# DATA ANALYSIS PORTFOLIO

PREPAIRED BY:

MAYANK KUMAR GANDHARV

# PROFESSIONAL BACKGROUND:

I am a MBA post graduate with a dual specialization in INTERNATIONAL BUSINESS AND BUSINES ANALYTICS, I also did my graduation in B.Tech Automobile (Integrated B.tech + MBA – 5yrs), 2022 Pass out. After, my MBA I prepared for Indian Armed Force and gave 3 attempts for the same and thus reached my satisfaction level, Now, I am reconsidering and preparing myself for a switch to data science world and upskilling myself for Data Analyst Profile.

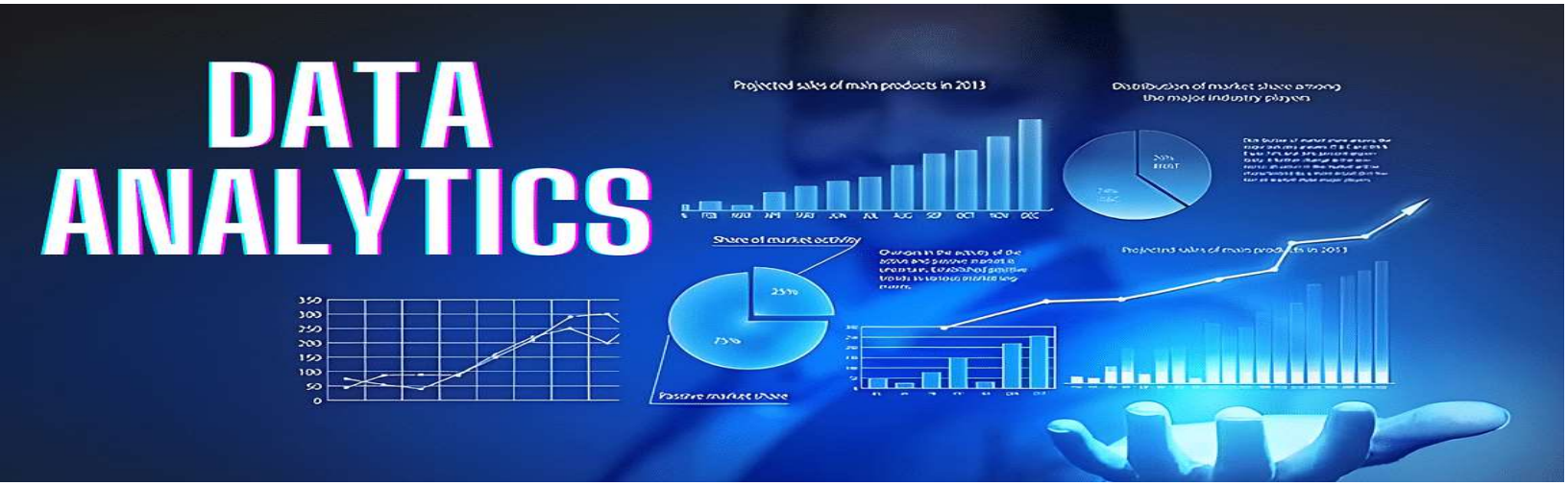
I worked with Calvin Klein for 9 months as a full time for Public Relations and brushed up my communication skills and know the work ethics for working in a team and under a supervisor.

I have also designed a Deca Copter for my final project of B.Tech, and worked on several projects related to Data Analytics, Machine Learning, 3-D modelling, etc.

Technically, I am fresher looking up to the world of DATA and want myself to see working as a Data Analyst for a top MNC. It would be great to experience the real challenges of the corporate world and understand how things work. Being a fresher, I think I am very flexible and adaptive to learn new things. I have theoretical knowledge. But I am waiting to use my theoretical knowledge in a practical way. And I believe by putting significant efforts I will learn. I have a very technical and analytical approach and I can use that for projects offered as a Data Analyst.

# TABLE OF CONTENTS






# MODULE 1- DATA ANALYTICS PROCESS

This process is basic yet crucial for data analytics fundamentals. There is no big data involved in this project, but the approach towards any data analytics process is followed. The process include PLAN, PREPARE, PROCESS, ANALYZE SHARE, ACT.

My task is to give the example(s) of such a real-life situation where we use Data Analytics and link it with the data analytics process. And so I used the data of Amazon sale where I have to find the best laptop with best specs and in the range (budget).

Objective: I want to purchase a Gaming Laptop, from AMAZON'S Great Festive Sale under 60,000 RS. I am confused about which laptop should I buy?

PLAN: Before buying any laptop, I will consider the laptop's specifications, pricing, discount, additional promotions, etc.

PREPARE: I will check my maximum budget i.e 60,000 Rs.

PROCESS (COLLECTING DATA): I will search 'Gaming laptop' on search bar and use the filter for budget.

Following I will get a list of gaming laptops on amazon (My data).

ANALYSE (CLEANING DATA): My main focus will be on CPU, GPU, Refresh Rate, RAM, SSD. And accordingly I will shortlist some laptops.

SHORTLISTED LAPTOPS:
1. ACER ASPIRE 5 (46,000 RS AFTER DISCOUNT)
2. LENOVO IDEAPAD GAMING 3 (56,000 RS AFTER DISC.)
3. ASUS TUF GAMING F-15 ( 53,000 RS AFTER DISC.)
4. HP VICTUS GAMING (59,000 RS AFTER DISC)

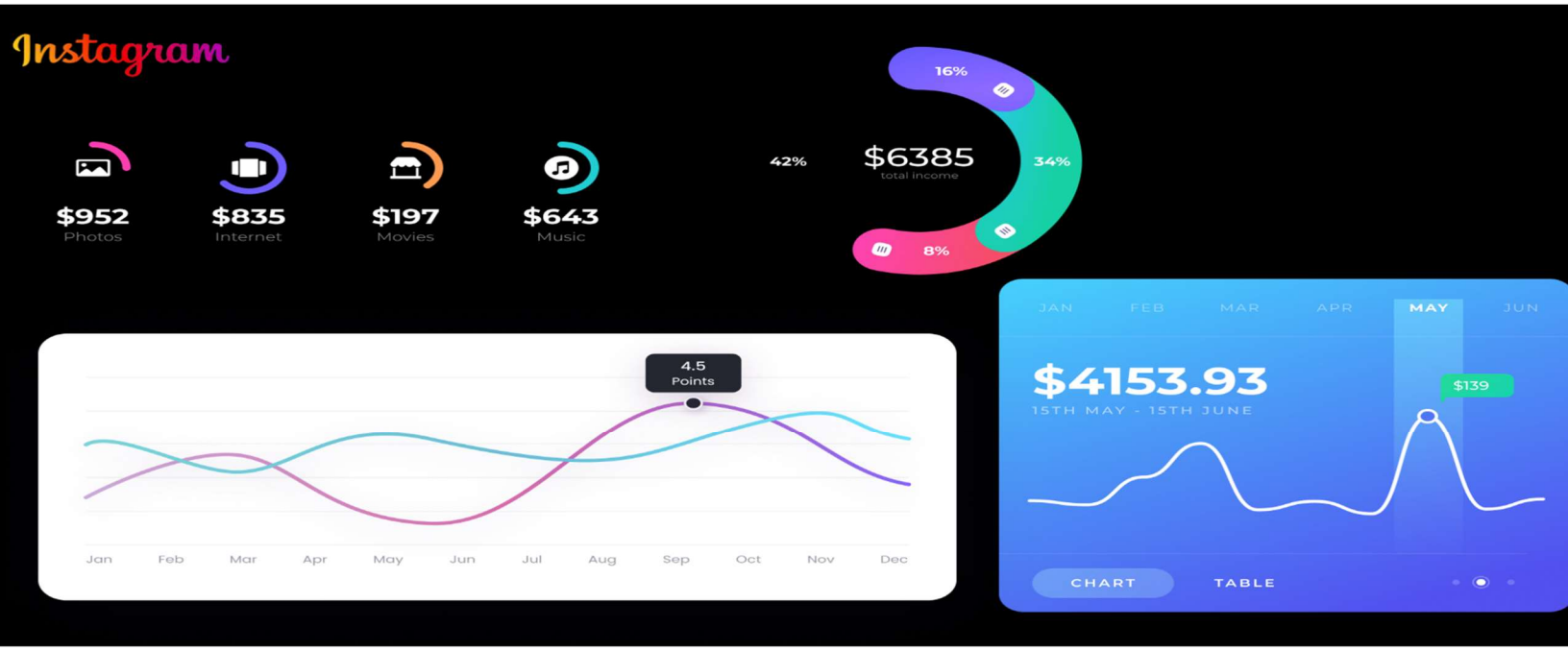
5. MSI GF63 (53,000 RS After disc.)

| NAME | CPU | GPU | RAM | SSD | REF. RATE |
|---|---|---|---|---|---|
| ACER | 12 GEN i5 | RTX 2050 | 16 GB | 512 GB | . |
| LENOVO | RYZ 7 5 GEN | RTX 3050 | 8 GB | 512 GB | 120 HZ |
| ASUS | 11 GEN | i5 RTX 2050 | 16 GB | 512 GB | 144HZ |
| HP | 12 GEN | i5 RTX 3050 | 16 GB | 512 GB | 144HZ |
| MSI | 11 GEN | i7 GTX 1650 | 16 GB | 512 GB | 144HZ |

SHARE: Best gaming laptop is based on best CPU and GPU configuration.

ACT: After analyzing and short listing 5 best laptops under 60,000 Rs, I will choose HP VICTUS Gaming

Laptop because it has the best processor and the best GPU combination of them all and it fits in the budget as well.




# MODULE 2- INSTAGRAM USER ANALYTICS

This project is totally done on My SQL and one task is done on Power Bi. The data is given in the form of SQL script that is copied and pasted on My SQL and executed. The tasks that were assigned as a Data Analyst were used to get various insights about the customers and how the social media platform can improve its approach towards the public.

My role involves analyzing user interactions and engagement with the Instagram app to provide valuable insights that can help the business grow.

User analysis involves tracking how users engage with a digital product, such as a software application or a mobile app. The insights derived from this analysis can be used by various teams within the business.

The tasks are as follows:

**A) Marketing Analysis:**

1. **Loyal User Reward:** The marketing team wants to reward the most loyal users, i.e., those who have been using the platform for the longest time.
   **Task:** Identify the five oldest users on Instagram from the provided database.

**SQL QUERY:**

```
select * from users
order by created_at
limit 5;
```

**OUTPUT:**

| id | username |
|---|---|
| 80 | Darby_Herzog |
| 67 | Emilio_Bernier52 |
| 63 | Elenor88 |
| 95 | Nicole71 |
| 38 | Jordyn.Jacobson2 |

2. **Inactive User Engagement:** The team wants to encourage inactive users to start posting by sending them promotional emails.
   **Task:** Identify users who have never posted a single photo on Instagram.

   **SQL QUERY:**

```
select * from photos
right join users
on photos.user_id=users.id
where user_id is null;
```

**OUTPUT:**

| id | username |
|---|---|
| 5 | Aniya_Hackett |
| 7 | Kasandra_Homenick |
| 14 | Jaclyn81 |
| 21 | Rocio33 |
| 24 | Maxwell.Halvorson |
| 25 | Tierra.Trantow |
| 34 | Pearl7 |
| 36 | Ollie_Ledner37 |
| 41 | Mckenna17 |
| 45 | David.Osinski47 |
| 49 | Morgan.Kassulke |
| 53 | Linnea59 |
| 54 | Duane60 |
| 57 | Julien_Schmidt |
| 66 | Mike.Auer39 |
| 68 | Franco_Keebler64 |
| 71 | Nia_Haag |
| 74 | Hulda.Macejkovic |
| 75 | Leslie67 |
| 76 | Janelle.Nikolaus81 |
| 80 | Darby_Herzog |
| 81 | Esther.Zulauf61 |
| 83 | Bartholome.Bernhard |
| 89 | Jessyca_West |
| 90 | Esmeralda.Mraz57 |
| 91 | Bethany20 |

3. **Contest Winner Declaration:** The team has organized a contest where the user with the most likes on a single photo wins.
   **Task:** Determine the winner of the contest and provide their details to the team.

   **SQL QUERY:**

```
with cte as (select photo_id, count(user_id) as likes from likes
group by photo_id
order by count(user_id) desc)
select * from cte
left join photos
on cte.photo_id = photos.id
join users
on photos.user_id = users.id;
```

**OUTPUT:**

| photo_id | likes | user id | username |
|---|---|---|---|
| 145 | 48 | 52 | Zack_Kemmer93 |
| 127 | 43 | 46 | Malinda_Streich |
| 182 | 43 | 65 | Adelle96 |
| 123 | 42 | 44 | Seth46 |

4. **Hashtag Research:** A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.
   **Task**: Identify and suggest the top five most commonly used hashtags on the platform.

   **SQL QUERY:**

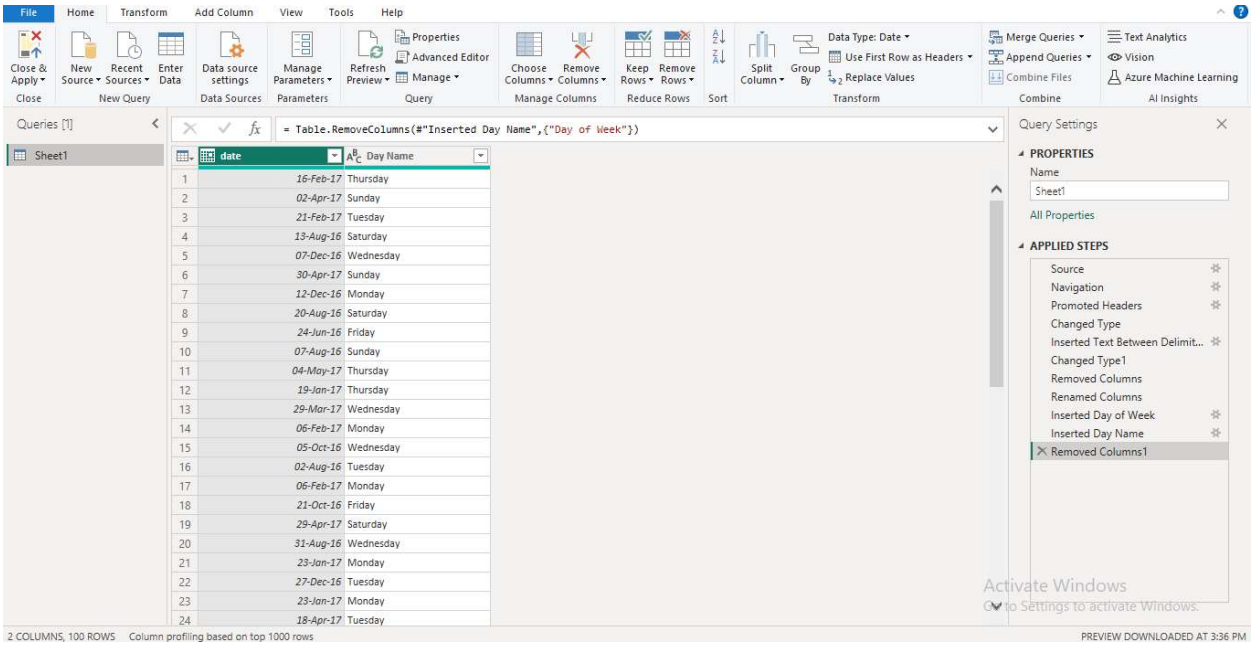
```
with cte as (select tag_id, count(photo_id) tags from photo_tags
group by tag_id
order by tags desc)
select * from cte
left join tags
on cte.tag_id = tags.id
limit 5;
```

   **OUTPUT:**

| tag_id | count(photo_id) | tag_name |
|---|---|---|
| 21 | 59 | smile |
| 20 | 42 | beach |
| 17 | 39 | party |
| 13 | 38 | fun |
| 5 | 24 | food |

5. **Ad Campaign Launch:** The team wants to know the best day of the week to launch ads. **Task:** Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

   FOR THIS TASK POWERBI IS BEEN USED:




| Day Name | Count of Day Name |
|---|---|
| Sunday | 16 |
| Thursday | 16 |
| Friday | 15 |
| Monday | 14 |
| Tuesday | 14 |
| Wednesday | 13 |
| Saturday | 12 |

**INFERENCE:** Most users register on SUNDAY and THURSDAY. According to me an ad campaign should be held on **Thursday and Sunday**

**B) Investor Metrics:**

1. **User Engagement:** Investors want to know if users are still active and posting on Instagram or if they are making fewer posts.
   **Task**: Calculate the average number of posts per user on Instagram. Also, provide the total number of photos on Instagram divided by the total number of users.

