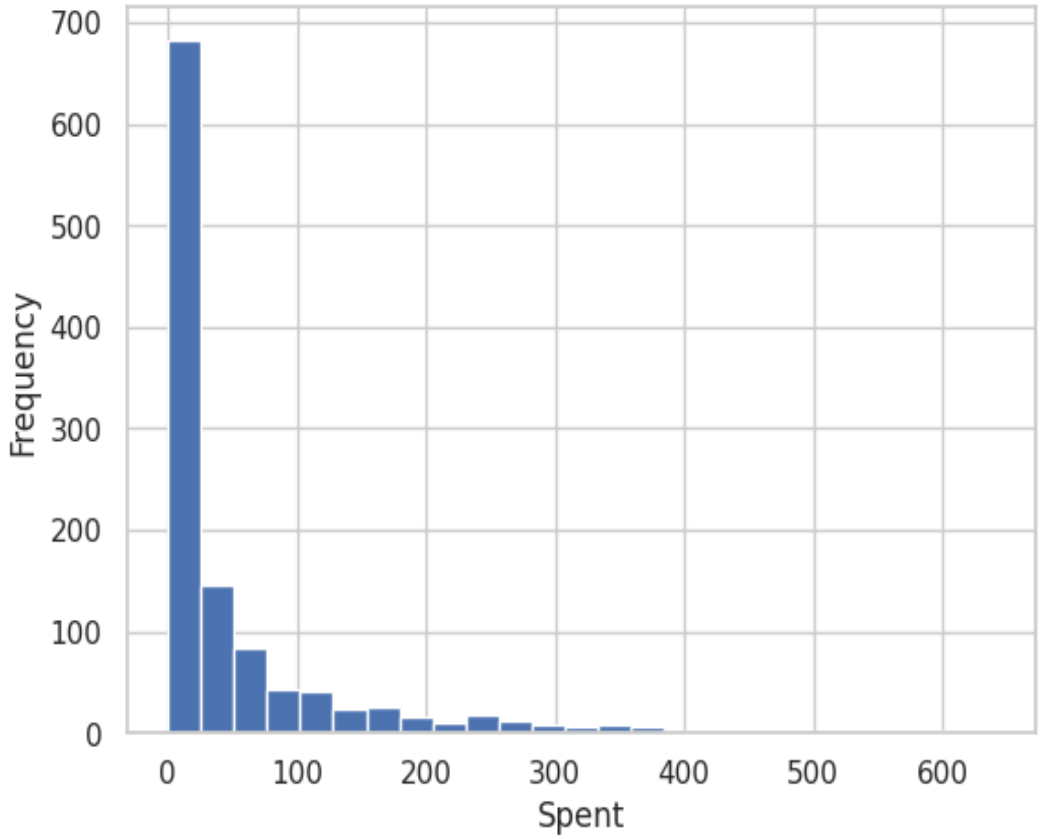
```
<Axes: xlabel='xyz_campaign_id', ylabel='Approved_Conversion'>
```



We created a grouped bar plot that shows the relationship between campaign ID, approved conversions, and gender. This allows us to see the gender distribution along each campaign. Both genders show almost similar interest in each campaign.

