**[sales-customer-recommendation](https://github.com/praveennani384/sales-customer-recommendation)**

**I. Introduction**

* **Project Title:** E-commerce Data Analysis App
* **Brief Overview:** This Streamlit web application analyzes e-commerce data, providing sales forecasts, customer segments, and product recommendations.
* **Purpose:** It helps businesses understand customer behavior, predict sales, and offer personalized recommendations.
* **Target Audience:** This documentation is for developers, data scientists, or business users.

**II. Data**

* **Dataset:**
  + **Name and Source:** "Online Retail.xlsx" dataset from UCI Machine Learning Repository.
  + **Description:** This dataset contains transactional data for an online retail store between 2010 and 2011.
* **Data Cleaning and Preprocessing:**
  + **Missing Values:** Removed rows with missing customer IDs.
  + **Data Types:** Converted the CustomerID column to an integer type.
  + **Synthetic Data:** Generated synthetic customer demographics using the Faker library.
* **Feature Engineering:**
  + **RFM Features:** Created RFM features (Recency, Frequency, Monetary) for customer segmentation.

**III. Models and Algorithms**

* **Sales Forecasting:**
  + **Algorithm:** ARIMA model for time series forecasting.
  + **Parameters:** ARIMA order (7,1,2).
  + **Preprocessing:** Data aggregated to the daily level, and lagged features used to capture seasonal variation.
* **Customer Segmentation:**
  + **Algorithm:** K-Means clustering.
  + **Clusters:** k=3 was chosen based on the elbow method analysis
  + **Features:** RFM features are used for clustering.
* **Product Recommendation:**
  + **Method:** Item-based collaborative filtering.
  + **Similarity:** Cosine similarity.
  + **Recommendations:** Top 5 products are recommended.

**IV. Application Architecture**

* **Overview of Components:**
  + Streamlit app as the user interface.
  + Includes logic for data loading, preprocessing, and model implementation.
* **Functionality Breakdown:**
  + **Sales Forecasting:** Provides sales predictions over time.
  + **Customer Segmentation:** Shows different customer segments and their mean features.
  + **Product Recommendation:** Provides top product recommendations for a given user.

**V. Installation and Setup**

* **System Requirements:**
  + Python 3.8 or higher.
  + Required packages (specified in requirements.txt).
* **Setup Steps:**
  + Clone the GitHub repository.
  + Install the requirements: pip install -r requirements.txt.
  + Place the Online Retail.xlsx dataset, trained model files (arima\_model.pkl, kmeans\_model.pkl, scaler.pkl, item\_similarity.joblib), and style.css in the same directory as app.py.
  + Run Streamlit: streamlit run app.py.

**VI. Usage**

* **Navigation:**
  + Use the sidebar to navigate to different sections: "Sales Forecasting", "Customer Segmentation", "Product Recommendation."
* **Sales Forecasting:** This page displays a combined graph for the train and test dataset.
* **Customer Segmentation:** View customer segments, with an option to specify customer id.
* **Product Recommendation:** Enter the customer ID, and get the top product recommendations.

**VII. Limitations and Future Work (Optional)**

* **Current Limitations:**
  + The ARIMA model might require further hyperparameter tuning.
  + The recommendation system is based on a simple collaborative filtering approach.
* **Potential Improvements:**
  + Hyperparameter tuning of the ARIMA model to further refine results.
  + Use other time-series models for more refined sales forecasting.
  + Explore alternatives to item-based collaborative filtering to improve the product recommendations.