   **SQL QUERY:**

```
Select avg ceil(posts)
from (select user_id, count(user_id) as posts from photos
group by user_id) as avg_post;
```

   **OUTPUT:**
   **4**

   **For total number of photos per total number of users**

   **SQL QUERY:**

```
select ((select count(id) from photos
)/(
select count(id) from users)) as total_photos_per_total_users  ;
```

   **OUTPUT:**

   **2.57**

2. **Bots & Fake Accounts:** Investors want to know if the platform is crowded with fake and dummy accounts.
   **Task**: Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

**SQL QUERY:**

```
with cte as (select user_id, count(photo_id) from likes
group by user_id
having likes = 257
order by count(photo_id) desc)
select * from cte
left join users
on cte.user_id = users.id ;
```

**OUTPUT:**

| username | user_id | likes |
|---|---|---|
| Aniya_Hackett | 5 | 257 |
| Jaclyn81 | 14 | 257 |
| Rocio33 | 21 | 257 |
| Maxwell.Halvorson | 24 | 257 |
| Ollie_Ledner37 | 36 | 257 |
| Mckenna17 | 41 | 257 |
| Duane60 | 54 | 257 |
| Julien_Schmidt | 57 | 257 |
| Mike.Auer39 | 66 | 257 |
| Nia_Haag | 71 | 257 |
| Leslie67 | 75 | 257 |
| Janelle.Nikolaus81 | 76 | 257 |
| Bethany20 | 91 | 257 |

LEARNINGS:

This project helped me to learn the structured query language efficiently, functions like join, aggregations, Common table expressions, is now something that I can use based on the query I am writing. The project really helped me to model the data and to derive the data driven insights thoroughly.


Module 3:

# Operation Analytics and Investigating Metric Spike

Operational Analytics is a crucial process that involves analyzing a company's end-to-end operations. This analysis helps identify areas for improvement within the company. As a Data Analyst, onel work closely with various teams, such as operations, support, and marketing, helping them derive valuable insights from the data they collect.

One of the key aspects of Operational Analytics is investigating metric spikes. This involves understanding and explaining sudden changes in key metrics, such as a dip in daily user engagement or a drop in sales.

**CASE STUDY 1: JOB DATA ANALYSIS**

**Creating Database and Tables:**

First I imported the provided database of case study 1 in MySQL.

```
use casestudy1_proj3;
# creating table for case study 1 ds    job_id  actor_id    event    language    time_spent  org

create table job_data (
ds varchar(50) ,
job_id int ,
actor_id int ,
event varchar(10) ,
language varchar(10) ,
time_spent int,
org varchar(10)
);

load data infile "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/job_data.csv"
into table job_data
fields terminated by ','
enclosed by '"'
lines terminated by '\n'
ignore 1 rows;
```

After importing the dataset by creating the table job_data, I further did my analysis using MySQL Workbench.

1. **Jobs Reviewed Over Time:**

- Objective: Calculate the number of jobs reviewed per hour for each day in November 2020.
- Task: Write an SQL query to calculate the number of jobs reviewed per hour for each day in November 2020

**SQL QUERY:**

```
1  select * from job_data;
2  select ds, count(job_id) *3600/sum(time_spent) as number_of_jobs_reviewed_per_hour
3  from job_data
4  group by ds
5  order by ds;
```

**OUTPUT:**

| ds | number_of_jobs_reviewed_per_hour |
|---|---|
| 25-11-20 | 80.0000 |
| 26-11-20 | 64.2857 |
| 27-11-20 | 34.6154 |
| 28-11-20 | 218.1818 |
| 29-11-20 | 180.0000 |
| 30-11-20 | 180.0000 |

2. **Throughput Analysis:**

   a. **Objective**: Calculate the 7-day rolling average of throughput (number of events per second).

   **b. Task:** Write an SQL query to calculate the 7-day rolling average of throughput. Additionally, explain whether you prefer using the daily metric or the 7-day rolling average for throughput, and why.

**SQL QUERY:**

```
1  select * from job_data;
2  select ds, throughput, avg(throughput)
3  over(order by ds rows between 6 preceding and current row) as 7_day_rolling_avg
4  from(
5  select ds, count(event)/sum(time_spent) as throughput  from job_data
6  group by ds
7  order by ds) as Throughput_calculation;
8
```

**OUTPUT:**

| ds | throughput | 7_day_rolling_avg |
|---|---|---|
| 25-11-20 | 0.0222 | 0.02220000 |
| 26-11-20 | 0.0179 | 0.02005000 |
| 27-11-20 | 0.0096 | 0.01656667 |
| 28-11-20 | 0.0606 | 0.02757500 |
| 29-11-20 | 0.0500 | 0.03206000 |
| 30-11-20 | 0.0500 | 0.03505000 |

3. **Language Share Analysis:**

    a. Objective: Calculate the percentage share of each language in the last 30 days.
    b. Task: Write an SQL query to calculate the percentage share of each language over the last 30 days.

**SQL QUERY:**

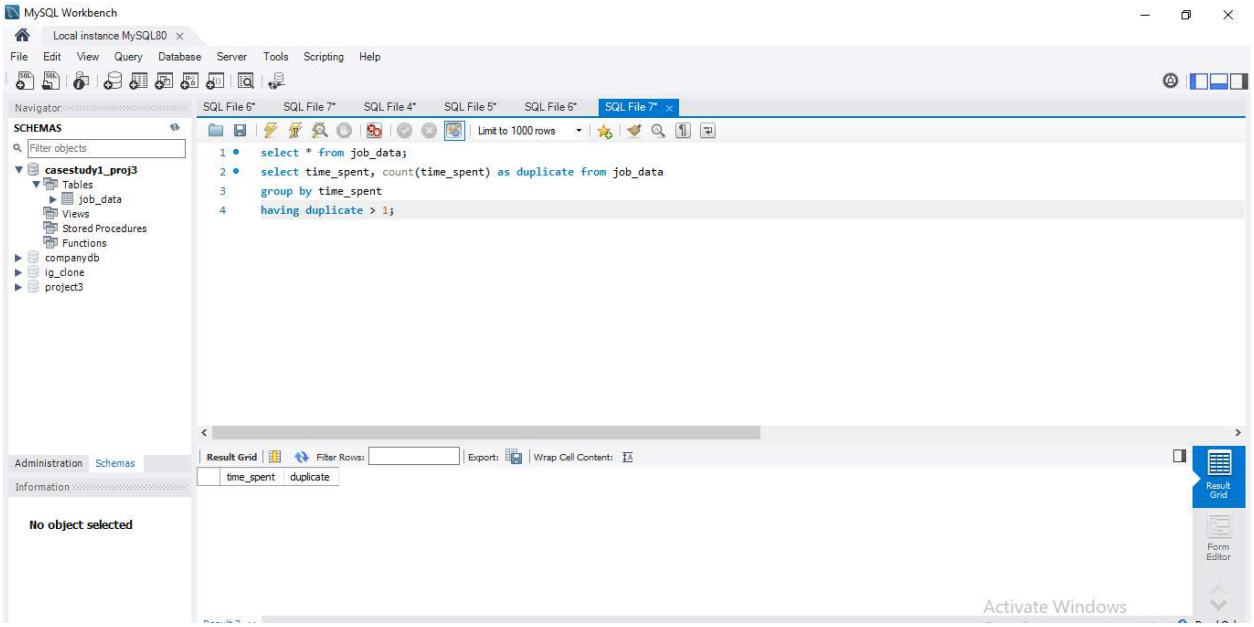
```
use casestudy1_proj3;
select count(language) from job_data;
select language, count(language), round(count(language)*100/8, 2) as percentage_share
FROM job_data
group by language;
```

**OUTPUT:**

| language | count(language) | percentage_share |
|---|---|---|
| English | 1 | 12.50 |
| Arabic | 1 | 12.50 |
| Persian | 3 | 37.50 |
| Hindi | 1 | 12.50 |
| French | 1 | 12.50 |
| Italian | 1 | 12.50 |

4. **Duplicate Rows Detection:**

   - **Objective**: Identify duplicate rows in the data.
   - **Task**: Write an SQL query to display duplicate rows from the job_data table.




**OUTPUT:**

No Duplicate value found.

# Case Study 2: Investigating Metric Spike

A. **Weekly User Engagement:**
   - Objective: Measure the activeness of users on a weekly basis.
   - Your Task: Write an SQL query to calculate the weekly user engagement

SQL QUERY:

```
use project3;
select * from events;
select extract(week from occured_at) as number_of_week,
count(distinct(user_id)) as number_of_users
from events
where event_type = 'engagement'
group by number_of_week
order by number_of_users;
```

**OUTPUT:**

| number_of_week | number_of_users |
|---|---|
| 17 | 663 |
| 18 | 1068 |
| 19 | 1113 |
| 20 | 1154 |
| 21 | 1121 |
| 22 | 1186 |
| 23 | 1232 |
| 24 | 1275 |
| 25 | 1264 |
| 26 | 1302 |
| 27 | 1372 |
| 28 | 1365 |
| 29 | 1376 |
| 30 | 1467 |
| 31 | 1299 |
| 32 | 1225 |
| 33 | 1225 |
| 34 | 1204 |
| 35 | 104 |

B. **User Growth Analysis:**

- Objective: Analyze the growth of users over time for a product.
- Your Task: Write an SQL query to calculate the user growth for the product.

**SQL QUERY:**

```
use project3;
select * from users;
select year_ , week_number,activated_users, sum(activated_users) over( order by year_ , week_number) as total_users
from(
select extract(year from activated_at) as year_, extract(week from activated_at) as week_number, count(distinct user_id) as activated_users
from users
group by year_, week_number
order by year_, week_number) as inner_query;
```

OUTPUT:

| year_ | week_number | activated_users | total_users |
|---|---|---|---|
| 2013 | 0 | 23 | 23 |
| 2013 | 1 | 30 | 53 |
| 2013 | 2 | 48 | 101 |
| 2013 | 3 | 36 | 137 |
| 2013 | 4 | 30 | 167 |
| 2013 | 5 | 48 | 215 |
| 2013 | 6 | 38 | 253 |
| 2013 | 7 | 42 | 295 |
| 2013 | 8 | 34 | 329 |
| 2013 | 9 | 43 | 372 |
| 2013 | 10 | 32 | 404 |
| 2013 | 11 | 31 | 435 |
| 2013 | 12 | 33 | 468 |
| 2013 | 13 | 39 | 507 |
| 2013 | 14 | 35 | 542 |
| 2013 | 15 | 43 | 585 |
| 2013 | 16 | 46 | 631 |
| 2013 | 17 | 49 | 680 |
| 2013 | 18 | 44 | 724 |
| 2013 | 19 | 57 | 781 |
| 2013 | 20 | 39 | 820 |
| 2013 | 21 | 49 | 869 |

| | | | |
|---|---|---|---|
| 2013 | 22 | 54 | 923 |
| 2013 | 23 | 50 | 973 |
| 2013 | 24 | 45 | 1018 |
| 2013 | 25 | 57 | 1075 |
| 2013 | 26 | 56 | 1131 |
| 2013 | 27 | 52 | 1183 |
| 2013 | 28 | 72 | 1255 |
| 2013 | 29 | 67 | 1322 |
| 2013 | 30 | 67 | 1389 |
| 2013 | 31 | 67 | 1456 |
| 2013 | 32 | 71 | 1527 |
| 2013 | 33 | 73 | 1600 |
| 2013 | 34 | 78 | 1678 |
| 2013 | 35 | 63 | 1741 |
| 2013 | 36 | 72 | 1813 |
| 2013 | 37 | 85 | 1898 |
| 2013 | 38 | 90 | 1988 |
| 2013 | 39 | 84 | 2072 |
| 2013 | 40 | 87 | 2159 |
| 2013 | 41 | 73 | 2232 |
| 2013 | 42 | 99 | 2331 |
| 2013 | 43 | 89 | 2420 |
| 2013 | 44 | 96 | 2516 |
| 2013 | 45 | 91 | 2607 |
| 2013 | 46 | 88 | 2695 |
| 2013 | 47 | 102 | 2797 |
| 2013 | 48 | 97 | 2894 |
| 2013 | 49 | 116 | 3010 |
| 2013 | 50 | 124 | 3134 |
| 2013 | 51 | 102 | 3236 |
| 2013 | 52 | 47 | 3283 |
| 2014 | 0 | 83 | 3366 |
| 2014 | 1 | 126 | 3492 |
| 2014 | 2 | 109 | 3601 |
| 2014 | 3 | 113 | 3714 |
| 2014 | 4 | 130 | 3844 |
| 2014 | 5 | 133 | 3977 |
| 2014 | 6 | 135 | 4112 |
| 2014 | 7 | 125 | 4237 |
| 2014 | 8 | 129 | 4366 |
| 2014 | 9 | 133 | 4499 |
| 2014 | 10 | 154 | 4653 |
| 2014 | 11 | 130 | 4783 |

| | | | |
|---|---|---|---|
| 2014 | 12 | 148 | 4931 |
| 2014 | 13 | 167 | 5098 |
| 2014 | 14 | 162 | 5260 |
| 2014 | 15 | 164 | 5424 |
| 2014 | 16 | 179 | 5603 |
| 2014 | 17 | 170 | 5773 |
| 2014 | 18 | 163 | 5936 |
| 2014 | 19 | 185 | 6121 |
| 2014 | 20 | 176 | 6297 |
| 2014 | 21 | 183 | 6480 |
| 2014 | 22 | 196 | 6676 |
| 2014 | 23 | 196 | 6872 |
| 2014 | 24 | 229 | 7101 |
| 2014 | 25 | 207 | 7308 |
| 2014 | 26 | 201 | 7509 |
| 2014 | 27 | 222 | 7731 |
| 2014 | 28 | 215 | 7946 |
| 2014 | 29 | 221 | 8167 |
| 2014 | 30 | 238 | 8405 |
| 2014 | 31 | 193 | 8598 |
| 2014 | 32 | 245 | 8843 |
| 2014 | 33 | 261 | 9104 |
| 2014 | 34 | 259 | 9363 |
| 2014 | 35 | 18 | 9381 |

INSIGHTS:

The **maximum** number of users growth was seen in **33rd week of 2014** and **minimum** users growth in **35th week of 2014**.