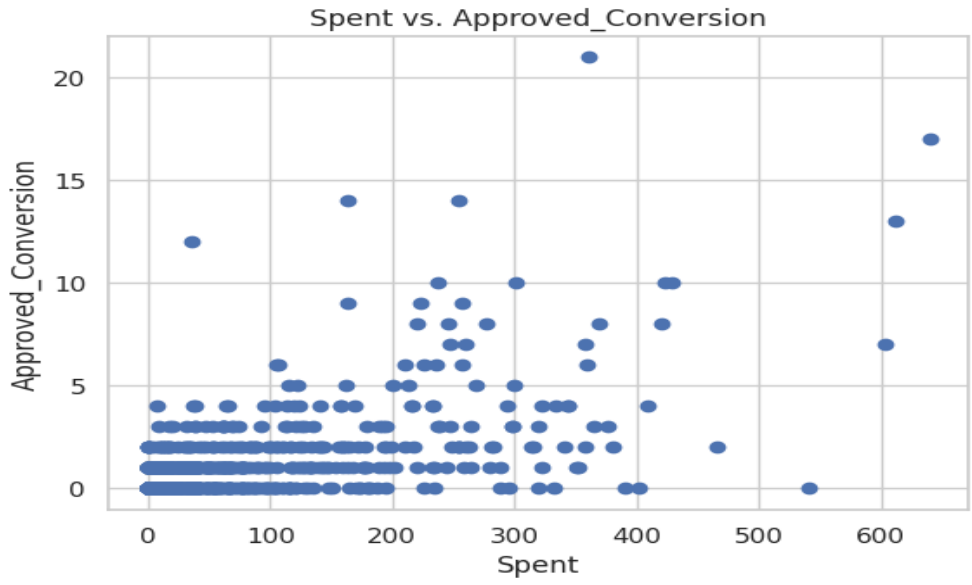
Spent

```
[ ]  plt.hist(df['Spent'], bins = 25)
     plt.xlabel("Spent")
     plt.ylabel("Frequency")
     plt.show()
```



We tried to plot a histogram, to see the number of data points in the campaigns spent at each spend rate.

```
# No. of products bought vs the acquistion cost
plt.scatter(df["Spent"], df["Approved_Conversion"])
plt.title("Spent vs. Approved_Conversion")
plt.xlabel("Spent")
plt.ylabel("Approved_Conversion")
plt.show()
```

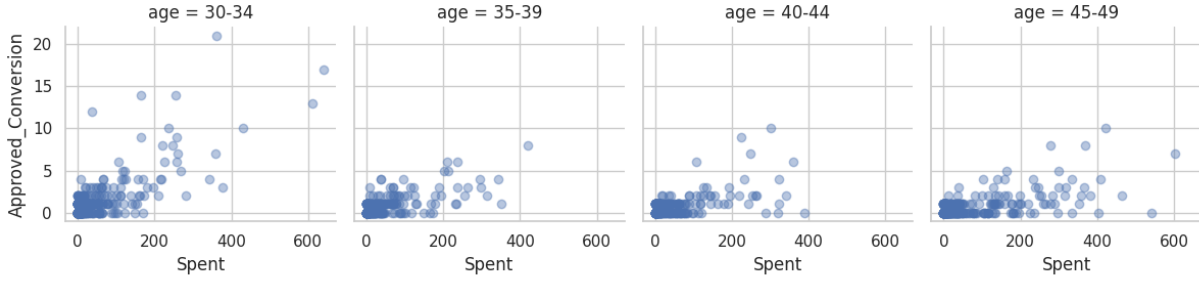


We can see, as the amount of money spent increases, number of products bought increases.

```
# Create a FacetGrid with age as the column variable
g = sns.FacetGrid(df, col="age")

# Map a scatter plot of Spent versus Approved_Conversion for each age group
g.map(plt.scatter, "Spent", "Approved_Conversion", alpha=.4)

# Add a legend to the plot
g.add_legend()
```



We created a grid of scatter plots(facet grid), with each plot showing the relationship between "Spent" and "Approved_Conversion" for all age groups.

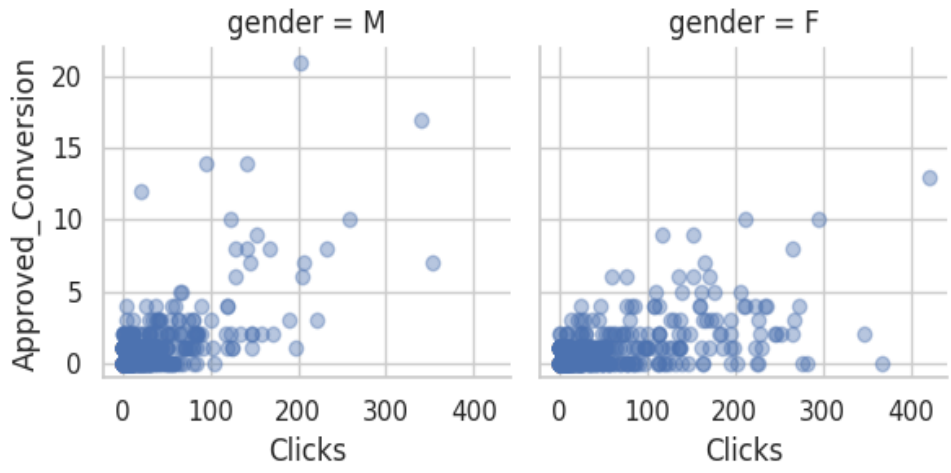
Now moving on to the vital attributes, “People who actually bought the product”.

