Data Science Intern Assignment | Zeotap

Task 1: Lookalike Model

Github- https://github.com/PriyabrataBehera-24/eCommerce_Transactions

Github code- https://github.com/PriyabrataBehera-24/eCommerce_Transactions/blob/main/Priyabrata_Behera_lookalike.ipynb

Here's the enhanced report for your Lookalike Model, formatted for clarity and professionalism:

---

1. Introduction

This report details the development and execution of a **Lookalike Model**, designed to identify customers with similar purchasing behaviors. By analyzing transaction data, product details, and customer demographics, the model generates comprehensive customer profiles and calculates similarity scores. These insights enable businesses to implement personalized marketing strategies and improve customer segmentation.

---

2. Data and Preprocessing

The analysis used three primary datasets:

1. **Customers.csv -** Includes CustomerID and Region, capturing demographic details.
2. **Products.csv -** Contains ProductID, Category, and Price, summarizing product information.
3. **Transactions.csv-** Records transaction data, including CustomerID, ProductID, Quantity, and TotalValue.

Data Merging and Aggregation:

- The datasets were merged to form enriched customer profiles.
- Profiles included **total spending**, **product category preferences**, and **transaction history**.
- Preprocessing steps ensured data quality, removed duplicates, and standardized column formats.

3. Methodology

The process of building the Lookalike Model involved the following steps:

3.1. Feature Engineering

- Customer profiles were converted into **numerical feature vectors**.
- Techniques such as **CountVectorizer** were applied to encode product preferences.
- **Total spending** was added as a numerical feature to capture overall customer activity.

3.2. Similarity Calculation

- Pairwise **cosine similarity** was used to measure the resemblance between customer feature vectors.
- The metric assigns scores between 0 and 1, where 1 indicates perfect similarity.

3.3. Lookalike Identification

- The top 3 most similar customers were identified for each of the 20 selected customers.
- Self-similarity was excluded to ensure meaningful recommendations.

---

4. Results

The model successfully generated a **Lookalike Map**, detailing the most similar customers for each target individual.

- This map is saved as **lookalike.csv**, enabling further analysis and integration into business applications.

---

5. Conclusion

The Lookalike Model provides a robust framework for identifying customers with similar purchasing patterns.

Business Applications:

- **Targeted Advertising:** Reach groups with aligned interests.
- **Personalized Recommendations:** Suggest products based on similar customer behavior.
- **Customer Segmentation:** Refine customer groupings to inform strategic decisions.

6. Code Implementation

The model was implemented in Python using the following libraries:

- **pandas**: For data manipulation.
- **scikit-learn**: For feature vectorization and similarity computation.
- **scipy**: For efficient similarity calculations.

Code Features:

1. Data loading and cleaning.
2. Creation of customer profiles.
3. Feature vectorization and aggregation.
4. Similarity computation and lookalike identification.
5. Saving results in a structured format for easy access.