C. **Weekly Retention Analysis:**

- Objective: Analyze the retention of users on a weekly basis after signing up for a product.
- Your Task: Write an SQL query to calculate the weekly retention of users based on their sign-up cohort

SQL QUERY:

```
select * from events;
select user_id1, count(user_id1) as total_retention_week from (
with left_table as (select distinct user_id, extract(week from occured_at) as sighnup_week from events
where event_name='complete_signup'
and event_type='signup_flow'),
right_table as (
select distinct user_id as user_id1, extract(week from occured_at) as retention_week from events
where event_type='engagement')
select * from left_table
left join right_table
on left_table.user_id=right_table.user_id1
order by left_table.user_id) as cte
group by user_id
order by total_retention_week desc;
```

D. **Weekly Engagement Per Device:**

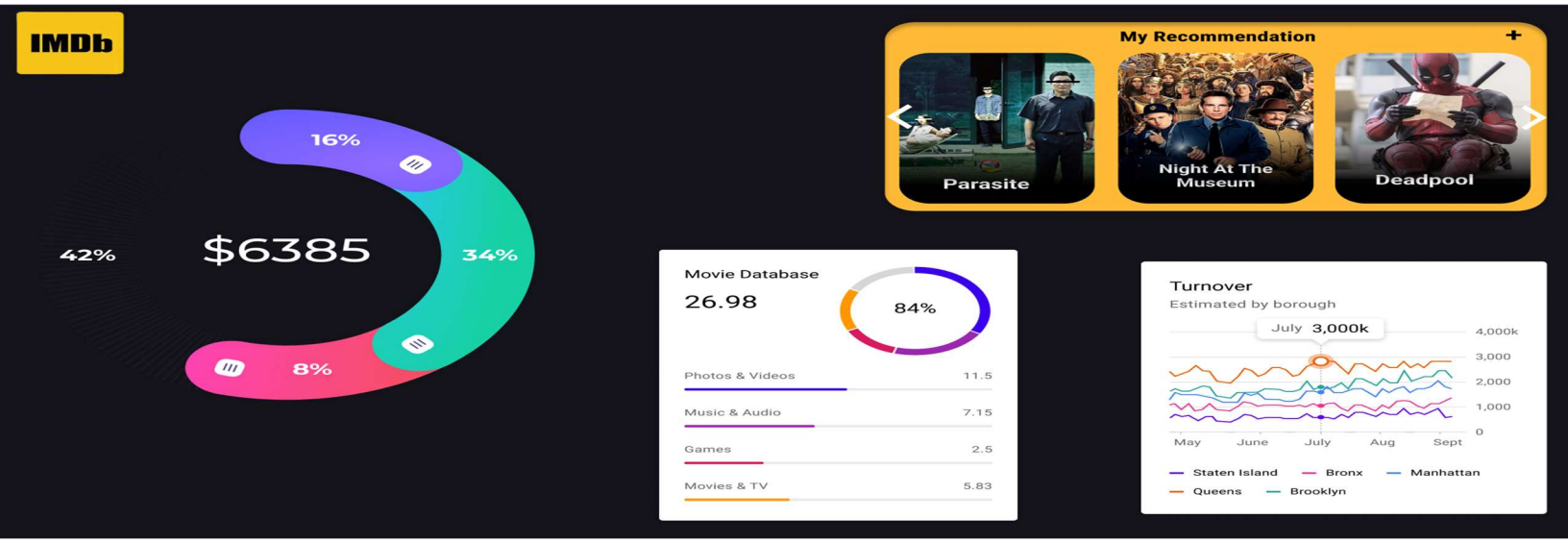
- Objective: Measure the activeness of users on a weekly basis per device.
- Your Task: Write an SQL query to calculate the weekly engagement per device.

SQL QUERY:

```
use project3;
select * from events;
select device, extract(year from occured_at) as year_no, extract(week from occured_at) as week_no, count(distinct user_id) as users from events
where event_type='engagement'
group by week_no, year_no, device
order by users desc ;
```

**INSIGHTS:**

The total email sent to users (sent_weekly_digest + sent_reengagement_email) are 60,920 out of wich only 20,459 are opened and 9010 were clicked through the opened mail.




# MODULE 4:IMDB Movie Analysis

**Problem Statement**: The dataset provided is related to IMDB Movies. A potential problem to investigate could be: "What factors influence the success of a movie on IMDB?" Here, success can be defined by high IMDB ratings. The impact of this problem is significant for movie producers, directors, and investors who want to understand what makes a movie successful to make informed decisions in their future projects.

**DATA CLEANING:** To clean data I used power query in Power BI as well Excel and table format in Excel to eliminate Duplicate Data. There are various columns that plays no role in the analysis but still I kept them in the model except column “Color”. The movie_title has letter “Â” at the end of each title row so, in order to eliminate it Split Column by Delimiter is been used (shown in screenshot)

**Data Analytics Tasks:**

A. **Movie Genre Analysis**: Analyze the distribution of movie genres and their impact on the IMDB score**.**

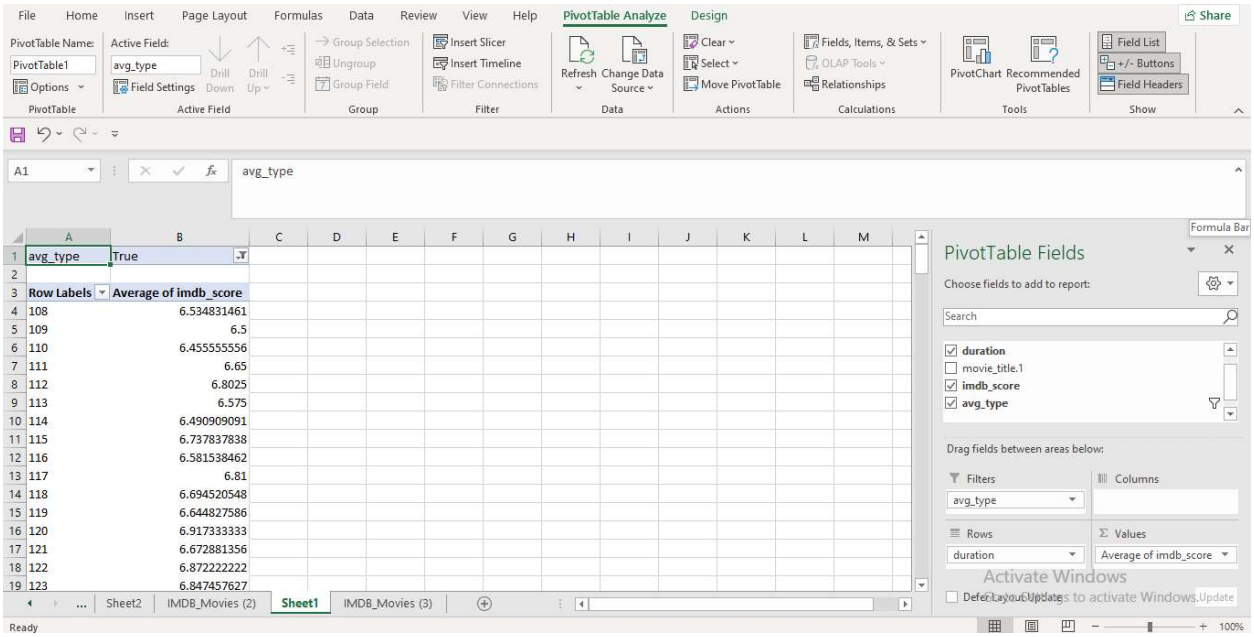
| Genre | Count | Max | Min | Average | Variance | StdDev | Median | Range | mode |
|---|---|---|---|---|---|---|---|---|---|
| Action | 1113 | 9.1 | 1.7 | 6.231626 | 1.252668 | 1.119226495 | 6.3 | 7.4 | 6.6 |
| Adventure | 888 | 8.9 | 1.9 | 6.436712 | 1.291458 | 1.136423313 | 6.6 | 7 | 6.7 |
| Animation | 240 | 8.6 | 1.7 | 6.575 | 1.309414 | 1.144296389 | 6.7 | 6.9 | 6.7 |
| Biography | 291 | 8.9 | 4.5 | 7.148797 | 0.525197 | 0.724704646 | 7.2 | 4.4 | 7 |
| Comedy | 1848 | 9.5 | 1.7 | 6.192857 | 1.190875 | 1.091272091 | 6.3 | 7.8 | 6.7 |
| Crime | 869 | 9.3 | 2.4 | 6.563061 | 1.059037 | 1.029095313 | 6.6 | 6.9 | 6.6 |
| Documentary | 121 | 8.7 | 1.6 | 7.180165 | 1.11627 | 1.056536782 | 7.4 | 7.1 | 7.5 |
| Drama | 2536 | 9.3 | 2 | 6.7653 | 0.909415 | 0.953632394 | 6.9 | 7.3 | 7.2 |
| Family | 534 | 8.7 | 1.7 | 6.23764 | 1.46419 | 1.210037311 | 6.4 | 7 | 6.7 |
| Fantasy | 583 | 8.9 | 1.7 | 6.302744 | 1.362054 | 1.167070825 | 6.4 | 7.2 | 6.7 |
| Film-Noir | 6 | 8.2 | 7.1 | 7.633333 | 0.186667 | 0.43204938 | 7.65 | 1.1 | #N/A |
| Game-Show | 1 | 2.9 | 2.9 | 2.9 | #DIV/0! | #DIV/0! | 2.9 | 0 | #VALUE! |
| History | 203 | 8.9 | 2 | 7.085714 | 0.786775 | 0.887003442 | 7.2 | 6.9 | 7.5 |
| Horror | 540 | 8.7 | 2.2 | 5.80463 | 1.255285 | 1.120394863 | 5.9 | 6.5 | 6.2 |
| Music | 212 | 8.5 | 1.6 | 6.406132 | 1.39982 | 1.183139907 | 6.6 | 6.9 | 6.5 |
| Musical | 131 | 8.5 | 2.1 | 6.500763 | 1.507769 | 1.227912311 | 6.7 | 6.4 | 7 |
| Mystery | 485 | 8.6 | 2.2 | 6.483918 | 1.174121 | 1.083568634 | 6.6 | 6.4 | 6.6 |
| News | 3 | 8.1 | 7.1 | 7.533333 | 0.263333 | 0.513160144 | 7.4 | 1 | #N/A |
| Reality-TV | 2 | 6.6 | 2.9 | 4.75 | 6.845 | 2.61629509 | 4.75 | 3.7 | #N/A |
| Romance | 1083 | 8.6 | 2.1 | 6.446076 | 0.997182 | 0.998589959 | 6.5 | 6.5 | 6.5 |
| Sci-Fi | 594 | 8.8 | 1.9 | 6.277778 | 1.480349 | 1.216695735 | 6.4 | 6.9 | 6.7 |
| Short | 5 | 7.1 | 5.2 | 6.38 | 0.557 | 0.746324326 | 6.5 | 1.9 | #N/A |
| Sport | 177 | 8.7 | 2 | 6.60113 | 1.230453 | 1.109257978 | 6.8 | 6.7 | 7.2 |
| Thriller | 1361 | 9 | 2.2 | 6.309111 | 1.116652 | 1.056717665 | 6.4 | 6.8 | 6.4 |
| War | 210 | 8.6 | 2.7 | 7.070952 | 0.767238 | 0.875921414 | 7.1 | 5.9 | 7.1 |
| Western | 94 | 8.9 | 3.8 | 6.703191 | 1.114506 | 1.055701584 | 6.8 | 5.1 | 6.8 |
| **Grand Total** | **14130** | **9.5** | **1.6** | **6.447898** | **1.214286** | **1.101946321** | **6.6** | **7.9** | **6.7** |

**INSIGHT:**

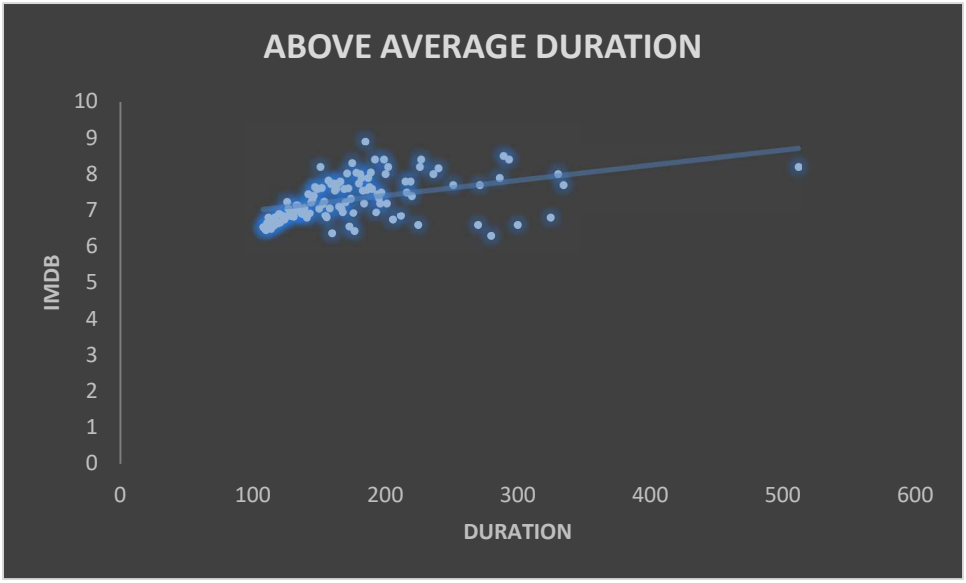
1. “Drama” genre is the most occurring movie genre in the dataset provided (2571 movies), followed by Comedy, Thriller, Action, Romance being the top 5 movie genre.
2. The most average IMDB Rating is given to the genre “Film-Noir” i.e 7.63 but there are very less movies for this genre. So for top 5 genre selection “Drama” is again the highest average IMDB Rated (6.76).

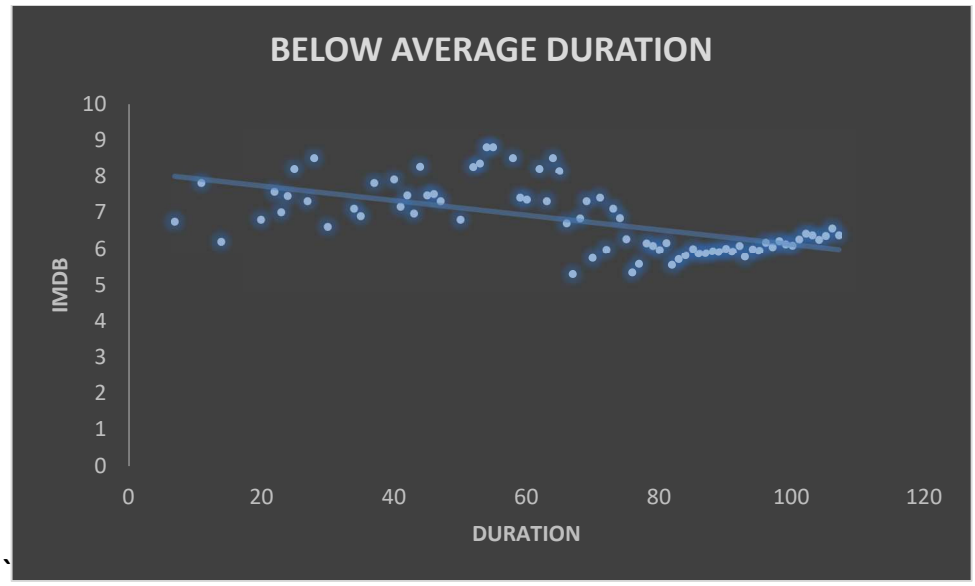
3. The highest IMDB rating is given to the movie: Towering Inferno (9.5) and the genre is “Comedy”.