People who actually bought the product

After Clicking the ad ?

Let's see people who actually went from clicking to buying the product.

```
[ ]  g = sns.FacetGrid(df, col="gender")
     g.map(plt.scatter, "Clicks", "Approved_Conversion", alpha=.4)
     g.add_legend();
```



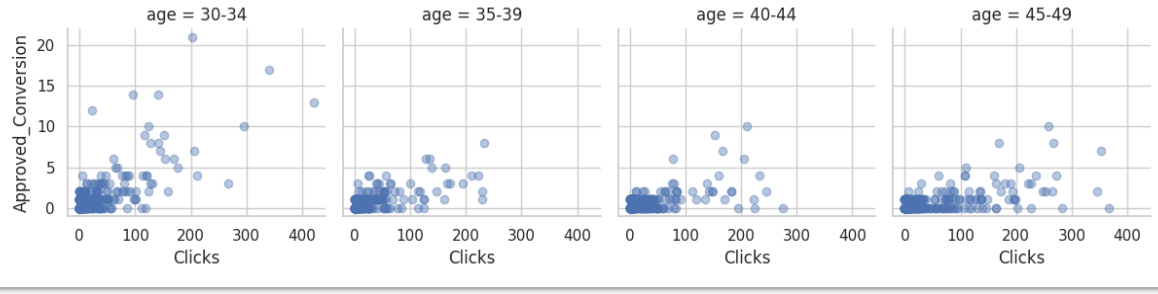
It seems men tend to click more than women but women buy more products than men after clicking the ad.

```
import seaborn as sns
import matplotlib.pyplot as plt

# Create a FacetGrid with age as the column variable
g = sns.FacetGrid(df, col="age")

# Map a scatter plot of Clicks versus Approved_Conversion for each age group
g.map(plt.scatter, "Clicks", "Approved_Conversion", alpha=.4)

# Add a legend to the plot
g.add_legend()
```



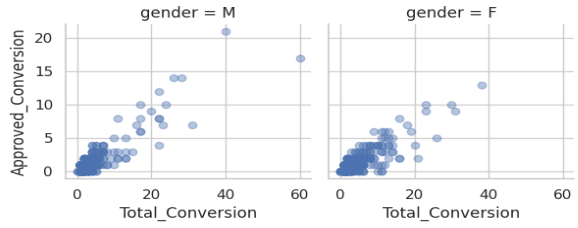
We created a grid of scatter plots, with each plot showing the relationship between "Clicks" and "Approved_Conversion" for a different age group.

People in the 30-34 age group have a greater tendency to buy products after clicking the ad.

After enquiring the product?

Let's see people who actually went from enquiring to buying the product.

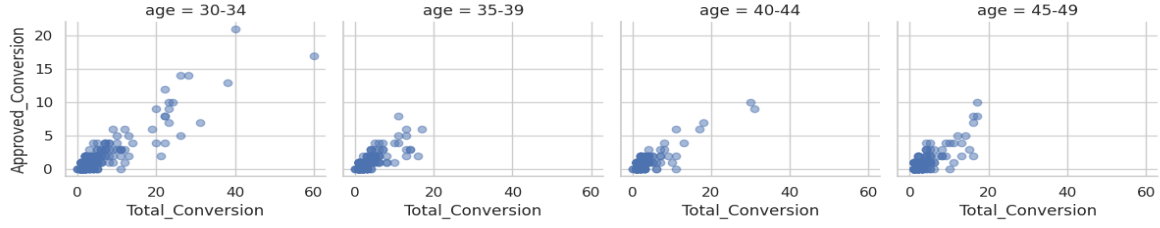
```
[ ] g = sns.FacetGrid(df, col="gender")
    g.map(plt.scatter, "Total_Conversion", "Approved_Conversion", alpha=.4)
    g.add_legend();
```



