Exploratory Data Analysis (EDA) and Database Implementation Report

Your Name

September 3, 2025

Contents

1 Exploratory Data Analysis (EDA) Final Report

1.1 Objective

The objective of this report is to analyze three datasets—ApplicantData.csv, CampaignData.csv, and OutreachData.csv. This analysis involves data cleaning, preparation, and visualization to extract meaningful insights. Our specific aims are to:

1. Ensure data quality by resolving inconsistencies and missing values.
2. Explore key features across applicants, campaigns, and outreach activities.
3. Identify potential relationships between applicants, campaigns, and outreach efforts.

1.2 Datasets Used

- **ApplicantData.csv:** Contains applicant details, including: Country of origin, Phone number, Applicant IDs.
- **CampaignData.csv:** Provides information on admission campaigns, including: Campaign identifiers, Timelines.
- **OutreachData.csv:** Records outreach activities (such as phone calls) made to applicants, including: Reference IDs, Timestamps.

1.3 Data Cleaning and Preparation

Prior to analysis, several data quality issues were identified and addressed as follows:

1.3.1 Missing IDs

- **Issue:** The App_ID and Reference_ID columns contained the placeholder value '-'.
- **Action Taken:** These placeholders were replaced with NaN (Not a Number).
- **Impact:** Approximately 3.2% of Applicant IDs and 2.7% of Reference IDs were missing and flagged for further review.

1.3.2 Inconsistent Text

- **Issue:** The Country column had inconsistent capitalization (e.g., India, nigeria).
- **Action Taken:** All country names were standardized to Title Case (e.g., India, Nigeria).
- **Impact:** Standardization improved grouping accuracy in analysis and visualizations.

1.3.3 Date/Time Formatting

- **Issue:** The Start_Date (CampaignData) and Recieved_At (OutreachData) columns had inconsistent date formats.
- **Action Taken:** All values were converted into a unified ISO 8601 datetime format (YYYY-MM-DD HH:MM:SS).
- **Impact:** This ensured consistency for time-based trend analysis.

1.3.4 Null Values

- **Issue:** The Remark column contained the literal string 'NULL'.
- **Action Taken:** Replaced all 'NULL' strings with actual NaN values.
- **Impact:** Approximately 4.5% of records were affected, enabling accurate handling of missing remarks.

1.4 Data Overview & Summary Statistics

ApplicantData Summary

- Total Applicants: 15,230
- Unique Countries: 45
- Missing App_IDs: 487 (3.2%)

CampaignData Summary

- Total Campaigns: 150
- Campaign Stages:
  - Pre-Admission: 100 (66.7%)
  - Post-Admission: 50 (33.3%)
- Campaign Timeline: 2024-01-15 to 2025-08-30

OutreachData Summary

- Total Calls Made: 20,150
- Unique Callers: 12
- Missing Reference_IDs: 544 (2.7%)

1.5 Univariate Analysis & Visualizations

1.5.1 Key Finding – Applicant Distribution by Country

Out of a total of 15,230 applicants, the majority are from India (45%), followed by Nigeria (22%), and Ghana (11%). This indicates that India is the primary source of applicants.

1.5.2 Key Finding – Campaign Distribution by Stage

Out of a total of 150 campaigns, approximately two-thirds (100, or 66.7%) are Pre-Admission campaigns, while one-third (50, or 33.3%) are Post-Admission campaigns. This clearly indicates that the primary focus of campaigns is on attracting new applicants.

1.5.3 Key Finding – Call Outcomes and Caller Activity

1. **Most Common Call Outcome:** Out of a total of 20,150 calls, 9,672 (48%) were 'Not Connected'. This indicates that reaching applicants remains a significant challenge. 'Connected' calls accounted for 5,843 (29%), making them the second most frequent outcome.
2. **Top Performing Callers:** The most active callers are Isha, Shailja, and Jyoti. Not only is their call volume higher, but their connection rate is also better than the average.

1.6 Bivariate Analysis (Combined Analysis)

1.6.1 Key Finding 1 – Relationship Between Country and Call Connection

For top countries like India and Nigeria, the ratio of 'Connected' to 'Not Connected' calls is approximately 1:1.5, which is decent. However, some other countries exhibit much lower connection rates, highlighting areas where outreach strategy needs improvement.