B. **Movie Duration Analysis:** Analyze the distribution of movie durations and its impact on the IMDB score.

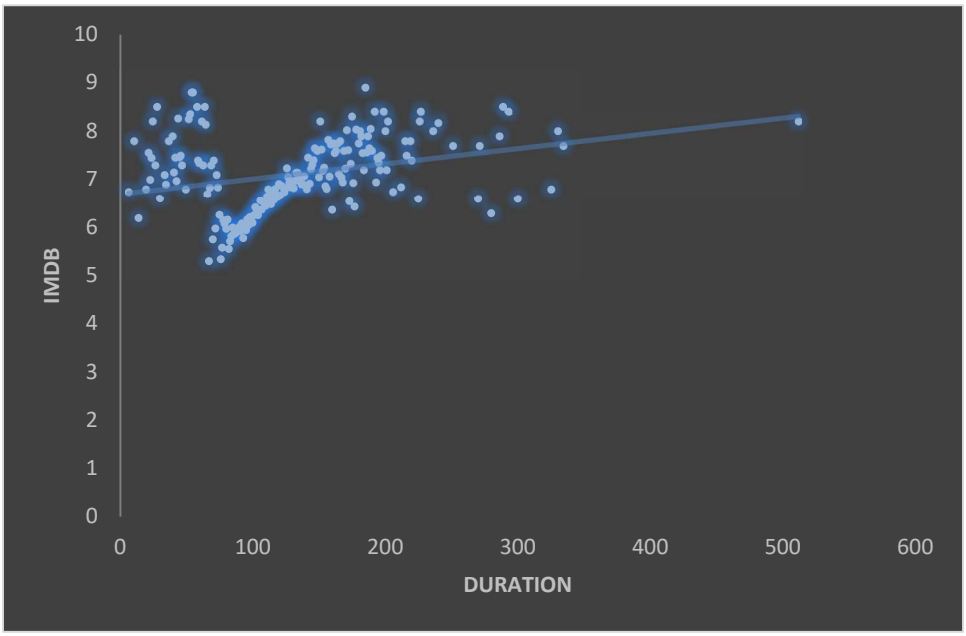


| avg_type | True |
|---|---|
| | |
| Row Labels | Average of imdb_score |
| 108 | 6.534831461 |
| 109 | 6.5 |
| 110 | 6.455555556 |
| 111 | 6.65 |
| 112 | 6.8025 |
| 113 | 6.575 |
| 114 | 6.490909091 |
| 115 | 6.737837838 |
| 116 | 6.581538462 |
| 117 | 6.81 |
| 118 | 6.694520548 |
| 119 | 6.644827586 |
| 120 | 6.917333333 |
| 121 | 6.672881356 |
| 122 | 6.872222222 |
| 123 | 6.847457627 |






`




**OUTPUT:**

Here the trend is positive but first there is a sudden drop and then the graph is increasing ( can be analyzed from the scatter plot). The correlation is 0.3.

**INSIGHT:**

It can be seen that movies with duration above 107 follows a positive trendline i.e with increasing movie duration the imdb also increases. But, movies below or equal to 107 have a negative trendline i.e with increasing movie duration the imdb decreases.

C. **Language Analysis:** Situation: Examine the distribution of movies based on their language.

| language | Count of movie_title | Average | StdDev | Median |
|---|---|---|---|---|
| Aboriginal | 2 | 6.95 | 0.777817459 | 6.95 |
| Arabic | 5 | 7.38 | 0.884307639 | 7.4 |
| Aramaic | 1 | 7.1 | #DIV/0! | 7.1 |
| Bosnian | 1 | 4.3 | #DIV/0! | 4.3 |
| Cantonese | 11 | 6.954545455 | 0.704788814 | 7.2 |
| Chinese | 3 | 5.666666667 | 0.550757055 | 5.7 |
| Czech | 1 | 7.4 | #DIV/0! | 7.4 |
| Danish | 5 | 7.5 | 1.077032961 | 8.1 |
| Dari | 2 | 7.5 | 0.141421356 | 7.5 |
| Dutch | 4 | 7.425 | 0.434932945 | 7.45 |
| Dzongkha | 1 | 7.5 | #DIV/0! | 7.5 |
| English | 4585 | 6.393740458 | 1.125155637 | 6.5 |
| Filipino | 1 | 6.7 | #DIV/0! | 6.7 |
| French | 73 | 7.038356164 | 0.726985812 | 7.2 |
| German | 19 | 7.342105263 | 0.954123093 | 7.6 |
| Greek | 1 | 7.3 | #DIV/0! | 7.3 |
| Hebrew | 5 | 7.58 | 0.334664011 | 7.6 |
| Hindi | 28 | 6.632142857 | 1.398955582 | 6.95 |
| Hungarian | 1 | 7.1 | #DIV/0! | 7.1 |
| Icelandic | 2 | 7.55 | 0.919238816 | 7.55 |
| Indonesian | 2 | 7.9 | 0.424264069 | 7.9 |
| Italian | 11 | 7.227272727 | 1.244259546 | 7.3 |
| Japanese | 17 | 7.347058824 | 1.000073527 | 7.5 |
| Kannada | 1 | 7.1 | #DIV/0! | 7.1 |
| Kazakh | 1 | 6 | #DIV/0! | 6 |
| Korean | 8 | 7.3875 | 0.825378701 | 7.5 |
| Mandarin | 24 | 6.7875 | 1.036848276 | 7.05 |
| Maya | 1 | 7.8 | #DIV/0! | 7.8 |
| Mongolian | 1 | 7.3 | #DIV/0! | 7.3 |
| None | 2 | 7.95 | 0.777817459 | 7.95 |
| Norwegian | 4 | 7.15 | 0.574456265 | 7.3 |
| Panjabi | 1 | 6.6 | #DIV/0! | 6.6 |
| Persian | 4 | 7.575 | 1.203813385 | 7.95 |
| Polish | 3 | 7.966666667 | 0.981495458 | 7.4 |
| Portuguese | 8 | 7.4875 | 0.883883476 | 7.7 |
| Romanian | 2 | 7.2 | 0.989949494 | 7.2 |
| Russian | 11 | 6.363636364 | 1.383671007 | 6.5 |
| Slovenian | 1 | 6.4 | #DIV/0! | 6.4 |

| | | | | |
|---|---|---|---|---|
| Spanish | 40 | 6.9375 | 0.855056603 | 7.15 |
| Swahili | 1 | 7.4 | #DIV/0! | 7.4 |
| Swedish | 5 | 7.44 | 0.756967635 | 7.6 |
| Tamil | 1 | 5.1 | #DIV/0! | 5.1 |
| Telugu | 1 | 8.4 | #DIV/0! | 8.4 |
| Thai | 3 | 6.633333333 | 0.450924975 | 6.6 |
| Urdu | 1 | 7 | #DIV/0! | 7 |
| Vietnamese | 1 | 7.4 | #DIV/0! | 7.4 |
| Zulu | 2 | 7.1 | 0.282842712 | 7.1 |
| **Grand Total** | **4908** | **6.436776691** | **1.127141831** | **7.3** |

There is #DIV/0! Error in Standard Deviation because Standard Deviation of single value i.e 1 cant be calculated.

**TO CALCULATE MEDIAN OF ENGLISH LANGUAGE:**
Syntax used: **=MEDIAN(IMDB_Movies__4[imdb_score])**

**OUTPUT OF MEDIAN:**
**6.5**

**INSIGHTS:**
"ENGLISH" is the most used language for movies. The average of the IMDB for English Language is 6.40 (rounded), with Standard deviation 1.0125 and median IMDB for all the English Language is 6.5 (Medium Value). The most occurring IMDB value (Mode) for English Language is 6.7 (206 count).

D. **Director Analysis:** Influence of directors on movie ratings.

Top 10 director list:

| Director Name | Average of imdb_score | Percentile using percent rank | imdb value based on percentile |
|---|---|---|---|
| John Blanchard | 9.5 | 1 | 9.5 |
| Sadyk Sher-Niyaz | 8.7 | 0.998 | 8.6 |
| Mitchell Altieri | 8.7 | 0.998 | 8.6 |
| Cary Bell | 8.7 | 0.998 | 8.6 |
| Mike Mayhall | 8.6 | 0.997 | 8.5 |
| Charles Chaplin | 8.6 | 0.997 | 8.5 |
| Raja Menon | 8.5 | 0.996 | 8.4856 |
| Ron Fricke | 8.5 | 0.996 | 8.4856 |
| Majid Majidi | 8.5 | 0.996 | 8.4856 |
| Damien Chazelle | 8.5 | 0.996 | 8.4856 |

Syntax for Percent Rank: **=PERCENTRANK.INC($E$4:$E$2398,E4).**

Syntax for Percentile: **=PERCENTILE.INC($E$4:$E$2398,F4).**

E. **Budget Analysis:** Explore the relationship between movie budgets and their financial success.

**TOP-10 MOVIES ACCORDING TO PROFIT MARGIN ARE:**

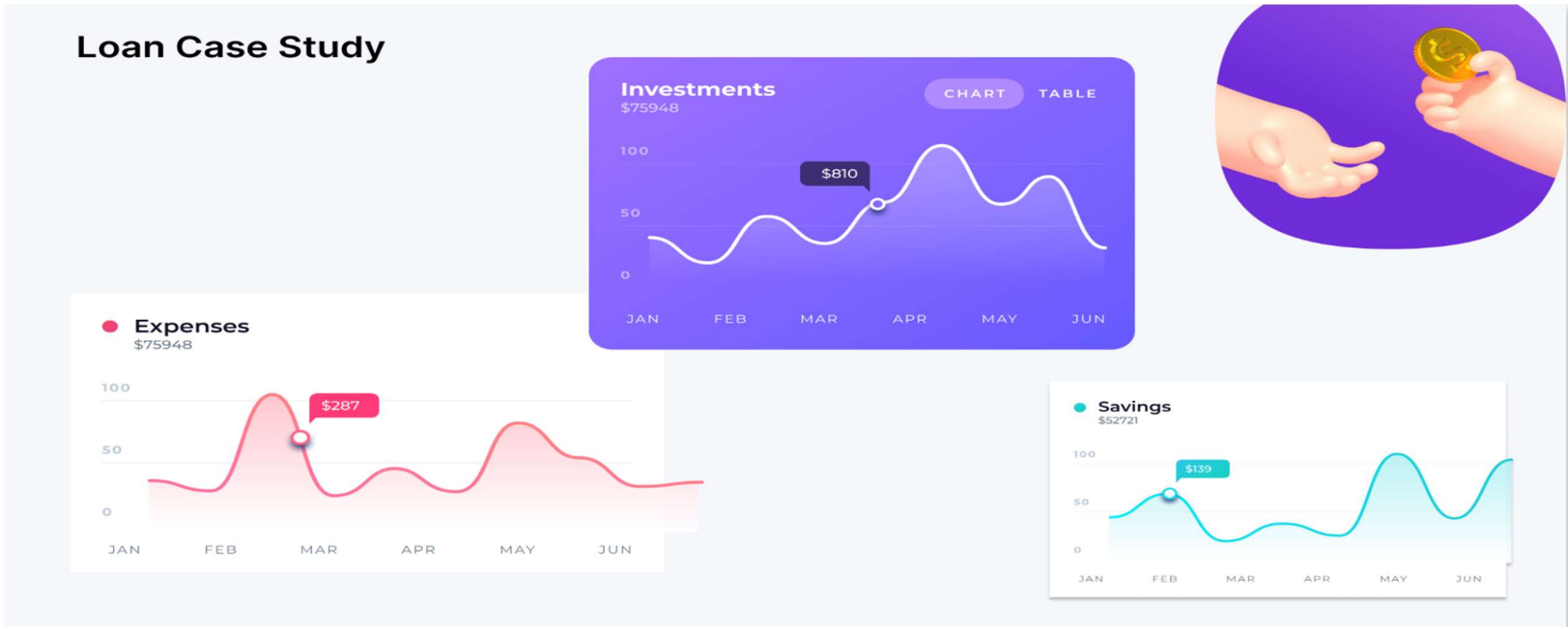
| Movie Title | Profit Margin |
|---|---|
| Avatar | 523505847 |
| Jurassic World | 502177271 |
| Titanic | 458672302 |
| Star Wars: Episode IV - A New Hope | 449935665 |
| E.T. the Extra-Terrestrial | 424449459 |
| The Avengers | 403279547 |
| The Lion King | 377783777 |
| Star Wars: Episode I - The Phantom Menace | 359544677 |
| The Dark Knight | 348316061 |
| The Hunger Games | 329999255 |

The Correlation Coefficient between Gross and Budget is 0.101033478

(Syntax: **=CORREL(IMDB_Movies__5[gross], IMDB_Movies__5[budget])**

**Insights:**
The movie that did the most profit is “Avatar” ( 523505847 ). The linear relation between the Gross Income and the Budget is somewhat straight (upward Direction) because the coefficient is 0.101.

Since Data is already been selected in descending order there is no need to use the MAX function in order to find the high profit making movies.




# MODULE 5: Bank Loan Case Study

A data analyst at a finance company that specializes in lending various types of loans to urban customers. The company faces a challenge: some customers who don't have a sufficient credit history take advantage of this and default on their loans. The task is to use Exploratory Data Analysis (EDA) to analyze patterns in the data and ensure that capable applicants are not rejected.

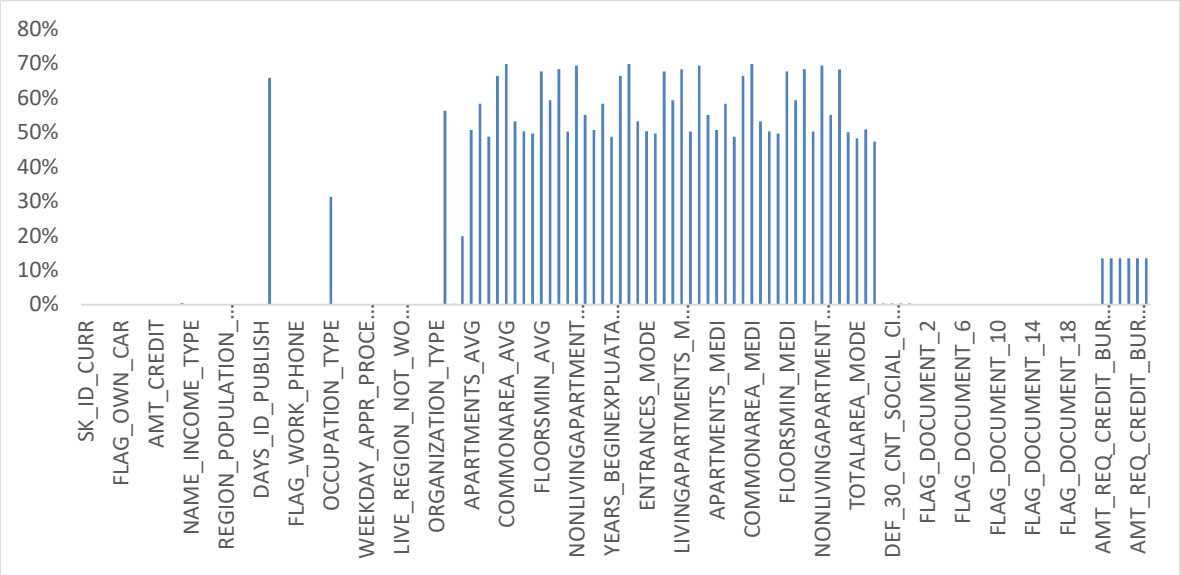
The main aim of this project is to identify patterns that indicate if a customer will have difficulty paying their installments. This information can be used to make decisions such as denying the loan, reducing the amount of loan, or lending at a higher interest rate to risky applicants. The company wants to understand the key factors behind loan default so it can make better decisions about loan approval.

**Data Analytics Tasks:**

A. **Identify Missing Data and Deal with it Appropriately:** As a data analyst, you come across missing data in the loan application dataset. It is essential to handle missing data effectively to ensure the accuracy of the analysis.

Steps involved to do this task:

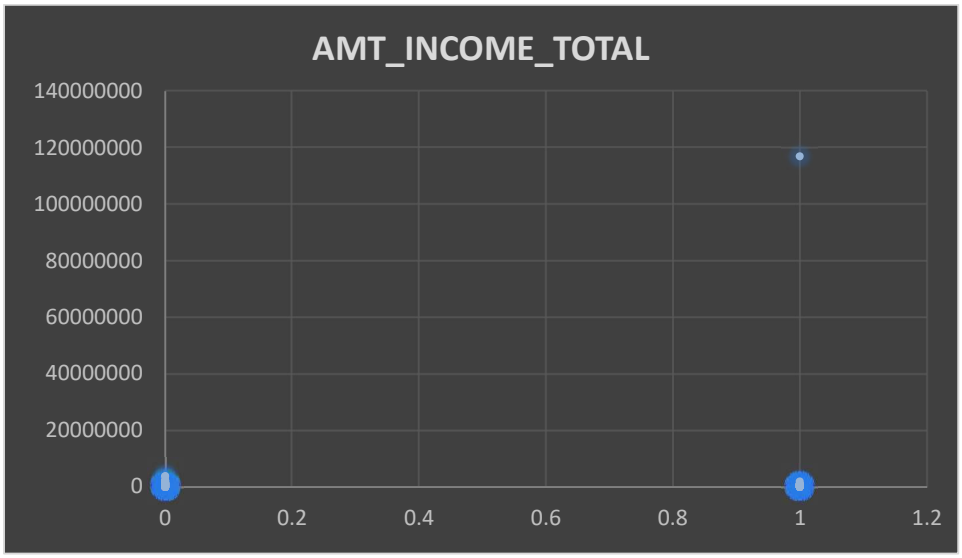
1. First I checked for any duplicate data.
2. Then I inserted 2 columns and did further analysis to find missing data.
3. First I used **"COUNTA"** Function to count each row for a particular column.
4. After that I calculated the missing value in form of percentage (Syntax: **=1-A2/$A$2**).
5. At last I got missing values for each column in form of percentage and marked red for every value which is above 30%.

