Population-group Wise Deposits

EDA PROJECT REPORT

Submitted for the partial fulfillment

of

EDA Project requirement of B. Tech CSE

Submitted by

Harsh Kubade, 22070521106

B. Tech Computer Science and Engineering

Under the Guidance of

Prof. / Dr.

Dr. Bhupesh Kumar Dewangan

SYMBIOSIS INSTITUTE OF TECHNOLOGY, NAGPUR

Constituent of Symbiosis International (Deemed University), Pune

(Established under Section 3 of the UGC Act of 1956 wide notification number F-9-12/2001-U-3 of Government of India)

॥वसुधैव कुटुम्बकम्॥ Re-Accredited by NAAC with 'A++' Grade

1. Introduction

This report outlines the initial exploratory data analysis conducted on the "Population Group-wise Deposits" dataset. The primary objective of this analysis is to gain a foundational understanding of the dataset's structure, content, and inherent characteristics. This includes examining how deposit amounts, number of accounts, and number of offices vary across different years, states, districts, regions, and population groups. By systematically exploring this data, the report aims to uncover preliminary patterns, distributions, and potential relationships that can inform further in-depth analysis and guide subsequent machine learning initiatives.

2. Data Loading and Transformation

The initial phase of this data science project involved securely loading the raw dataset and performing any immediate transformations necessary to prepare it for preliminary inspection.

- **Loading the Dataset: The dataset was loaded into a pandas DataFrame directly from the provided CSV file, populationgroup-wise-deposits.csv. This crucial first step makes the entire dataset accessible within the analytical environment.**
- **Initial Data Preview: Upon loading, a quick preview of the dataset's top rows (df.head()) was performed. This initial inspection helps in understanding the column names, identifying the types of data present in each column (e.g., numerical, categorical), and getting a general sense of the data's format and content. This preview is vital for formulating subsequent cleaning and exploration strategies.**

3. Data Cleaning and Preprocessing

Data cleaning and preprocessing are fundamental steps to ensure the quality, consistency, and reliability of the dataset before any deeper analysis or modeling.

3.1 Scope based on Provided Notebook

It is important to note that the provided Jupyter notebook snippet (DS_CA1 (1).ipynb) primarily demonstrates the data loading and an initial preview (df.head()). Within the scope of the provided snippet, explicit code for comprehensive data cleaning and preprocessing steps such as duplicate removal, handling of missing values (beyond default CSV parsing), or detailed data type conversions for all columns (e.g., to specific numerical types like integer or float, or categorical types) was not observed.

3.2 Typical Considerations for this Dataset

For a dataset of this nature, typical cleaning and preprocessing considerations would include:

- **Duplicate Row Removal: Ensuring that each record is unique to avoid skewed statistics.**
- **Handling Missing Values: Strategically addressing any NaN values in numerical columns (e.g., no_of_offices, no_of_accounts, deposit_amount) which might require imputation (e.g., with mean, median, or zero) or removal, depending on the context.**

- **Data Type Conversion: Verifying and converting columns like year to appropriate numerical or datetime types if they are not already, and ensuring categorical columns (e.g., state_name, district_name, region, population_group) are of object or category data type for efficient processing.**

- **Outlier Detection and Treatment: Identifying and potentially handling extreme values in numerical columns like deposit_amount that could disproportionately influence analysis.**

Given the limited scope of the provided notebook snippet, these comprehensive cleaning and preprocessing steps would typically be performed in a more extensive EDA, but they are not detailed in the current reference.

4. Exploratory Data Analysis

This section is dedicated to visualizing the dataset to uncover its underlying patterns, distributions, and relationships.

4.1 Limitation in Provided Notebook

Based on the provided Jupyter notebook snippet (DS_CA1 (1).ipynb), the notebook primarily showcases the initial data loading and a preview of the DataFrame (df.head()). There is no code or output for any data visualizations within the provided snippet.