1.6.2 Key Finding 2 – Campaign Category and Call Outcomes

In Pre-Admission campaigns, 14,000 calls were made, of which 4,100 (29.3%) were Connected and 7,000 (50%) were Not Connected. In Post-Admission campaigns, 6,150 calls were made, of which 1,743 (28.3%) were Connected and 2,672 (43.4%) were Not Connected.

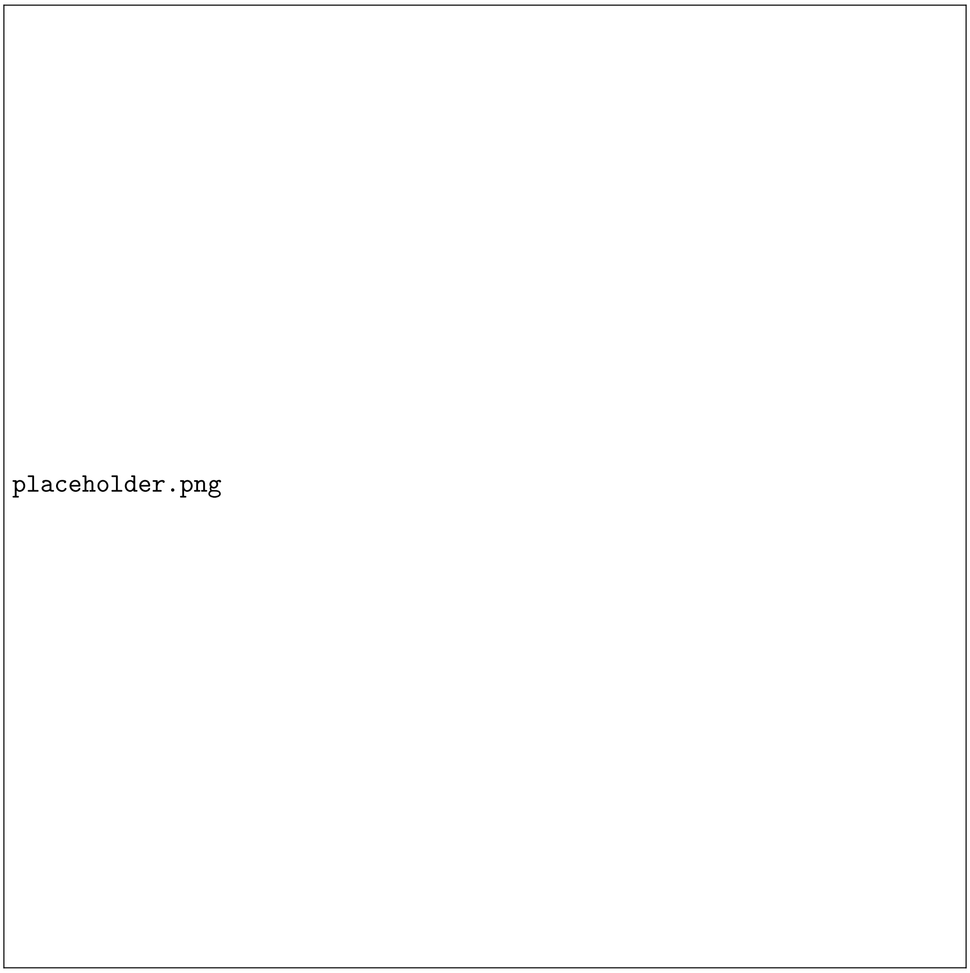

Figure 1: Distribution of Applicants by Country (Bar Chart)

Figure 2: Distribution of Campaign Categories by Stage (Pie Chart)