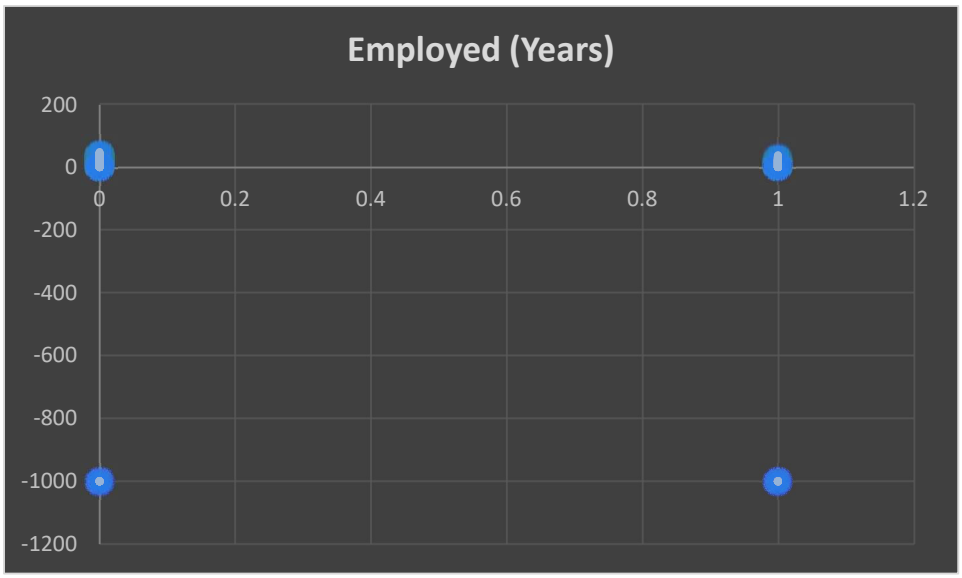



For Imputation of data in the dataset (for data whose missing value is less than 30%), I used MEDIAN AND AVERAGE.

**Inference**: There are many variables with a lot of missing values but can play an important role in analysis. In order to make the analysis more accurate data should be imputed. Now, for imputation there can be 2 ways, first being taking the average of the whole column and impute all the blanks with the average value. Second being the median way, this data imputation way is only viable if the data column has fixed variable values such as (1,2,3,4….), in this case the most occurring value should be imputed in the blanks.

B. **Identify Outliers in the Dataset:** Outliers can significantly impact the analysis and distort the results. You need to identify outliers in the loan application dataset.



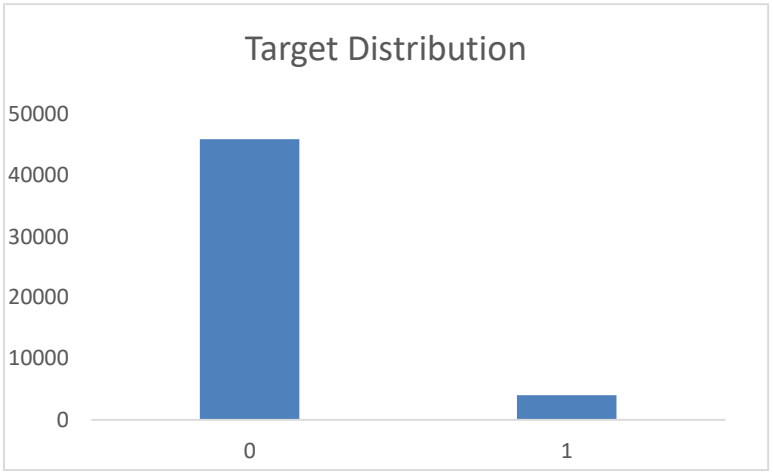




**INSIGHTS:**

For this dataset, the potential outliers can be in the column "**Employed (Years)**" and **"AMT_INCOME_TOTAL".** The value "1001" in Employed (years) is not practically possible and the value "117000000" in AMT_INCOME_TOTAL is also an outlier for this data. Rest outliers based on IQR is considerable for further analysis.

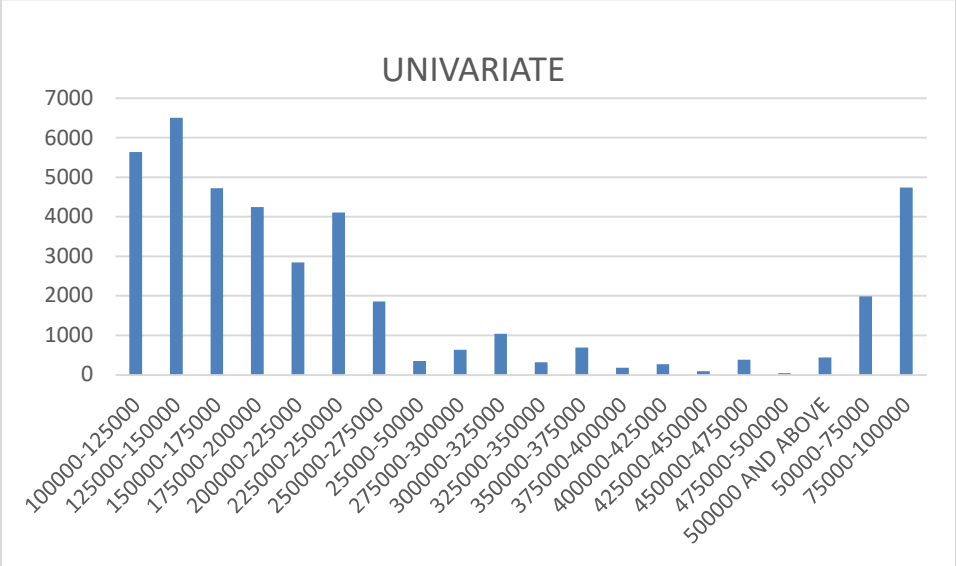
C. **Analyze Data Imbalance:** Data imbalance can affect the accuracy of the analysis, especially for binary classification problems. Understanding the data distribution is crucial for building reliable models.

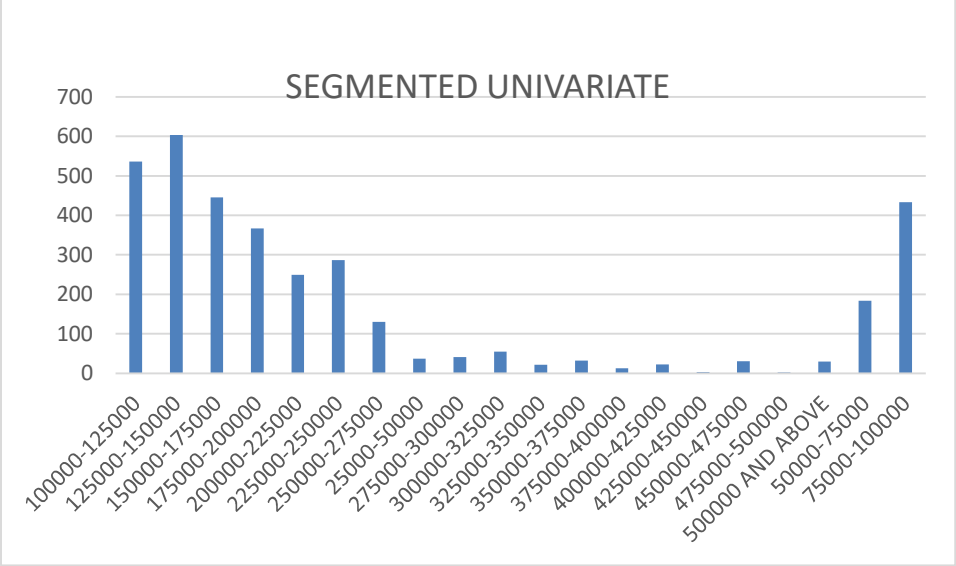
| Target | Count of TARGET | Percentage |
|---|---|---|
| 0 | 45973 | 92% |
| 1 | 4026 | 8% |
| **Grand Total** | **49999** | **100%** |

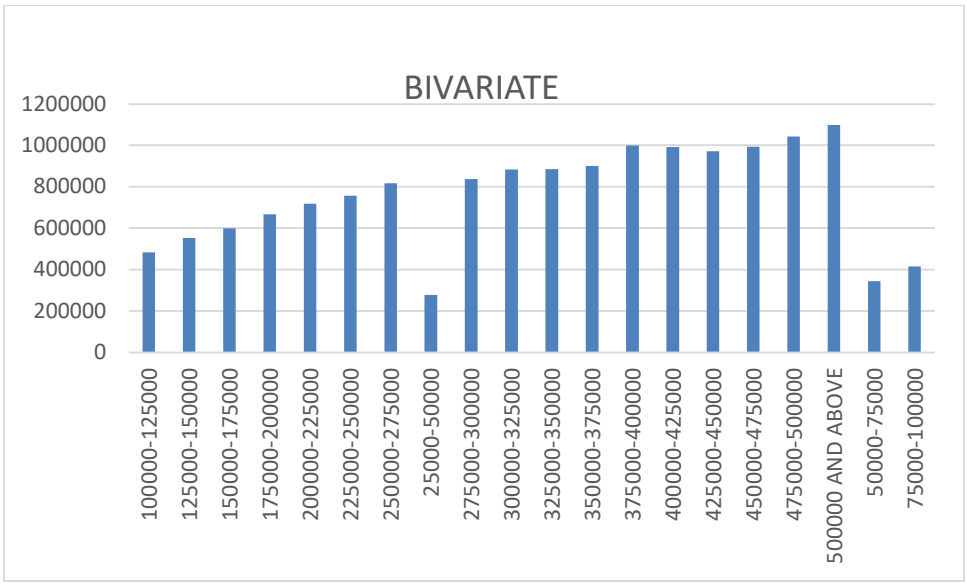


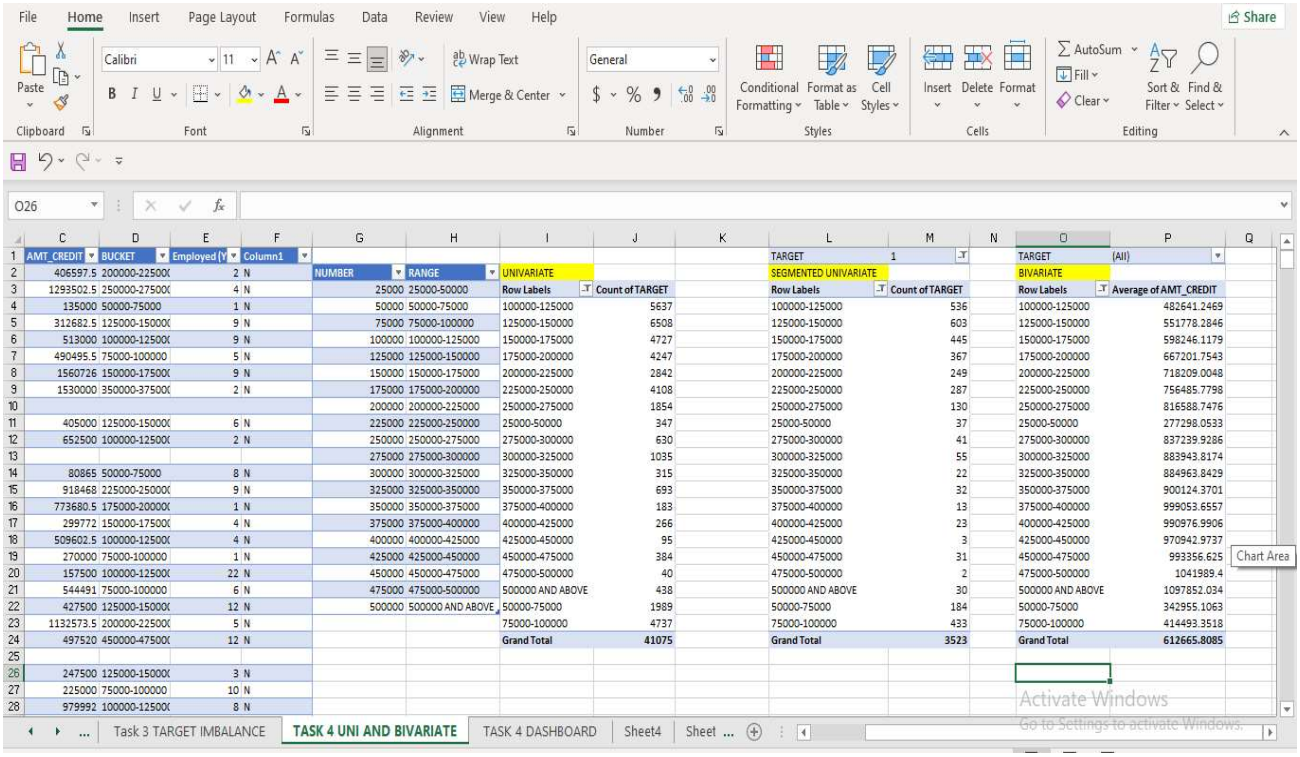

**Inference**: the distribution of target values has huge difference

D. **Perform Univariate, Segmented Univariate, and Bivariate Analysis:** To gain insights into the driving factors of loan default, it is important to conduct various analyses on consumer and loan attributes.






| AMT_CREDIT | BUCKET | Employed (Y | Column1 |
|---|---|---|---|
| 406597.5 | 200000-225000 | 2 | N |
| 1293502.5 | 250000-275000 | 4 | N |
| 135000 | 50000-75000 | 1 | N |
| 312682.5 | 125000-150000 | 9 | N |
| 513000 | 100000-125000 | 9 | N |
| 490495.5 | 75000-100000 | 5 | N |
| 1560726 | 150000-175000 | 9 | N |
| 1530000 | 350000-375000 | 2 | N |
| 405000 | 125000-150000 | 6 | N |
| 652500 | 100000-125000 | 2 | N |
| 80865 | 50000-75000 | 8 | N |
| 918468 | 225000-250000 | 9 | N |
| 773680.5 | 175000-200000 | 1 | N |
| 299772 | 150000-175000 | 4 | N |
| 509602.5 | 100000-125000 | 4 | N |
| 270000 | 75000-100000 | 1 | N |
| 157500 | 100000-125000 | 22 | N |
| 544491 | 75000-100000 | 6 | N |
| 427500 | 125000-150000 | 12 | N |
| 1132573.5 | 200000-225000 | 5 | N |
| 497520 | 450000-475000 | 12 | N |
| 247500 | 125000-150000 | 3 | N |
| 225000 | 75000-100000 | 10 | N |
| 979992 | 100000-125000 | 8 | N |

