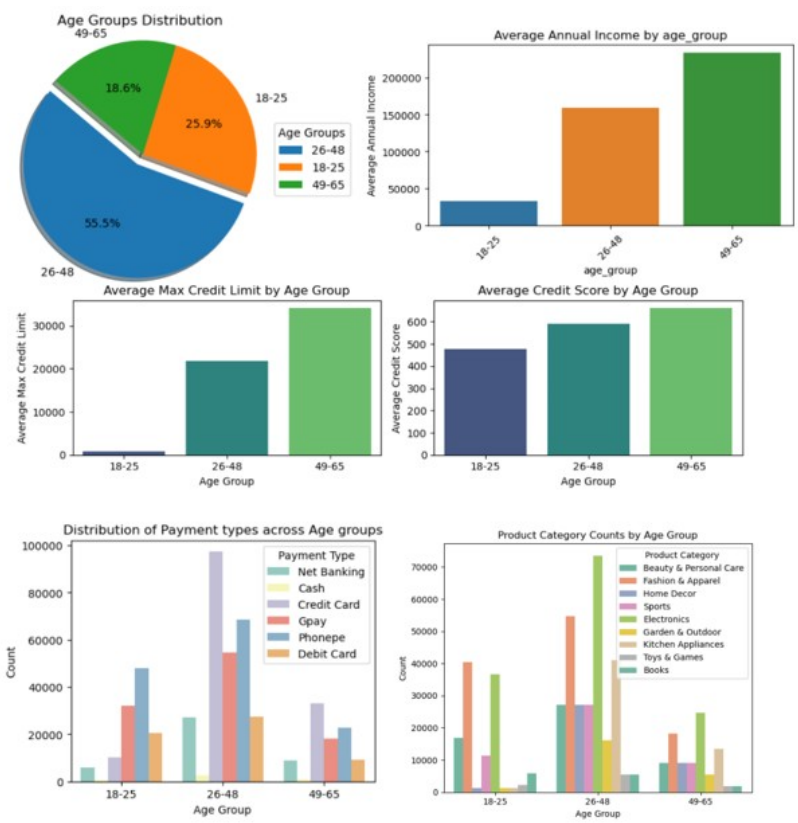
# AtliQo Bank Credit Card Launch

## PHASE 2 - Targeting Untapped Market

- People with age group of 18 -25 accounts to ~25% of customer base in the data
- Avg annual income of this age group is less than 50k
- They don't have much credit history which is getting reflected in their credit score and max credit limit
- Usage of credit cards as payment type is relatively low compared to other groups
- Avg transaction amount made with credit cards is also low compared to other groups
- Top 3 most used shopping products categories : Electronics, Fashion & Apparel, Beauty & Personal care

```python
from IPython.display import Image, display

display(Image(filename="E:/Study/Data Analysis/DS - BC 2025/2 - Math & Statistics/5 - PROJECT/PHASE 2/analysis.png"))
```




## (1) Pre-Campaign

We want to do a trial run for our new credit card.

For this we need to figure out **How many customers do we need for our A/B testing?**

And for that, we will form a **control and test group**.

For both of these groups we can figure out number of customers we need based on the statistical power and effect size that we agree upon after discussing with business.

```
import statsmodels.stats.api as sms
import statsmodels.api as sm
import pandas as pd
import numpy as np
from scipy import stats as st
from matplotlib import pyplot as plt
import seaborn as sns

alpha = 0.05
power = 0.8
effect_size=0.2

sms.tt_ind_solve_power(
    effect_size=0.2,
    alpha=alpha,
    power=power,
    ratio=1,
    alternative='two-sided'
)
```

```
393.4056989990351
```

For **effect size : 2, we need 393 customers**.

We have to keep in mind budgeting restrictions while running this campaign, hence, let us run this for different effect sizes and then we'll discuss with business to find out which sample size would be optimal

```
# Calculating the required sample size for different effect sizes

effect_sizes = [0.1, 0.2, 0.3, 0.4, 0.5, 1.0]  #  std_dev greater than
control group

for effect_size in effect_sizes:
    sample_size = sms.tt_ind_solve_power(effect_size=effect_size,
alpha=alpha, power=power, ratio=1, alternative='two-sided')
    print(f"Effect Size: {effect_size}, Required Samples:
{int(sample_size)} customers")
```

```
Effect Size: 0.1, Required Samples: 1570 customers
Effect Size: 0.2, Required Samples: 393 customers
Effect Size: 0.3, Required Samples: 175 customers
Effect Size: 0.4, Required Samples: 99 customers
```

```
Effect Size: 0.5, Required Samples: 63 customers
Effect Size: 1.0, Required Samples: 16 customers
```

Based on business requirements, the test should be capable of detecting a minimum 0.4 standard deviation difference between the control and test groups.

For the effect size 0.4, we need 100 customers and when we discussed with business, 100 customers is ok in terms of their budgeting constraints for this trail run.

### Forming control and test groups

- We have identified approximately 246 customers within the age group of 18 to 25. From this pool, we will select 100 customers for the initial campaign launch.
- The campaign is launched for 100 customers, as determined by the effective size calculation and by considering budgeting costs, and will run campaign for a duration of 2 months.
- Got a conversion rate of ~40% ( implies 40 out of 100 customers in test group started using credit card)
- To maintain a similar sample size, a control group consisting of 40 customers will be created. Importantly, this control group will be completely exclusive of initial 100 customers used as test group.
- So now we have 40 customers in each of control and test groups.