It seems women buys more products than men after enquiring the product. However men tends to enquire more about the product.

```
[ ] g = sns.FacetGrid(df, col="age")
    g.map(plt.scatter, "Total_Conversion", "Approved_Conversion",alpha=.5)
    g.add_legend()
```

```
<seaborn.axisgrid.FacetGrid at 0x7fef34eb5700>
```



It seems people in age group 30-34 are more likely to buy the product after enquiring the product.

Summary -

Correlations:

- "Impressions" and "Total_Conversion" are more correlated with "Approved_Conversion" than "Clicks" and "Spent".

Campaign_C:

1. campaign_c has the most number of ads.
2. compaign_c has a better Approved_conversion count, i.e. most people bought products in campaign_c.

Age_group:

3. In campaign_c and campaign_b, the age group of 30-34 shows more interest, whereas in campaign_a the age group of 40-44 shows more interest.

Gender:

4. Both the genders show similar interests in all three campaigns.

Interest:

5. Although the count of interest after 100 is less,there is a rise of users after 100 who actually bought the product. Rest of the distribution is according to what was expected.

Money spent:

6. As the amount of money spent increases, the number of products bought increases.
7. There is a sudden rise in the Approved_Conversion after a certain point in Impressions.

Product bought after clicking the ad:

8. It seems men tend to click more than women but women buy more products than men after clicking the ad.
9. People in the 30-34 age group have a greater tendency to buy products after clicking the ad.

Product bought after enquiring the ad:

10. It seems women buy more products than men after enquiring about the product. However men tend to enquire more about the product.
11. It seems people in the 30-34 age group are more likely to buy the product after enquiring about the product.

Instructive_conclusion:

12. For campaign_c, fb_campaign_ids around 145000 have more Approved_Conversion than around 180000

• Project Management

Implementation status report

Work completed:

• Description

Loading and Preprocessing of dataset

Performing exploratory data analysis

Exploring data using visualisations

• Responsibility (Task, Person)

Exploratory Data Analysis - Nagasai Gummadi, Divya Sri Vakkala

Data Visualization - Harsha Buddana, Eswara Reddy Thimmapuram

• Contributions (members/percentage)

Harsha Buddana - 25%

Nagasai Gummadi - 25%

Divya Sri Vakkala - 25%

Eswara Reddy Thimmapuram - 25%

§ Work to be completed

• Description

Creating a Regression model and deployment of web app

• Responsibility (Task, Person)

Creating a Regression model - Nagasai Gummadi, Divya Sri Vakkala

Data Visualization - Harsha Buddana, Eswara Reddy Thimmapuram

Harsha Buddana - 25%

Nagasai Gummadi - 25%

Divya Sri Vakkala - 25%

Eswara Reddy Thimmapuram - 25%

• Issues/Concerns:

• References/Bibliography:

1. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. O'Reilly Media.
2. VanderPlas, J. (2016). Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media.
3. Raschka, S., & Mirjalili, V. (2017). Python Machine Learning. Packt Publishing.
4. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd Ed.). Springer.
5. Subha, B. “Social Media Advertisement and Its Effect in Sales Prediction - An Analysis.” Shanlax International Journal of Management, vol. 8, no. 2, 2020, pp. 40–44.
6. Shi, Yuying and Karniouchina, Ekaterina and Uslay, Can, (When) Can Social Media Buzz Data Replace Traditional Surveys for Sales Forecasting? (April 1, 2020). Rutgers Business Review, Vol. 5, No. 1, 2020, pp.43-60.
7. Shi, Yuying and Karniouchina, Ekaterina and Uslay, Can, (When) Can Social Media Buzz Data Replace Traditional Surveys for Sales Forecasting? (April 1, 2020). Rutgers Business Review, Vol. 5, No. 1, 2020, pp.43-60.