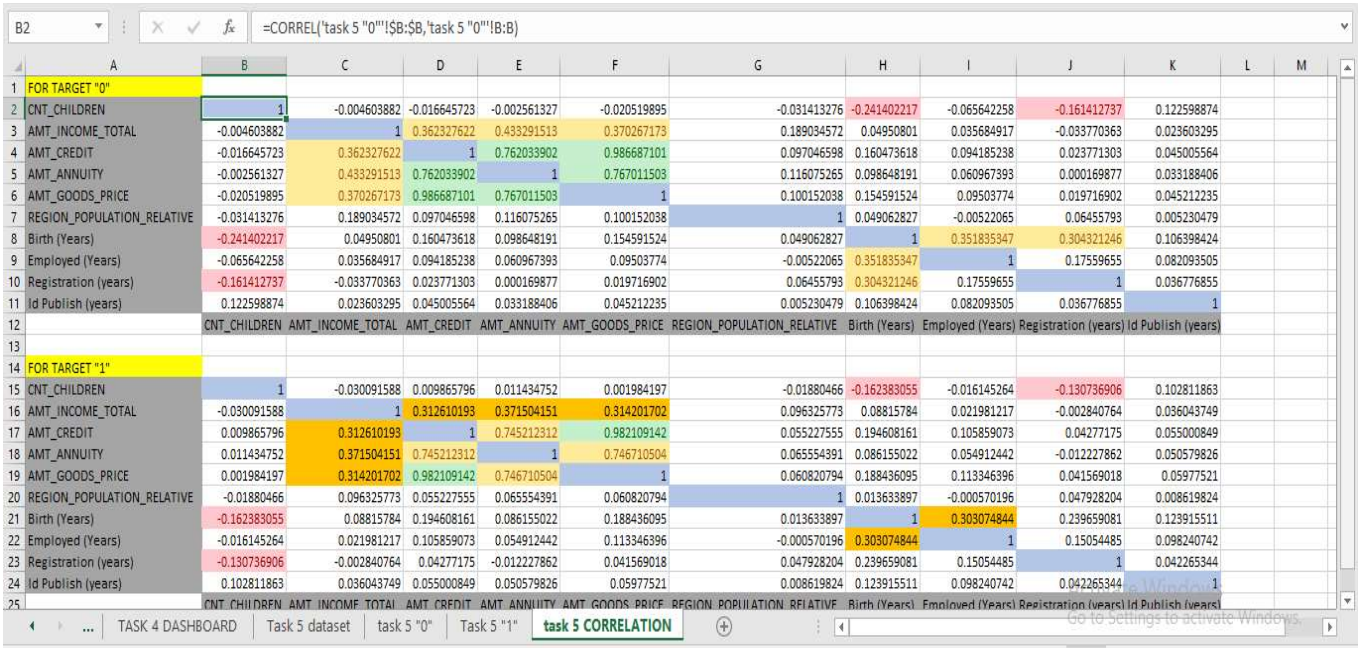
| NUMBER | RANGE |
|---|---|
| 25000 | 25000-50000 |
| 50000 | 50000-75000 |
| 75000 | 75000-100000 |
| 100000 | 100000-125000 |
| 125000 | 125000-150000 |
| 150000 | 150000-175000 |
| 175000 | 175000-200000 |
| 200000 | 200000-225000 |
| 225000 | 225000-250000 |
| 250000 | 250000-275000 |
| 275000 | 275000-300000 |
| 300000 | 300000-325000 |
| 325000 | 325000-350000 |
| 350000 | 350000-375000 |
| 375000 | 375000-400000 |
| 400000 | 400000-425000 |
| 425000 | 425000-450000 |
| 450000 | 450000-475000 |
| 475000 | 475000-500000 |
| 500000 | 500000 AND ABOVE |

| UNIVARIATE | |
|---|---|
| Row Labels | Count of TARGET |
| 100000-125000 | 5637 |
| 125000-150000 | 6508 |
| 150000-175000 | 4727 |
| 175000-200000 | 4247 |
| 200000-225000 | 2842 |
| 225000-250000 | 4108 |
| 250000-275000 | 1854 |
| 25000-50000 | 347 |
| 275000-300000 | 630 |
| 300000-325000 | 1035 |
| 325000-350000 | 315 |
| 350000-375000 | 693 |
| 375000-400000 | 183 |
| 400000-425000 | 266 |
| 425000-450000 | 95 |
| 450000-475000 | 384 |
| 475000-500000 | 40 |
| 500000 AND ABOVE | 438 |
| 50000-75000 | 1989 |
| 75000-100000 | 4737 |
| Grand Total | 41075 |

| TARGET | 1 |
|---|---|
| SEGMENTED UNIVARIATE | |
| Row Labels | Count of TARGET |
| 100000-125000 | 536 |
| 125000-150000 | 603 |
| 150000-175000 | 445 |
| 175000-200000 | 367 |
| 200000-225000 | 249 |
| 225000-250000 | 287 |
| 250000-275000 | 130 |
| 25000-50000 | 37 |
| 275000-300000 | 41 |
| 300000-325000 | 55 |
| 325000-350000 | 22 |
| 350000-375000 | 32 |
| 375000-400000 | 13 |
| 400000-425000 | 23 |
| 425000-450000 | 3 |
| 450000-475000 | 31 |
| 475000-500000 | 2 |
| 500000 AND ABOVE | 30 |
| 50000-75000 | 184 |
| 75000-100000 | 433 |
| Grand Total | 3523 |

| TARGET | (All) |
|---|---|
| BIVARIATE | |
| Row Labels | Average of AMT_CREDIT |
| 100000-125000 | 482641.2469 |
| 125000-150000 | 551778.2846 |
| 150000-175000 | 598246.1179 |
| 175000-200000 | 667201.7543 |
| 200000-225000 | 718209.0048 |
| 225000-250000 | 756485.7798 |
| 250000-275000 | 816588.7476 |
| 25000-50000 | 277298.0533 |
| 275000-300000 | 837239.9286 |
| 300000-325000 | 883943.8174 |
| 325000-350000 | 884963.8429 |
| 350000-375000 | 900124.3701 |
| 375000-400000 | 999053.6557 |
| 400000-425000 | 990976.9906 |
| 425000-450000 | 970942.9737 |
| 450000-475000 | 993356.625 |
| 475000-500000 | 1041989.4 |
| 500000 AND ABOVE | 1097852.034 |
| 50000-75000 | 342955.1063 |
| 75000-100000 | 414493.3518 |
| Grand Total | 612665.8085 |

E. **Identify Top Correlations for Different Scenarios:** Understanding the correlation between variables and the target variable can provide insights into strong indicators of loan default.



| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | FOR TARGET "0" | | | | | | | | | | |
| 2 | CNT_CHILDREN | 1 | -0.004603882 | -0.016645723 | -0.002561327 | -0.020519895 | -0.031413276 | -0.241402217 | -0.065642258 | -0.161412737 | 0.122598874 |
| 3 | AMT_INCOME_TOTAL | -0.004603882 | 1 | 0.362327622 | 0.433291513 | 0.370267173 | 0.189034572 | 0.04950801 | 0.035684917 | -0.033770363 | 0.023603295 |
| 4 | AMT_CREDIT | -0.016645723 | 0.362327622 | 1 | 0.762033902 | 0.986687101 | 0.097046598 | 0.160473618 | 0.094185238 | 0.023771303 | 0.045005564 |
| 5 | AMT_ANNUITY | -0.002561327 | 0.433291513 | 0.762033902 | 1 | 0.767011503 | 0.116075265 | 0.098648191 | 0.060967393 | 0.000169877 | 0.033188406 |
| 6 | AMT_GOODS_PRICE | -0.020519895 | 0.370267173 | 0.986687101 | 0.767011503 | 1 | 0.100152038 | 0.154591524 | 0.09503774 | 0.019716902 | 0.045212235 |
| 7 | REGION_POPULATION_RELATIVE | -0.031413276 | 0.189034572 | 0.097046598 | 0.116075265 | 0.100152038 | 1 | 0.049062827 | -0.00522065 | 0.06455793 | 0.005230479 |
| 8 | Birth (Years) | -0.241402217 | 0.04950801 | 0.160473618 | 0.098648191 | 0.154591524 | 0.049062827 | 1 | 0.351835347 | 0.304321246 | 0.106398424 |
| 9 | Employed (Years) | -0.065642258 | 0.035684917 | 0.094185238 | 0.060967393 | 0.09503774 | -0.00522065 | 0.351835347 | 1 | 0.17559655 | 0.082093505 |
| 10 | Registration (years) | -0.161412737 | -0.033770363 | 0.023771303 | 0.000169877 | 0.019716902 | 0.06455793 | 0.304321246 | 0.17559655 | 1 | 0.036776855 |
| 11 | Id Publish (years) | 0.122598874 | 0.023603295 | 0.045005564 | 0.033188406 | 0.045212235 | 0.005230479 | 0.106398424 | 0.082093505 | 0.036776855 | 1 |
| 12 | | CNT_CHILDREN | AMT_INCOME_TOTAL | AMT_CREDIT | AMT_ANNUITY | AMT_GOODS_PRICE | REGION_POPULATION_RELATIVE | Birth (Years) | Employed (Years) | Registration (years) | Id Publish (years) |
| 13 | | | | | | | | | | | |
| 14 | FOR TARGET "1" | | | | | | | | | | |
| 15 | CNT_CHILDREN | 1 | -0.030091588 | 0.009865796 | 0.011434752 | 0.001984197 | -0.01880466 | -0.162383055 | -0.016145264 | -0.130736906 | 0.102811863 |
| 16 | AMT_INCOME_TOTAL | -0.030091588 | 1 | 0.312610193 | 0.371504151 | 0.314201702 | 0.096325773 | 0.08815784 | 0.021981217 | -0.002840764 | 0.036043749 |
| 17 | AMT_CREDIT | 0.009865796 | 0.312610193 | 1 | 0.745212312 | 0.982109142 | 0.055227555 | 0.194608161 | 0.105859073 | 0.04277175 | 0.055000849 |
| 18 | AMT_ANNUITY | 0.011434752 | 0.371504151 | 0.745212312 | 1 | 0.746710504 | 0.065554391 | 0.086155022 | 0.054912442 | -0.012227862 | 0.050579826 |
| 19 | AMT_GOODS_PRICE | 0.001984197 | 0.314201702 | 0.982109142 | 0.746710504 | 1 | 0.060820794 | 0.188436095 | 0.113346396 | 0.041569018 | 0.05977521 |
| 20 | REGION_POPULATION_RELATIVE | -0.01880466 | 0.096325773 | 0.055227555 | 0.065554391 | 0.060820794 | 1 | 0.013633897 | -0.000570196 | 0.047928204 | 0.008619824 |
| 21 | Birth (Years) | -0.162383055 | 0.08815784 | 0.194608161 | 0.086155022 | 0.188436095 | 0.013633897 | 1 | 0.303074844 | 0.239659081 | 0.123915511 |
| 22 | Employed (Years) | -0.016145264 | 0.021981217 | 0.105859073 | 0.054912442 | 0.113346396 | -0.000570196 | 0.303074844 | 1 | 0.15054485 | 0.098240742 |
| 23 | Registration (years) | -0.130736906 | -0.002840764 | 0.04277175 | -0.012227862 | 0.041569018 | 0.047928204 | 0.239659081 | 0.15054485 | 1 | 0.042265344 |
| 24 | Id Publish (years) | 0.102811863 | 0.036043749 | 0.055000849 | 0.050579826 | 0.05977521 | 0.008619824 | 0.123915511 | 0.098240742 | 0.042265344 | 1 |
| 25 | | CNT_CHILDREN | AMT_INCOME_TOTAL | AMT_CREDIT | AMT_ANNUITY | AMT_GOODS_PRICE | REGION_POPULATION_RELATIVE | Birth (Years) | Employed (Years) | Registration (years) | Id Publish (years) |

**The most correlated variables are:**

FOR TARGET "0":

1. **AMT_ANNUITY AND AMT_CREDIT (VICE-VERSA)**: 0.762033902
2. **AMT_CREDIT AND AMT_GOODS_PRICE (VICE-VERSA)**: 0.986687101
3. **AMT_ANNUITY AND AMT_GOODS_PRICE (VICE-VERSA)**: 0.767011503

FOR TARGET "1":

1. **AMT_CREDIT AND AMT_GOODS_PRICE (VICE-VERSA)**: 0.982109142

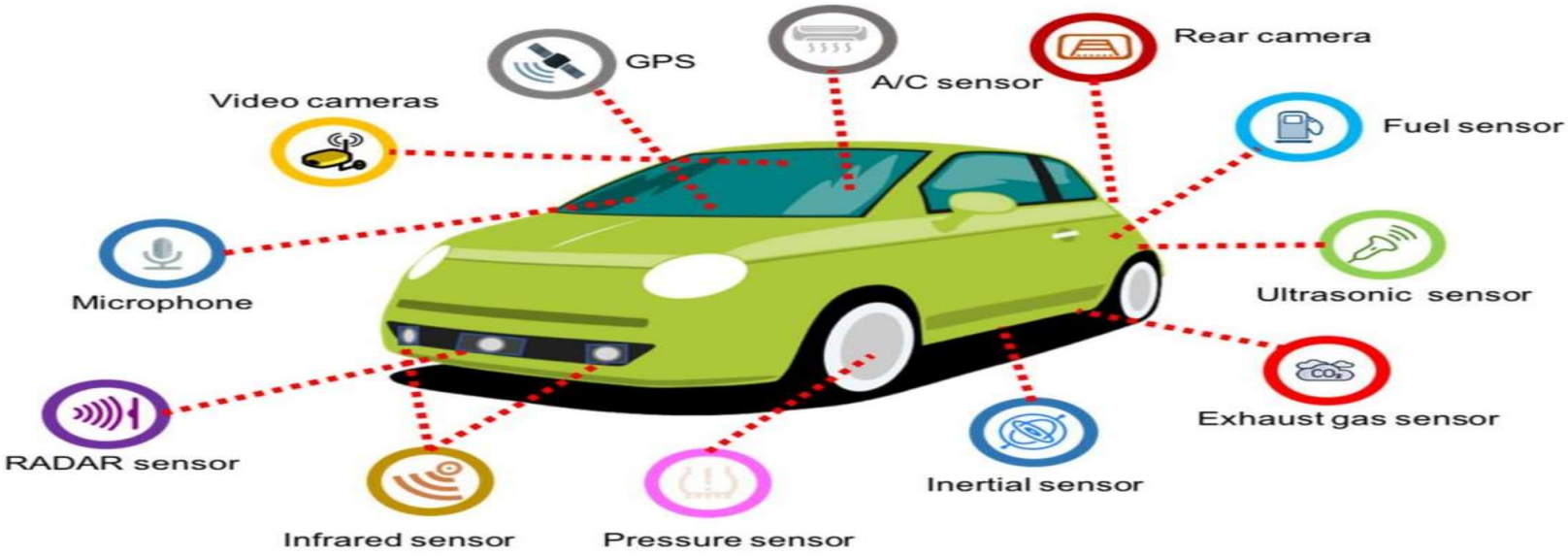
**The least correlated variables are:**

FOR TARGET "0":

1. **BIRTH (YEARS) AND CNT_CHILDREN (VICE-VERSA)**: -0.241402217
2. **REGISTRATION (YEARS) AND CNT_CHILDREN (VICE-VERSA)**: -0.161412737

FOR TARGET "1":

1. **BIRTH (YEARS) AND CNT_CHILDREN (VICE-VERSA)**: -0.162383055
2. **REGISTRATION (YEARS) AND CNT_CHILDREN (VICE-VERSA)**: -0.130736906




# MODULE 6:

## Analyzing the Impact of Car Features on Price and Profitability

**Project Description:**

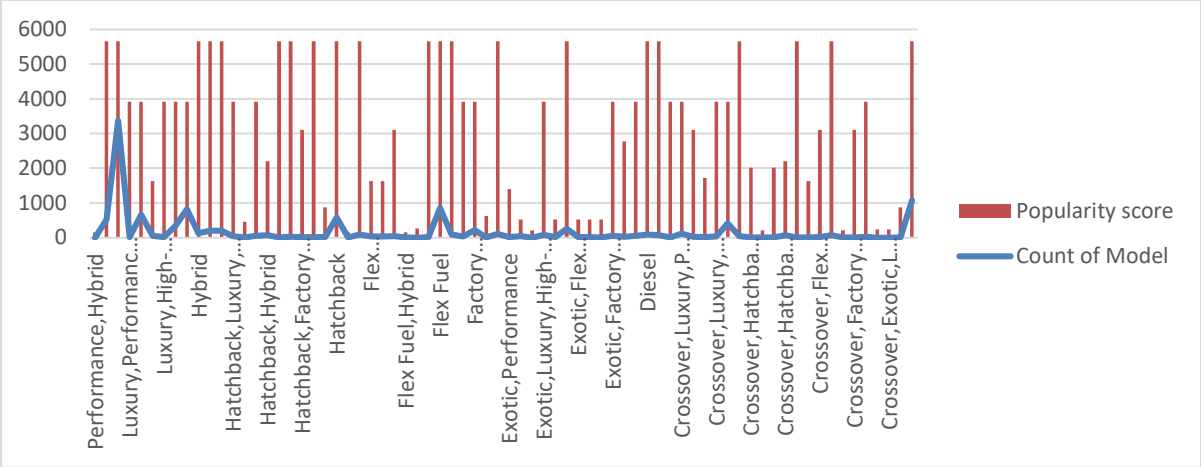
This project aims on the relation of impact of car features and price. For this project various tasks have been assigned along with making dashboards on Microsoft Excel.
The dataset given for this project is car brand data which has information about the car model and the specifications (number of doors, power, miles per gallon information, etc.).

