Mastering Customer Segmentation Playbook: Boost Marketing Efforts

Why Segmentation Matters

In today's digital marketplace, personalized marketing is essential for business growth. By segmenting your audience based on data, you can create highly targeted campaigns that drive conversions. This playbook introduces three powerful segmentation models and provides step-by-step guidance on how to implement them effectively.

---

1. Storefront Activity Analysis: Turn Browsers into Buyers

What It Is

This model analyzes user behavior on your website, including clicks, product page visits, cart additions, and abandonment rates. It helps identify first-time visitors and tailor marketing messages accordingly.

Step-by-Step Implementation

1. **Collect Data:** Use tools like Google Analytics or CRM software to track user behavior (page views, session duration, abandoned carts, etc.).
2. **Analyze Data:**
   - Import your Google Analytics data into Excel.
   - Use pivot tables to analyze user behavior trends (e.g., high bounce rates, frequent cart abandonment).
3. **Segment Users:** Group visitors into categories:
   - **Browsers:** High page views but no purchases.
   - **Cart Abandoners:** Added items but didn’t checkout.
   - **Repeat Visitors:** Visited multiple times without purchasing.
4. **Develop Targeted Campaigns:**
   - **For Browsers:** Send reminder emails showcasing best-selling products.
   - **For Cart Abandoners:** Offer discounts or free shipping incentives.
   - **For Repeat Visitors:** Provide tailored recommendations based on previous browsing history.

5. **Monitor & Optimize:** Track the impact of your campaigns and adjust based on engagement rates.

---

2. Lift Model: Encourage Repeat Purchases

What It Is

This predictive model identifies complementary products that are frequently purchased together, helping businesses recommend the right products at the right time. By using **association rule mining**, you can generate rules that predict which product combinations have the highest probability of being purchased together.

Step-by-Step Implementation

1. Data Collection & Preparation

Using Excel:

- Import sales data with columns for transaction IDs, products purchased, and dates.
- Clean the data by removing duplicates and irrelevant transactions.

Using Python:

- Use pandas to import and clean the dataset.

```
import pandas as pd
data = pd.read_csv('sales_data.csv')
```

---

2. Apply Association Rule Mining

In Python:

- Use the mlxtend library to apply the **Apriori algorithm**.

```
from mlxtend.frequent_patterns import apriori, association_rules
frequent_itemsets = apriori(data, min_support=0.1, use_colnames=True)
rules = association_rules(frequent_itemsets, metric='lift', min_threshold=1)
print(rules)
```

- Analyze the resulting rules to identify high-confidence associations.

Example Rules:

- **Rule:** ***If a customer buys a smartphone, they are 75%*** likely to buy a screen protector (Lift = **1.8**).
- **Rule:** Customers who purchase coffee are 50% more likely to buy a croissant (Lift = 1.5).

Sample Output of the Apriori Algorithm

Antecedent	Consequent	Support	Confidence	Lift	Leverage	Conviction
{Smartphone}	***{Screen Protector}***	***0.10***	***0.75***	***1.8***	0.02	2.5
{Running Shoes}	{Athletic Socks}	0.08	0.70	1.5	0.015	2.2
{Laptop}	{Wireless Mouse}	0.12	0.85	2.0	0.03	3.0
{Bread, Peanut Butter}	{Jam}	0.05	0.60	1.3	0.01	1.8

Explanation of the Columns:

- **Antecedent:** The item or set of items that customers have already purchased (e.g., {Smartphone}).
- **Consequent:** The item that is likely to be purchased next (e.g., {Screen Protector}).
- **Support:** The proportion of transactions that contain both the antecedent and consequent (e.g., 10% of all transactions include both a smartphone and a screen protector).
- **Confidence:** The probability that customers who bought the antecedent will also buy the consequent ***(e.g., 75% of customers who buy a smartphone also buy a screen protector).***
- **Lift:** How much more likely the consequent is purchased when the antecedent is present compared to random chance. A Lift value > 1 indicates a positive correlation (e.g., a Lift **of 1.8 means** customers are 1.8 times more likely to buy a screen protector with a smartphone).
- **Leverage and Conviction:** Advanced metrics to assess the strength and reliability of the rule

---

3. Develop Cross-Sell Strategies

- **Post-Purchase Emails:** Suggest complementary products based on the rules generated.
    - *Example: "Customers like you also loved these screen protectors." (For customers that purchased a smartphone)*
- **Bundled Offers:** Create product bundles with special discounts.
    - *Example: Offer a 20% discount when a customer buys a laptop with a wireless mouse.*

---

4. Test & Optimize

- **A/B Test Recommendations:** Experiment with different recommendation formats (emails, pop-ups, homepage sections).
- **Track KPIs:** Monitor repeat purchase rates, cross-sell conversion rates, and customer lifetime value.
- **Refine Rules:** Continuously update your association rules based on new sales data to keep recommendations fresh and relevant.