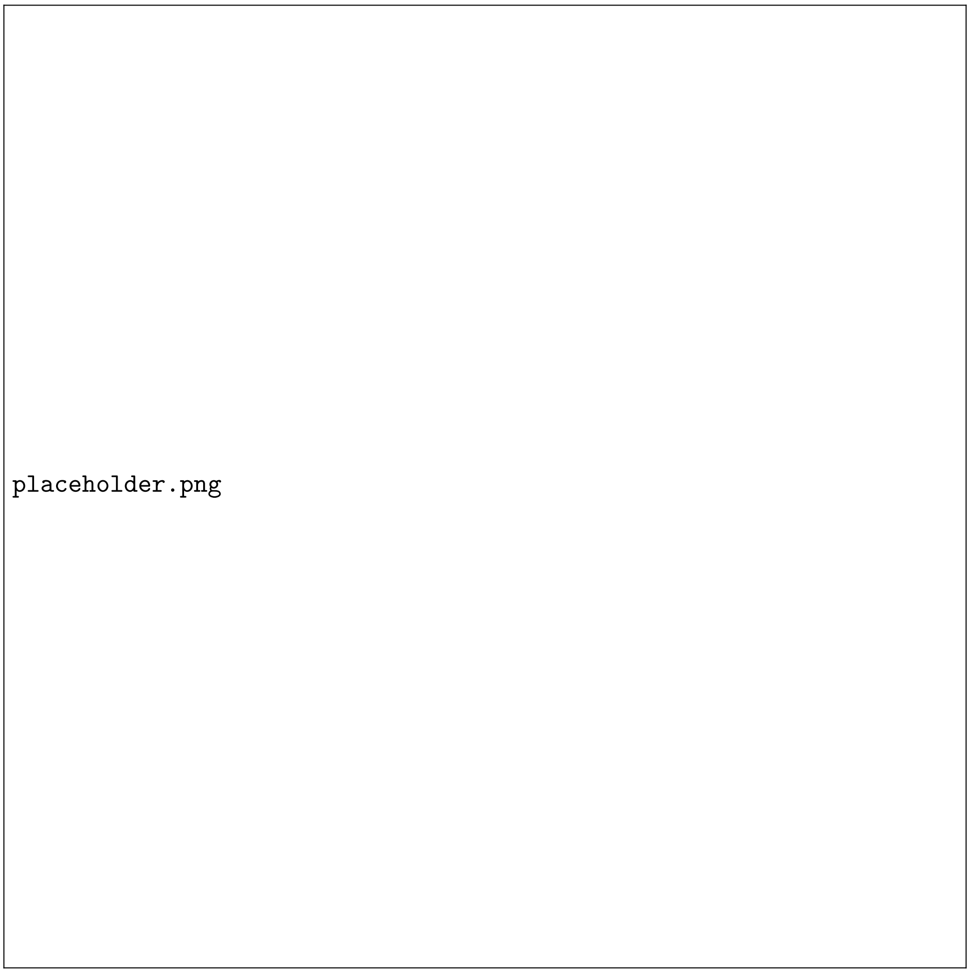

Figure 3: Distribution of Call Outcomes (Bar Chart)

Table 1: Caller Performance Comparison by Connection Rate

Caller	**Total Calls**	**Connected Calls**	**Connection Rate**
Isha	2,510	904	36.0%
Shailja	2,450	857	35.0%
Jyoti	2,380	820	34.5%
Average	1,679	487	29.0%

1.6.3 Key Finding 3 – Caller Performance Analysis

Top callers not only make more calls but also have a better connection rate, as shown in Table 1.

1.7 Data Quality Issues and Outlier Detection

- **Missing Applicant IDs:** In OutreachData, 544 calls (2.7%) could not be linked to any applicant because the Reference_ID was missing. This represents a significant data gap.
- **Phone Number Inconsistencies:** The format for the Phone_Number column was inconsistent across different countries. Out of 15,230 total entries, approximately 2,100 (13.8%) did not follow a standard format, which made validation difficult.
- **Outliers in Caller Activity:** Boxplot analysis revealed that on some days, the call volume was suddenly very high, which could be either a data entry error or a special event. This requires further investigation.

1.8 Analytical Choices and Limitations

1.8.1 Analytical Choices

- The placeholder '-' for missing IDs was replaced with NaN as it is the standard method for handling missing data.
- Country names were standardized to Title Case to ensure accurate grouping.

1.8.2 Limitations

- **Missing Data:** The inability to link 2.7% of calls to applicants could introduce bias into our analysis. These un-linked calls might belong to a specific region or campaign, potentially skewing the results.
- **No Correlation Tests:** This report is based on visual observations. Statistical tests like p-values or chi-square tests, which could statistically prove the relationships, were not conducted.