At the end of the 2-month campaign period (from 09-10-23 to 11-10-23), we obtained daily data showing the average transaction amounts made by the entire group of 40 customers in both the control and test groups using existing and newly launched credit cards respectively

The key performance indicator (KPI) for this A/B Test aims to enhance average transaction amounts facilitated by the new card

## (2) Post-Campaign

### Two Sample Z Test for Our Hypothesis Testing

```
# Loading campaign results data
df = pd.read_csv('E:/Study/Data Analysis/DS - BC 2025/2 - Math & Statistics/5 - PROJECT/PHASE 2/data/avg_transactions_after_campaign.csv')
df.head(5)
```

| | campaign_date | control_group_avg_tran | test_group_avg_tran |
|---|---|---|---|
| 0 | 2023-09-10 | 259.83 | 277.32 |
| 1 | 2023-09-11 | 191.27 | 248.68 |
| 2 | 2023-09-12 | 212.41 | 286.61 |
| 3 | 2023-09-13 | 214.92 | 214.85 |
| 4 | 2023-09-14 | 158.55 | 344.08 |

```
df.shape
```

```
(62, 3)
```

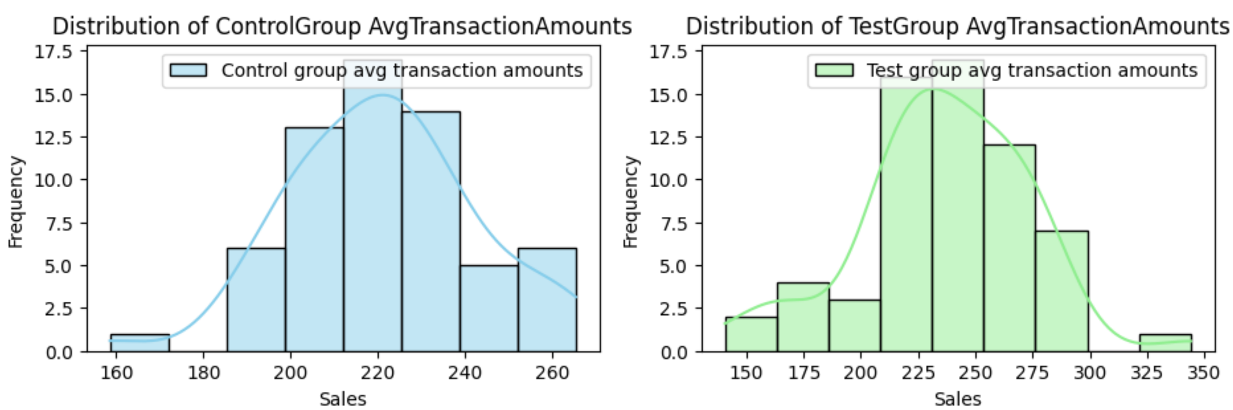
```
# Let's look at distributions of avg transactions amounts in both groups

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(11, 3))  # Creating a 1x2 grid of subplots

# Distribution of Campaign A Sales
sns.histplot(df['control_group_avg_tran'], kde=True, color='skyblue', label='Control group avg transaction amounts', ax=ax1)
ax1.set_xlabel('Sales')
ax1.set_ylabel('Frequency')
ax1.set_title('Distribution of ControlGroup AvgTransactionAmounts')
ax1.legend()

# Distribution of Campaign B Sales
sns.histplot(df['test_group_avg_tran'], kde=True, color='lightgreen', label='Test group avg transaction amounts', ax=ax2)
ax2.set_xlabel('Sales')
ax2.set_ylabel('Frequency')
ax2.set_title('Distribution of TestGroup AvgTransactionAmounts')
ax2.legend()

plt.show()
```




Performing Hypothesis Testing Using Two Sample Z-test

```
control_mean = df["control_group_avg_tran"].mean().round(3)
control_std = df["control_group_avg_tran"].std().round(3)
control_mean, control_std
```

```
(221.175, 21.359)
```

```
test_mean = df["test_group_avg_tran"].mean().round(3)
test_std = df["test_group_avg_tran"].std().round(3)
test_mean, test_std
```

```
(235.984, 36.658)
```

```
sample_size = df.shape[0]
sample_size
```

```
62
```

Test Using Rejection Region (i.e. Critical Z Value)

```
a = (control_std**2 / sample_size)
b = (test_std**2 / sample_size)

Z_score = ((test_mean - control_mean) / np.sqrt(a + b)).round(3)
Z_score
```

```
2.748
```

```
# For a significance level of 5% (0.05) in a right-tailed test, the
critical Z-value is approximately 1.645

critical_z_value = (st.norm.ppf(1 - alpha)).round(3)  # Right-tailed
test at 5% significance level
critical_z_value
```

```
1.645
```

```
Z_score > critical_z_value
```

```
True
```

**Z-score > critical Z value, we can reject the null hypothesis**.

Test Using p-Value

```
# Calculating the p-value corresponding to z score for a right-tailed
test

p_value = 1 - st.norm.cdf(Z_score)
p_value
```

```
0.0029980004542962124
```

```
p_value < alpha
```

```
True
```

p-Value < significance level (i.e. alpha), we can reject the null hypothesis.