The tasks have been divided into 2 parts.

1. Analysis
2. Building the dashboard

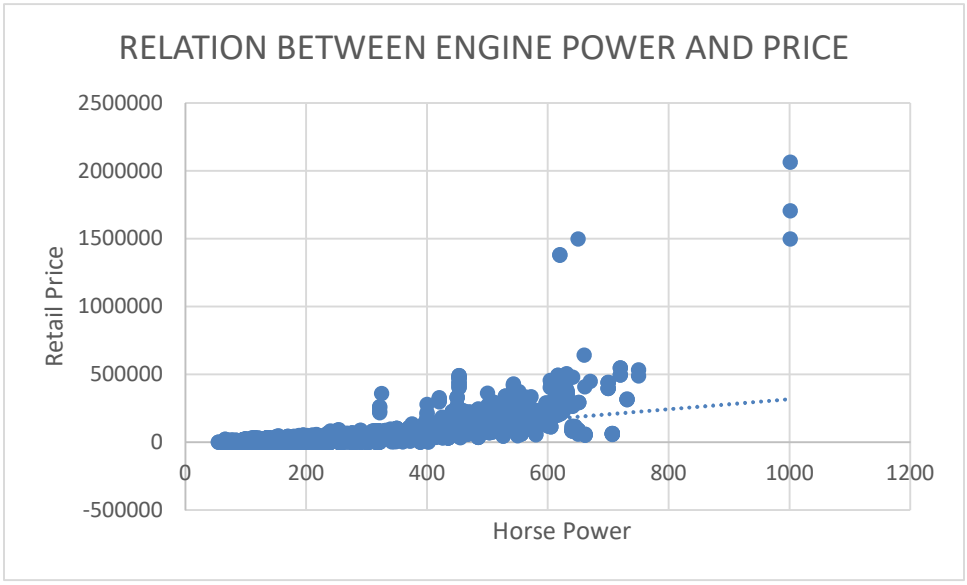
**TASKS: ANALYSIS**

- **Task 1**. **:** Create a combo chart that visualizes the relationship between market category and popularity




**Insight Required:** What is the relationship between a car's engine power and its price?

- **Task 2:** Create a scatter chart that plots engine power on the x-axis and price on the y-axis. Add a trendline to the chart to visualize the relationship between these variables.

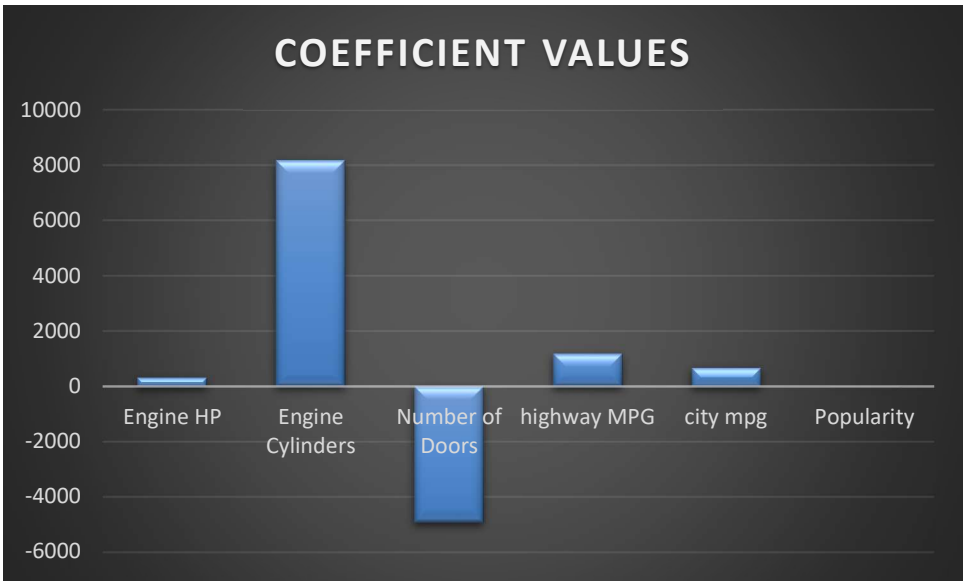



The correlation coefficient between both the variables is: 0.659174

**Insight**: with increasing Horse power the Price also increases.

**Insight Required:** Which car features are most important in determining a car's price?

- **Task 3:** Use regression analysis to identify the variables that have the strongest relationship with a car's price. Then create a bar chart that shows the coefficient values for each variable to visualize their relative importance.




The visualization clearly states that engine cylinders is important to determine the car's price.

The visualization was done by using regression modeling in excel.

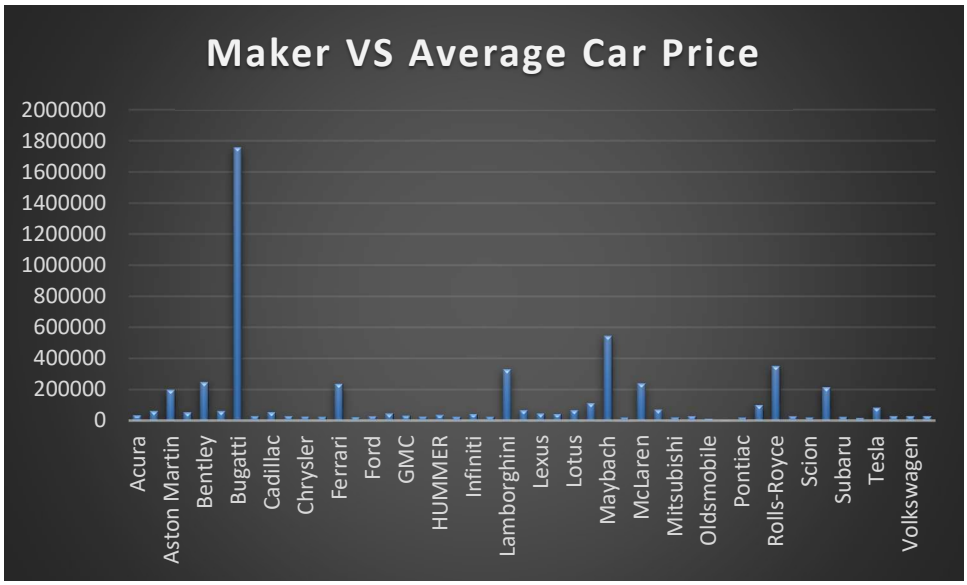
**Insight Required:** How does the average price of a car vary across different manufacturers?

- **Task 4.A:** Create a pivot table that shows the average price of cars for each manufacturer.

| Row Labels | Average of MSRP |
|---|---|
| Acura | 35087.4878 |
| Alfa Romeo | 61600 |
| Aston Martin | 198123.4615 |
| Audi | 54574.1215 |
| Bentley | 247169.3243 |
| BMW | 62162.55864 |
| Bugatti | 1757223.667 |
| Buick | 29034.18947 |
| Cadillac | 56368.26515 |
| Chevrolet | 29074.72576 |
| Chrysler | 26722.96257 |
| Dodge | 24857.04537 |
| Ferrari | 238218.8406 |
| FIAT | 22670.24194 |
| Ford | 28511.30788 |
| Genesis | 46616.66667 |
| GMC | 32444.08506 |
| Honda | 26655.14781 |
| HUMMER | 36464.41176 |
| Hyundai | 24926.26255 |
| Infiniti | 42640.27134 |

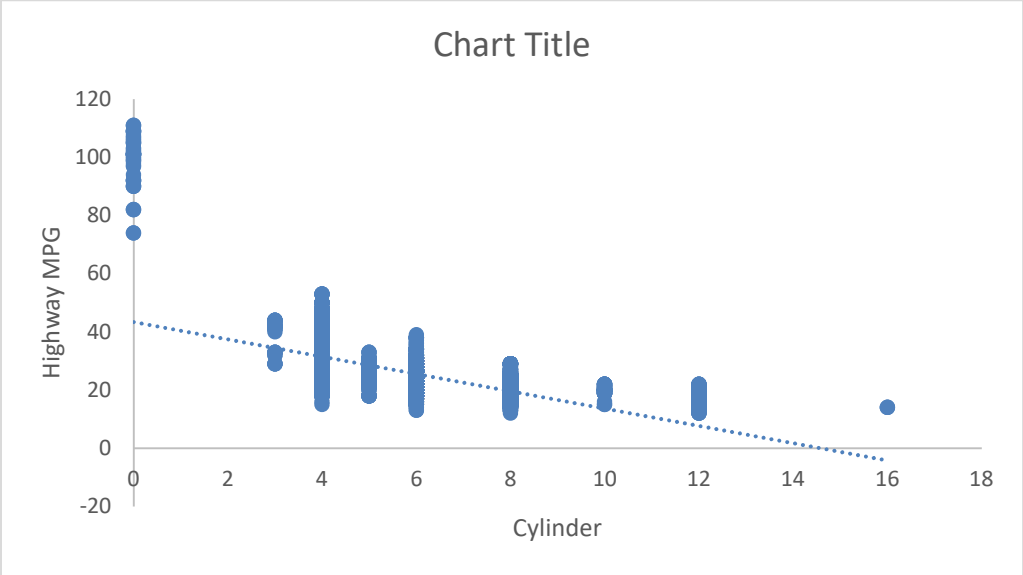
| | |
|---|---|
| Kia | 25513.75546 |
| Lamborghini | 331567.3077 |
| Land Rover | 68067.08633 |
| Lexus | 47549.06931 |
| Lincoln | 43860.825 |
| Lotus | 68377.14286 |
| Maserati | 113684.4909 |
| Maybach | 546221.875 |
| Mazda | 20416.62379 |
| McLaren | 239805 |
| Mercedes-Benz | 72069.52786 |
| Mitsubishi | 21340.5625 |
| Nissan | 28921.15245 |
| Oldsmobile | 12843.79545 |
| Plymouth | 3296.873239 |
| Pontiac | 19800.0442 |
| Porsche | 101622.3971 |
| Rolls-Royce | 351130.6452 |
| Saab | 27879.80734 |
| Scion | 19932.5 |
| Spyker | 214990 |
| Subaru | 24240.67364 |
| Suzuki | 18026.4152 |
| Tesla | 85255.55556 |
| Toyota | 28846.5605 |
| Volkswagen | 28978.52289 |
| Volvo | 29724.68421 |
| **Grand Total** | **41925.92714** |

- **Task 4.B:** Create a bar chart or a horizontal stacked bar chart that visualizes the relationship between manufacturer and average price.




**Insight Required:** What is the relationship between fuel efficiency and the number of cylinders in a car's engine?

**Task 5:** Create a scatter plot with the number of cylinders on the x-axis and highway MPG on the y-axis. Then create a trendline on the scatter plot to visually estimate the slope of the relationship and assess its significance.

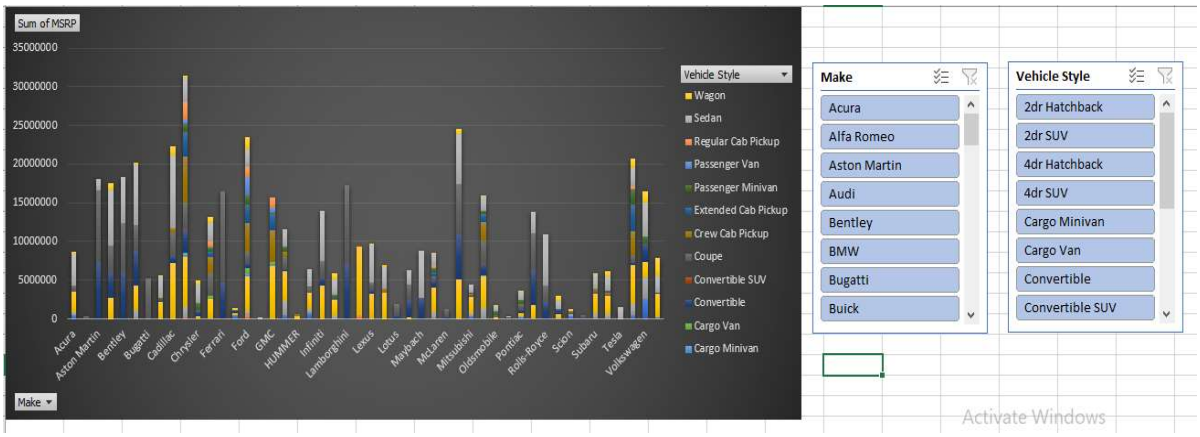



CORRELATION : -0.65746

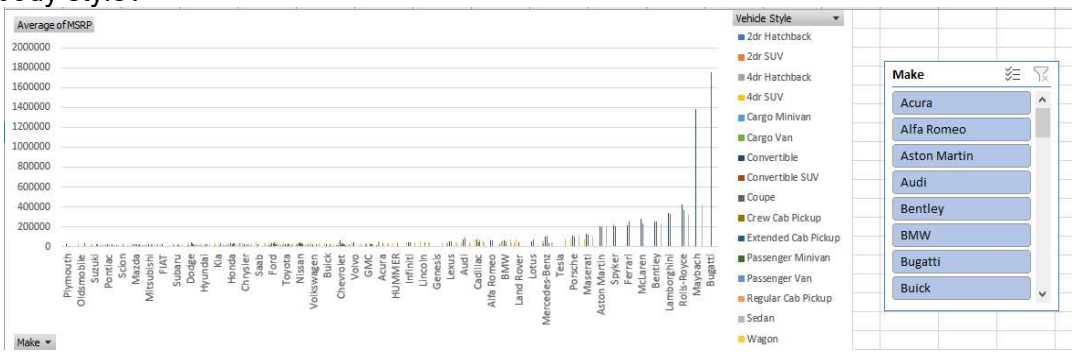
With increasing number of cylinder the Highway MPG decreases.