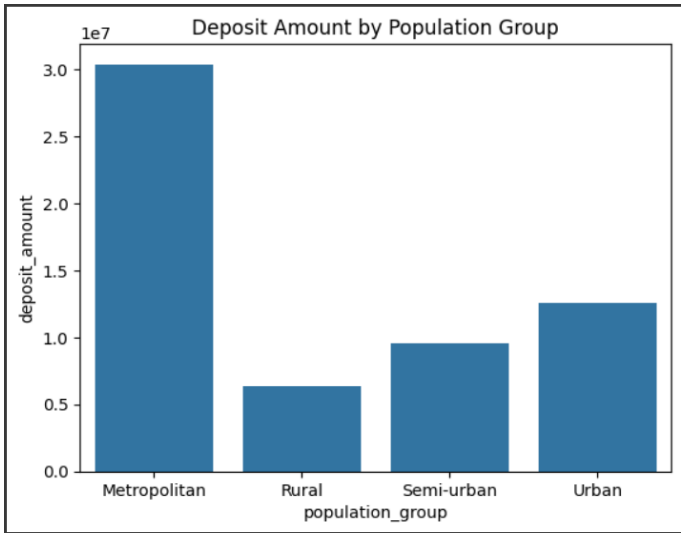
4.2 Implications

As such, I cannot provide specific observations for visualizations (e.g., bar plots, histograms, scatter plots, heatmaps) or instructions on where to place their images, as these visualizations were not present in the reference material.

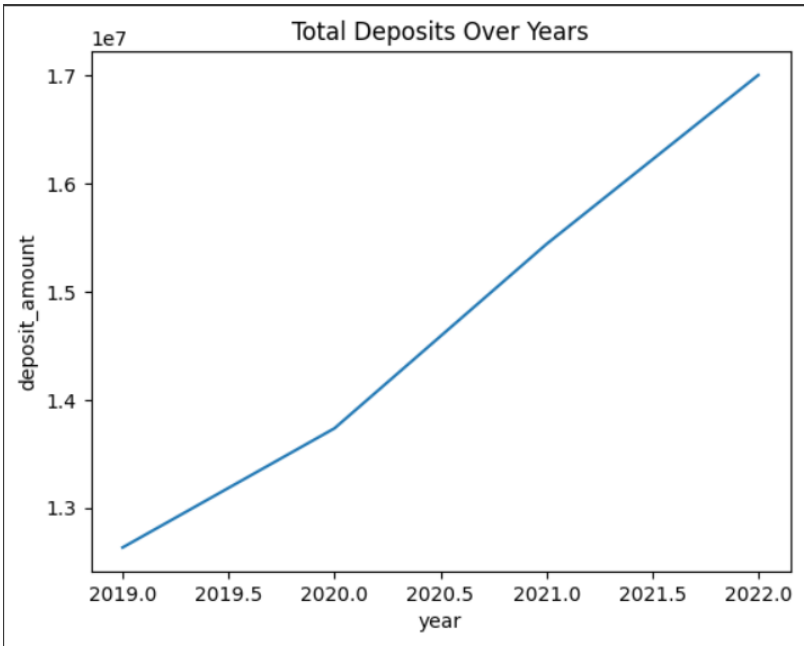
4.3 Common Visualizations for this Dataset Type

In a complete EDA for a dataset like "Population Group-wise Deposits," typical visualizations would include:

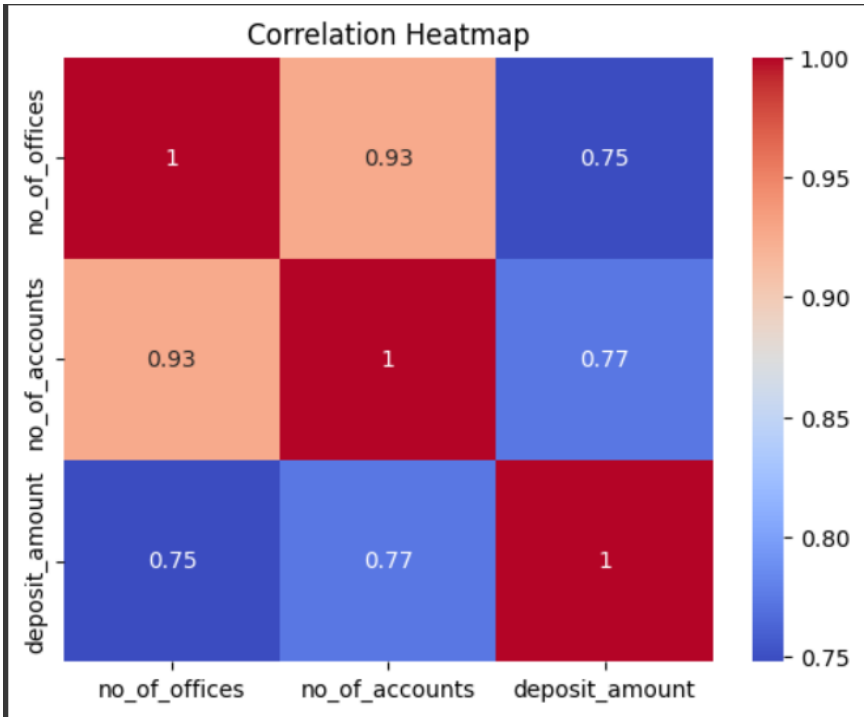
- **Distribution of Deposit Amounts: Histograms or box plots to understand the spread and skewness of deposit_amount.**