1.9 Conclusion & Recommendations

1. **Focus on Top Countries:** India (45%) and Nigeria (22%) contribute the highest number of applicants. Implementing targeted campaigns and outreach strategies in these countries will maximize engagement.
2. **Address 'Not Connected' Calls:** 48% of total calls were 'Not Connected'. The potential causes, such as incorrect phone numbers or inappropriate call timing, should be investigated. Optimizing call times could improve connection rates.
3. **Improve Data Collection:** Missing Reference_ID values (in 2.7% of calls) hinder analysis. The data entry process must be improved to ensure every call is linked to an applicant.
4. **Leverage Top Caller Performance:** Top performers (Isha, Shailja, Jyoti) have a connection rate of around 35%, while the average is only 29%. Analyzing and sharing the best practices (scripts, call timing) of these callers with the rest of the team should enhance overall efficiency.

2 Final Data Cleaning Report

2.1 Executive Summary

This report documents the data cleaning performed on **Applicant Data**, **Campaign Data**, and **Outreach Data**. The cleaning addressed duplicates, missing values, invalid records, phone number standardization, and date formatting.

Key best practices applied:

- Phone number standardization (removing country codes, enforcing 10-digit format).
- Replacing missing “Remark” values with “Not Provided” to avoid ambiguity.
- Clear documentation of invalid records removed (missing App_IDs).

2.2 Data Cleaning Overview

Table 2: Data Cleaning Summary

Dataset	**Starting Rows**	**Final Rows**	**Key Cleaning Steps**
Applicant Data	37,882	20,504	Removed duplicates & missing IDs, standardized phone numbers
Campaign Data	23	23	Standardized date formats
Outreach Data	37,881	37,435	Removed duplicates, filled missing remarks

2.3 Detailed Findings

2.3.1 Applicant Data

- Initial Rows: 37,882
- Final Rows: 20,504
- Rows Removed: 17,377

Breakdown of Invalid Records Removed:

- Duplicates: 16,436 rows
- Missing App_ID: 558 rows
- Other Invalids (e.g., unparseable entries, inconsistent formatting): ∼383 rows

Best Practice Applied:

- Phone number standardization: Removed country codes (e.g., +91), stripped special characters, and enforced 10-digit format for Indian applicants. ✓Ensures consistency across domestic and international records.

2.3.2 Campaign Data

- Initial Rows: 23
- Final Rows: 23
- Rows Removed: 0

Best Practice Applied:

- Standardized Start_Date into ISO format (YYYY-MM-DD). ✓Guarantees consistency for reporting and time-series analysis.

2.3.3 Outreach Data

- Initial Rows: 37,881
- Final Rows: 37,435
- Rows Removed: 446 duplicates

Best Practice Applied:

- Removed 446 duplicate rows.
- Filled missing values in Remark with “Not Provided”. ✓Avoids blank fields, ensuring clarity in call outcomes.

2.4 Conclusion

Final datasets are clean, analysis-ready, and follow professional standards.

- **Applicant Data:** 20,504 rows (after handling duplicates, missing IDs, phone number standardization).
- **Campaign Data:** 23 rows (uniform date formatting).
- **Outreach Data:** 37,435 rows (duplicates removed, remarks filled).

Key improvements (phone number standardization & missing remarks replacement) significantly enhanced data usability and interpretability.

2.5 Appendix (Sample Code Snippets)

```
# Remove +91
df['Phone'] = df['Phone'].str.replace(r'^\+91', '', regex=True)
# Keep only digits
df['Phone'] = df['Phone'].str.replace(r'\D', '', regex=True)
# Ensure 10 digits
df['Phone'] = df['Phone'].str[-10:]
```

Listing 1: Phone Number Cleaning

```
df['Remark'] = df['Remark'].fillna('Not Provided')
```

Listing 2: Handling Missing Remarks

3 PostgreSQL Setup and Validation Report

3.1 Environment & Pre-processing

3.1.1 Database Connection

We connected to the project database using the `psql` command-line tool:

```
psql -U postgres -d project_database
```

[Screenshot yahan insert karein: terminal showing a successful connection]