---

3. Collaborative Filtering Model: Retain Loyal Customers

What It Is

A machine learning-based approach that analyzes past user interactions to recommend products based on shared preferences. This model predicts customer preferences based on similar user behaviors.

Step-by-Step Implementation

Goal: Recommend products to a user based on what similar users have liked or purchased. We'll focus on **user-based collaborative filtering** using a matrix of user ratings.

Step 1: Collect and Prepare Data

Your dataset should consist of a matrix where rows represent users, columns represent items (products, movies, etc.), and values represent the ratings given by users to items.

- Example dataset:

User	Product A	Product B	Product C	Product D
User 1	5	0	3	1
User 2	4	0	0	1
User 3	1	1	0	5
User 4	0	1	5	4
User 5	0	0	4	4

Explanation:

- 0 means the user hasn't rated the product.

- Higher numbers (e.g., 5) indicate stronger preference.

Step 2: Calculate Similarity Between Users

The next step is to measure how similar users are to each other. Common similarity metrics include:

- **Cosine Similarity:** is a metric used to measure how similar two vectors are by calculating the cosine of the angle between them. In recommendation systems, it's commonly used to find how similar two users are based on their preferences or ratings.

In Python: Use the scipy library to calculate cosine similarity.

```
from sklearn.metrics.pairwise import cosine_similarity
import pandas as pd

data = {
    "Product A": [5, 4, 1, 0, 0],
    "Product B": [0, 0, 1, 1, 0],
    "Product C": [3, 0, 0, 5, 4],
    "Product D": [1, 1, 5, 4, 4]
}
df = pd.DataFrame(data)

similarity_matrix = cosine_similarity(df)
print(similarity_matrix)
```

The result will be a matrix showing the similarity between every pair of users.

Sample Output (Cosine Similarity Matrix)

	User 1	User 2	User 3	User 4	User 5
User 1	1.000	0.894	0.521	0.400	0.242
User 2	0.894	1.000	0.300	0.200	0.123
User 3	0.521	0.300	1.000	0.935	0.845
User 4	0.400	0.200	0.935	1.000	0.972
User 5	0.242	0.123	0.845	0.972	1.000

Step 3: Find Similar Users

Once the similarity matrix is created, you can identify the most similar users to the target user.

- Examples:
    - **User 1 and User 2** have a similarity score of **0.894**, meaning they have *very similar preferences*.
    - **User 3 and User 4** have a high similarity of **0.935**, indicating almost *identical preferences*.
    - **User 1 and User 4** have a low similarity of **0.4**, meaning they *are less similar*.

Step 4: Generate Recommendations

For each unrated item, predict the rating based on the ratings given by similar users. This can be done using a weighted average formula:

$$Predicted\ Rating = \frac{\sum(Similarity \times Rating)}{\sum Similarity}$$

Example: If User 1 hasn't rated **Product B**, you can predict the rating based on how similar users (User 3 and User 4) rated Product B.

$$Predicted\ Rating = \frac{(0.521 \times 1) + (0.4 \times 1)}{0.521 + 0.4}$$

$$Predicted\ Rating = \frac{0.921}{0.91} = \mathbf{1}$$

The predicted rating shows a very low score, indicating **there is little to no likelihood of User 1 purchasing Product B**. However, when we apply the same exercise for User 3 and Product C, we obtain a predicted rating **of 4.18/5, suggesting a much higher likelihood of purchase.**

$$Predicted\ Rating = \frac{(0.521 \times 3) + (0.935 \times 5) + (0.845 \times 4)}{0.521 + 0.935 + 0.845}$$

$$Predicted\ Rating = \frac{9.618}{2.301} = \mathbf{4.18}$$

Step 5: Recommend Top-N Items

After calculating the predicted ratings, recommend the **top-N items** with the highest predicted ratings.

- **Example:** *Suggest **Product C and Product D** to User 1 because their predicted ratings are the highest.*

Step 6: Evaluate and Optimize

- Continuously update the model with new user data to improve accuracy.

Take Action!

Effective segmentation leads to higher engagement, better conversion rates, and improved customer satisfaction. By implementing these models, you can personalize your marketing efforts and boost customer experience.