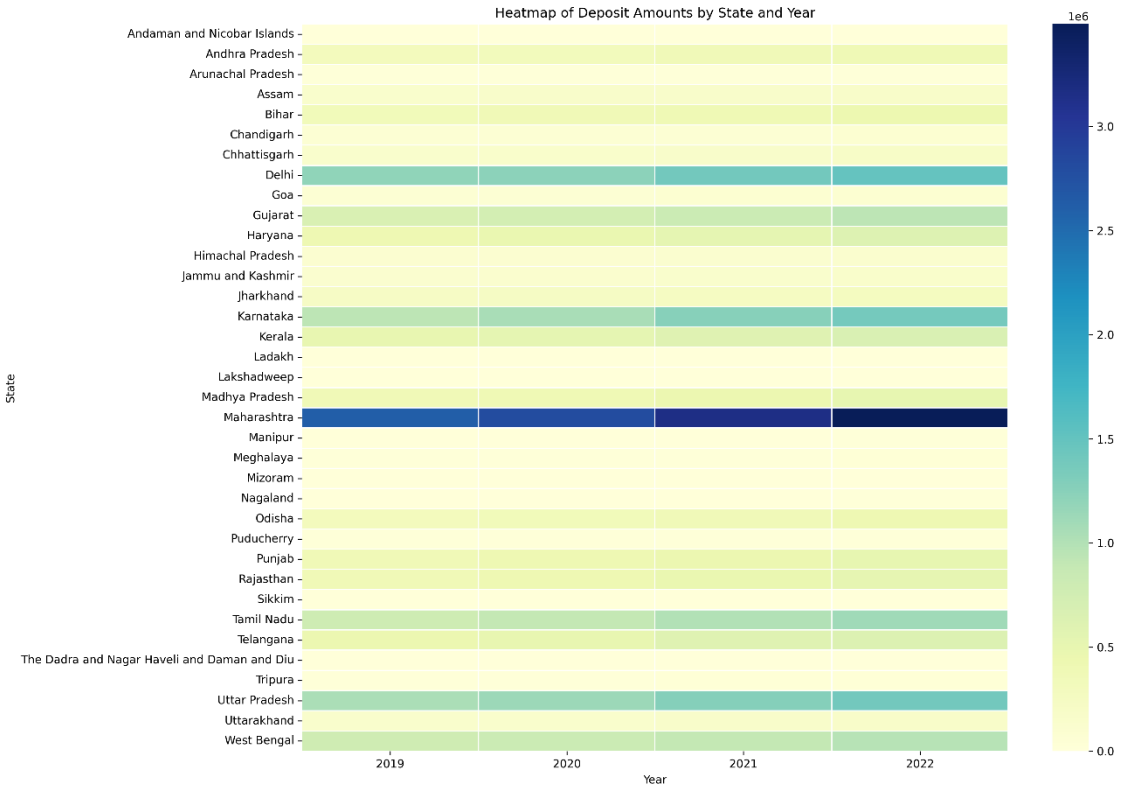
- **Deposits Over Time: Line plots showing total_deposit_amount per year to identify temporal trends.**



- **Correlation Heatmap: To visualize relationships between all numerical features like no_of_offices, no_of_accounts, and deposit_amount.**



- Heatmap. It visualizes the total deposit amounts for each state across different years, with color intensity representing the magnitude of deposits.