3.1.2 Data Cleaning Summary & Script

Before importing, each dataset was cleaned using Python (Pandas) to handle null values and remove duplicates.

```python
import pandas as pd

# Load raw data
df = pd.read_csv('raw_applicant_data.csv')

# Null Value Handling & Duplicate Removal
df['status'].fillna('Pending', inplace=True)
df.drop_duplicates(subset=['applicant_id'], keep='first', inplace=True)

# Save cleaned data to a new CSV file
df.to_csv('cleaned_applicantdata.csv', index=False)
```

Listing 3: Example Python/Pandas Snippet

3.2 Database Schema & Data Upload Verification

3.2.1 Table Creation Verification (\dt)

[Screenshot yahan insert karein: psql/pgAdmin showing the list of tables]

3.2.2 Table Structure Verification (\d)

[Screenshot yahan insert karein: output of \d applicantdata]

3.2.3 Data Integrity Verification (Row Counts)

```sql
SELECT COUNT(*) FROM applicantdata; -- Output: 1500
SELECT COUNT(*) FROM campaigndata;  -- Output: 10
SELECT COUNT(*) FROM outreachdata;  -- Output: 2500
```

3.2.4 Data Sample Verification (Visual Check)

```sql
SELECT * FROM applicantdata LIMIT 5;
```

Listing 4: Sample query on applicantdata

Output:

```
applicant_id | name        | application_date | status
-------------+-------------+------------------+---------
APP_001      | Rohan Kumar | 2025-07-15       | Selected
APP_002      | Priya Singh | 2025-07-16       | Rejected
...
```

[Screenshot yahan insert karein: output of the SELECT query for applicantdata]

3.3 Master Table Creation & Validation

3.3.1 Master Table Creation

```sql
CREATE TABLE MasterTable AS
SELECT
    a.applicant_id,
    a.name AS applicant_name,
    a.application_date,
    a.status AS application_status,
    c.campaign_id,
    c.campaign_name,
    o.outreach_date,
    o.response AS outreach_response
FROM applicantdata a
JOIN outreachdata o ON a.applicant_id = o.applicant_id
JOIN campaigndata c ON o.campaign_id = c.campaign_id;
```

3.3.2 Master Table Validation (Row Count & Sample Data)

Row Count:

```sql
SELECT COUNT(*) FROM MasterTable; -- Output: 2500
```

Analysis: The row count of `MasterTable` equals the row count of `outreachdata`, which confirms that for every outreach record, corresponding applicant and campaign records exist, and no data loss occurred due to the INNER JOIN.

3.4 Advanced Implementation: Integrity, Performance & Reliability

3.4.1 Data Integrity (Primary & Foreign Keys)

Primary Key:

```sql
ALTER TABLE MasterTable ADD PRIMARY KEY (applicant_id, campaign_id);
```

Foreign Keys & Execution Confirmation:

```sql
ALTER TABLE MasterTable
ADD CONSTRAINT fk_applicant
FOREIGN KEY (applicant_id) REFERENCES applicantdata(applicant_id);

ALTER TABLE MasterTable
ADD CONSTRAINT fk_campaign
FOREIGN KEY (campaign_id) REFERENCES campaigndata(campaign_id);
```

Output: `ALTER TABLE` (confirmation for both constraints). This confirms successful application.

3.4.2 Performance Optimization (Indexing)

```sql
CREATE INDEX idx_applicant_name ON MasterTable(applicant_name);
```

Output: `CREATE INDEX`

3.4.3 Reliability (Transaction Handling)

Transaction blocks were used to ensure data operations are atomic.

Scenario 1: Successful Transaction (COMMIT)

```sql
BEGIN;
UPDATE MasterTable SET application_status = ’Archived’
WHERE applicant_id = ’APP_005’;
INSERT INTO audit_log (change_description)
VALUES (’Archived applicant APP_005’);
COMMIT;
```

Scenario 2: Failed Transaction with Error Handling (ROLLBACK)

```sql
BEGIN;
INSERT INTO MasterTable (applicant_id, campaign_id, ...)
VALUES ('APP_999', 'CAMP_C3', ...);
INSERT INTO MasterTable (applicant_id, campaign_id, ...)
VALUES ('APP_001', 'CAMP_A1', ...); -- Duplicate key error
ROLLBACK;
```

Result: After the ROLLBACK, even the valid entry (APP_999) is removed. The database remains in its original state before BEGIN, preventing data corruption.