**Building the Dashboard:**

**Task 1:** How does the distribution of car prices vary by brand and body style?

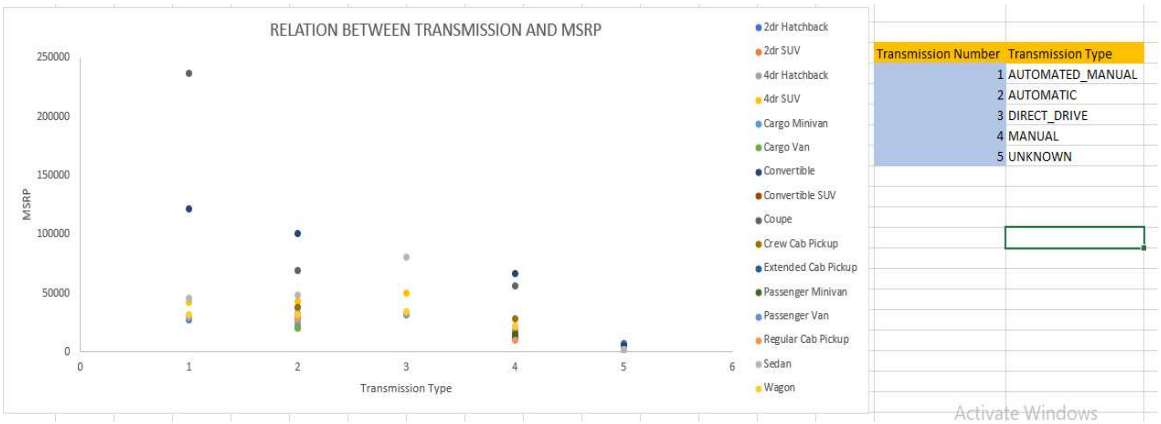



**Task 2:** Which car brands have the highest and lowest average MSRPs, and how does this vary by body style?

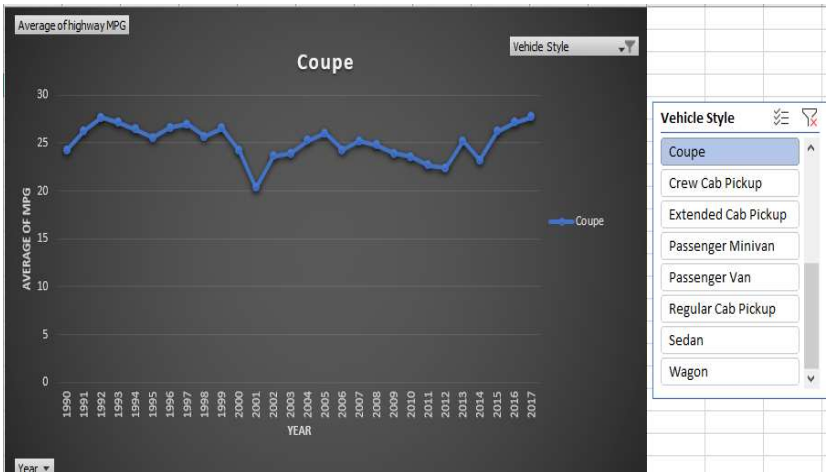



**Insight:**
car brands that have the highest and lowest average MSRPs is Bugatti (1757223.667) and Plymouth (3296.87) Respectively.

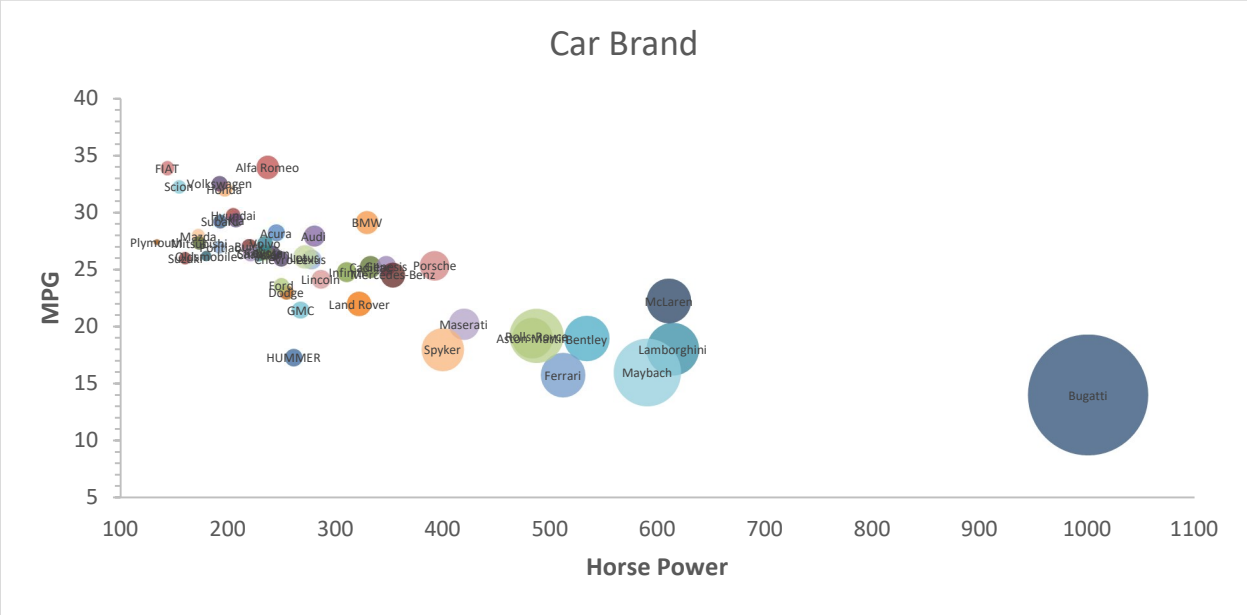
**Task 3:** How do the different feature such as transmission type affect the MSRP, and how does this vary by body style?




**Task 4:** How does the fuel efficiency of cars vary across different body styles and model years?




**Task 5:** How does the car's horsepower, MPG, and price vary across different Brands?




**Insight**: Bugatti is the highest in MSRP with the maximum horse power and the lowest MPG.

**Tech Stack Used:** Microsoft Excel 2021 was used for this project.

Power BI

# Power BI Dashboard – Adventure Works Data

This project contains all the contents of Power BI that a Data Analyst should know as a Developer. Contents that are covered :

1. Power Query Editor (ETL).
2. Building Relationship (Snowflake Schema) – Data Modelling.
3. DAX – Data Analysis Expressions.
4. Building Report (Dashboard).

The data provided was taken from Microsoft itself (Adventure Works Data) In form of csv., (2015,2016,2017).

| | | | |
|---|---|---|---|
| Sales | 12-Oct-23 9:08 PM | File folder | |
| AdventureWorksCalendar-210509-235702 | 06-Oct-23 9:22 AM | Microsoft Excel C... | 10 KB |
| AdventureWorksCustomers-210509-2357... | 06-Oct-23 9:27 AM | Microsoft Excel C... | 1,918 KB |
| AdventureWorksProductCategories-2105... | 06-Oct-23 9:22 AM | Microsoft Excel C... | 1 KB |
| AdventureWorksProducts-210509-235702 | 06-Oct-23 9:23 AM | Microsoft Excel C... | 57 KB |
| AdventureWorksProductSubcategories-2... | 06-Oct-23 9:22 AM | Microsoft Excel C... | 1 KB |
| AdventureWorksReturns-210509-235702 | 06-Oct-23 9:23 AM | Microsoft Excel C... | 34 KB |
| AdventureWorksTerritories-210509-235702 | 06-Oct-23 9:23 AM | Microsoft Excel C... | 1 KB |
| AW-Report-final | 01-Jan-24 11:47 PM | Microsoft.Microso... | 2,309 KB |
| ColumnfromExample-210509-235702 | 06-Oct-23 9:23 AM | Microsoft Excel W... | 94 KB |
| Fill-210509-235702 | 06-Oct-23 9:26 AM | Microsoft Excel W... | 36 KB |
| IncrementalRefresh-220430-164031 | 06-Oct-23 9:28 AM | Microsoft PowerP... | 211 KB |
| UnpivotData-210509-235702 | 06-Oct-23 9:27 AM | Microsoft Excel W... | 248 KB |

1. ETL: since Power BI has its own ETL (Export, Transform, Load) tool known as Power Query, a reason why I use Power Bi as Data Visualization software over Tablue.
   CONTENTS:
   A. D-Customers: Customer Key, Birth Date, Marital Status, Gender, Email Address, Annual Income, Total Children, Education Level, Occupation, Home Owner, Full Name, Username.
   B. D-Product: Product Key, Product Subcategory Key, SKU Type, SKU, Product Name, Model Name, Product Description, Product Color, Product Size, Product Style, Product Cost, Product Price, Revenue.
   C. D-Categories: Category Key, Category Name.
   D. D-Territories: Sales Category Key, Region, Country, Continent.
   E. D-Calendar: Date, Day Name, Start of Month, Month Name, Year.
   F. D-Subcategory: Product Subcategory Key, Subcategory Name, Product Category Key.
   G. F-Returns: Return Date, Territory Key, Product Key, Return Quantity.
   H. F-Sales: Order Date, Stock Date, Order Number, Product Key, Customer Key, Territory Key, Order Line Item, Order Quantity.

   Here D is Dimension and F is Fact.

2. Measures:

In this Project 25 Measures are created using different DAX functions, Measures are preferable over creating a column because it does not bloat the table (don't add up any size in the file).

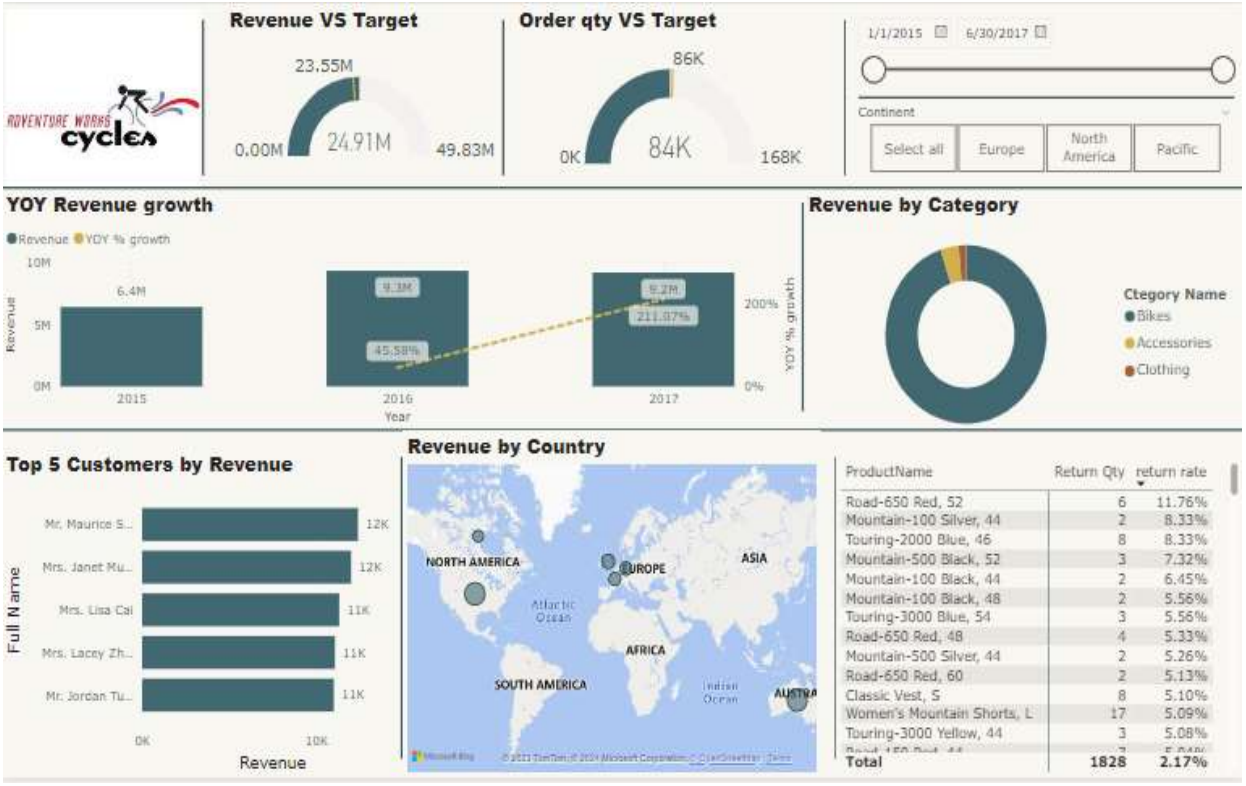
3. DAX:

There are many DAX functions used like:

A. Logical DAX
B. Conditional Column
C. SWITCH
D. Year and Today
E. Text Function (Left and Search)
F. RELATED AND RELATABLE
G. COUNT
H. CALCULATE
I. FILTER
J. SAMEPERIODLASTYEAR
K. ITERATOR FUNCTION (SUMX, MAXX)
L. CROSSFILER
M. ALL

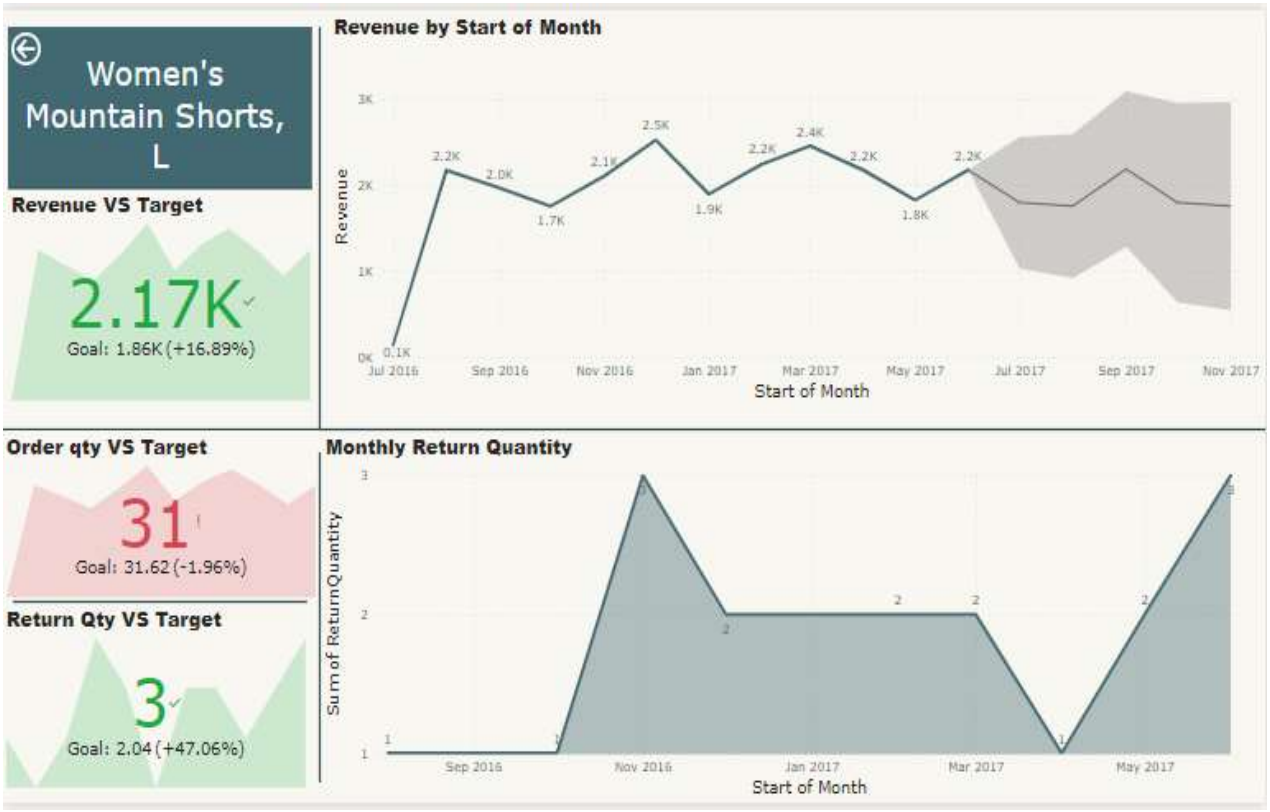
N. MONTH, QTR, YEAR TO DATE CALCULATION
O. PREVIOUS DAY, MONTH, QTR, YEAR
P. PERIODIC CALCULATION

4. BUILDING REPORT
   Various Visualizations are used eg. (matrix, slicer, stacked bar chart, line chart, kpi, map, Donut Chart, Gauge, etc)



PRODUCT DETAIL PANE:

Used drill through for this visualization pane so that it can show the details of the product when selected by the end user.




**CONCLUSION:**

In conclusion, I would like to tell that after doing a thorough analysis we were able to derive the insights from the data and was able to plot various graphs using that data. The data which once looked useless gave some very useful insights.

**APPENDIX**

**GITHUB LINK FOR DATASET:**

**https://github.com/Mayank449851/Dataset**