Automated Data Pipeline Report for Non-Resident Tourist Travel Data

Twaidit Singh Luthra
Mtr. No. - 23194090
Data Report

27th November,2024

Question

How do travel patterns of non-resident tourists visiting Argentina change by country of origin and mode of transport over time, and what trends emerge from this data?

Data Sources

1. Dataset Description

I collected a dataset that records travel data for non-resident tourists visiting Argentina. The dataset includes information about their countries or regions of origin, the modes of transport they used, and monthly data on total arrivals. I selected this dataset to gain insights into international tourism patterns, which are crucial for understanding Argentina's global connectivity and the economic impact of tourism.

2. Source Details

The data was provided by Yvera Platform, Argentina's official open data portal for tourism. I used the Tourism Dataset for my analysis respectively.

3. Data Structure and Quality

The dataset, formatted as a CSV file (structured data), includes the following attributes:

1. The origin of tourists (e.g., country or group of countries)
2. The mode of transport used
3. Monthly data on tourist counts

When I reviewed the dataset, I noticed inconsistent column naming conventions, missing values, and occasional formatting errors in numeric data.

4. License and Usage

The dataset operates under an open-data license from the Yvera platform, allowing use in public projects with proper attribution. I ensured compliance by citing the source in all publications derived from this project.

Data Pipeline

1. An Overview

I designed and implemented a data pipeline to process the dataset through the following stages:

1. **Download:** I retrieved the dataset using Python's `urllib`, bypassing SSL verification to resolve certificate issues.
2. **Transform:** I cleaned, standardized, and reformatted the data using `pandas` to ensure consistency.
3. **Load:** I stored the transformed data in an SQLite database, which makes it easily queryable for analysis.

2. Transformation and Cleaning Steps

I implemented the following data cleaning and transformation steps: I converted all column headers to lowercase and replaced spaces with underscores for consistency. I dropped missing entries, which accounted for a small portion of the dataset.I converted numeric columns explicitly and coerced invalid entries into `NaN`. I verified that all fields adhered to the expected data types.

3. Challenges and Solutions

During the process, I encountered the following challenges and addressed them effectively:

1. **SSL Certificate Errors:** I disabled SSL verification by programming using a custom `SSLContext` to ensure seamless data retrieval.
2. **Inconsistent Data Formatting:** I applied transformations to standardize column names and data types.
3. **Directory Management:** I organized the project by maintaining a structured directory to separate scripts and output files.

4. Meta-Quality Measures

I included the following measures to ensure the quality and reliability of the pipeline:

1. **Error Handling:** I incorporated exception handling for data download and transformations, logging issues and enabling the pipeline to stop gracefully if failures occurred.

2. **Dynamic Input Adaptability:** I designed the pipeline to dynamically adapt to changes in column names, standardizing them to prevent downstream errors.

3. **Timeliness & Relevancy :** I ensured that the data is relevant and updated periodically.

Result and Limitations

1. Output Data

My pipeline outputs an SQLite database named `tourism_data.db`, which contains a cleaned and standardized table (`tourism_data`) with records of non-resident tourist travel data.

2. Data Structure and Quality

The database table includes:

- `origin`: The country or group of origin (text)
- `mode_of_transport`: The mode of transport (text)
- `date`: The time period (date or string)
- `amount_of_tourists`: The count of tourists (integer)

I significantly improved data quality by handling missing values, ensuring type consistency, and removing invalid entries.

3. Data Format Choice

I chose **SQLite** as the output format because:

It provides an efficient and portable format for structured queries. It integrates seamlessly with Python, making analysis straightforward.

4. Critical Reflection and Anticipated Issues

I identified the following limitations and considerations:

1. **Data Bias:** The dataset may not account for informal or undocumented travel.

2. **Temporal Gaps:** Missing or inconsistent data for certain months could affect the reliability of trend analysis.

3. **Limited Attributes:** The dataset focuses primarily on travel numbers and does not include contextual data such as demographics or economic indicators.

I plan to address these limitations in future work by integrating supplementary datasets to enable a more comprehensive analysis.