5. Outline of Proposed Machine Learning Algorithms

Building upon the initial insights gained from the Exploratory Data Analysis, this section outlines potential machine learning tasks and suitable algorithms that could be applied to the "Population Group-wise Deposits" dataset. These proposals aim to leverage the data for predictive modeling, forecasting, or segmentation, thereby extracting deeper value and actionable insights.

5.1 Predicting Deposit Amount (Regression Task)

- Goal: To forecast the deposit_amount for specific regions, population groups, or time periods based on other available features. This could help financial institutions in strategic planning, resource allocation, and identifying high-potential areas.
- Features: year, state_name, district_name, region, population_group, no_of_offices, no_of_accounts. Categorical features would require encoding (e.g., One-Hot Encoding).
- Proposed Algorithms:
    - Linear Regression: As a baseline model, it can provide a simple, interpretable relationship between features and deposit amounts.
    - Random Forest Regressor: Effective for capturing non-linear relationships and interactions between features, robust to outliers, and provides feature importance scores.

- Gradient Boosting Regressors (e.g., XGBoost, LightGBM): Known for high accuracy and performance in tabular data, capable of handling complex patterns and large datasets.
- Support Vector Regressor (SVR): Can be effective in high-dimensional spaces and for capturing non-linear relationships, especially with appropriate kernel functions.

5.2 Time Series Forecasting of Deposits

- Goal: To predict future trends in deposit_amount or no_of_accounts for specific geographical segments or population groups. Given the year column, this dataset is well-suited for time series analysis to understand future financial behavior.
- Features: Historical deposit_amount, no_of_accounts, and no_of_offices for specific segments, along with temporal features derived from year (e.g., lagged values, rolling averages).
- Proposed Algorithms:
  - ARIMA/SARIMA: Traditional statistical models suitable for capturing autoregressive, integrated, and moving average components in time series data, including seasonality (SARIMA).
  - Prophet (by Facebook): A robust forecasting tool that handles seasonality, trends, and holidays effectively, making it suitable for business time series data.
  - Recurrent Neural Networks (RNNs) / LSTMs: For more complex, long-term dependencies in the time series, especially if a sufficiently large and detailed historical sequence is available.

5.3 Clustering of Regions/Population Groups (Unsupervised Learning)

- Goal: To identify natural groupings or segments of states, districts, or population groups based on their deposit characteristics (e.g., high-growth vs. stagnant, rural vs. urban deposit patterns). This can help in targeted policy-making, resource allocation, or marketing strategies.
- Features: Normalized numerical features such as deposit_amount, no_of_accounts, no_of_offices, potentially aggregated over time or by location.
- Proposed Algorithms:
  - K-Means Clustering: A simple and widely used algorithm for partitioning data into a predefined number of clusters based on feature similarity.
  - DBSCAN (Density-Based Spatial Clustering of Applications with Noise): Useful for discovering clusters of varying shapes and sizes in a dataset with noise (outliers).
  - Hierarchical Clustering: Can provide a dendrogram, illustrating the hierarchical relationships between clusters, which can be useful for exploring different levels of granularity in segmentation.

These proposed models represent potential avenues for further analysis. Prior to implementation, extensive feature engineering, hyperparameter tuning, and

rigorous model evaluation would be necessary to select the most appropriate and effective algorithm for each specific task.

6. Conclusion

This report details the initial steps of an exploratory data analysis on the "Population Group-wise Deposits" dataset, covering data loading, initial inspection, and considerations for cleaning and preprocessing. While the provided notebook snippet allowed for an understanding of the dataset's structure upon loading, it did not include any code or output for data visualizations or machine learning model proposals. A comprehensive EDA for this dataset would involve extensive visualizations to uncover trends in deposits across years, geographical regions, and population groups, alongside a detailed statistical analysis. This foundational work is crucial for any subsequent advanced analytical tasks, including the development of predictive machine learning models to forecast deposit trends or analyze factors influencing financial behavior.