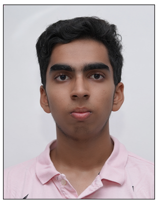
Exploratory Data Analysis on

Foreign Tourist Arrivals and their Economic Impact on India

by

Group 9

Dishant Patel
ID: 202201260
Course: BTech ICT



Madhav Kanjilimadom
ID: 202203018
Course: BTech MnC

Shravan Kakadiya
ID: 202201333
Course: BTech ICT

Course Code: IT 462
Semester: Autumn 2024

Under the guidance of

Dr. Gopinath Panda

Dhirubhai Ambani Institute of Information and Communication Technology

July 15, 2025

Acknowledgment

I am writing this letter to express my heartfelt gratitude for your guidance and support throughout the duration of my project titled ”Analyzing Foreign Tourist Arrivals and Their Economic Impact on India”. Your invaluable assistance has played a pivotal role in shaping the successful completion of this endeavour.

I am extremely fortunate to have had the opportunity to work under your mentorship. Your expertise, encouragement, and willingness to share your knowledge have been instrumental in elevating the quality and scope of my project. Your constructive feedback and insightful suggestions have helped me overcome challenges and develop a deeper understanding of the subject matter.

Furthermore, I would like to extend my appreciation to the entire team at Dhirubhai Ambani Institute of Information and Communication Technology for fostering an environment of collaboration and innovation. The resources and facilities provided have been crucial in conducting comprehensive research and analysis.

I would also like to express my gratitude to my peers and colleagues who have been supportive throughout this journey. Their valuable input and camaraderie have been a constant source of motivation.

Completing this project has been a tremendous learning experience, and I am confident that the knowledge and skills acquired during this endeavor will serve as a solid foundation for my future endeavors.

Once again, thank you for your unwavering guidance and belief in my abilities. Your mentorship has been invaluable, and I am truly grateful for the opportunity to work with you.

Sincerely,
Dishant Patel - 202201260
Madhav Kanjilimadom - 202203018
Shravan Kakadiya - 202201333

DECLARATION

We, the members of Group 9, IT462 - Exploratory Data Analysis Autumn 2024, hereby declare that the EDA project work presented in this report is our original work and has not been submitted for any other academic degree. All the sources cited in this report have been appropriately referenced.

We acknowledge that the data used in this project is obtained from the data.gov.in site. We also declare that we have adhered to the terms and conditions mentioned in the website for using the dataset. We confirm that the dataset used in this project is true and accurate to the best of our knowledge.

We acknowledge that we have received no external help or assistance in conducting this project, except for the guidance provided by our mentor Prof. Gopinath Panda. We declare that there is no conflict of interest in conducting this EDA project.

We hereby sign the declaration statement and confirm the submission of this report on 2nd December 2024.

DR Patel

Dishant Patel
ID: 202201260
Course: BTech ICT

Madhav Kanjilimadom
ID: 202203018
Course: BTech MnC

Shravan Kakadiya
ID: 202201333
Course: BTech ICT

CERTIFICATE

This is to certify that Group 9 comprising Dishant Patel, Madhav Kanjilimadom and Shravan Kakadiya has successfully completed an exploratory data analysis (EDA) project on Analyzing Foreign Tourist Arrivals and Their Economic Impact on India, which was obtained from data.gov.in.

The EDA project presented by Group 9 is their original work and has been completed under the guidance of the course instructor, Prof. Gopinath Panda, who has provided support and guidance throughout the project. The project is based on a thorough analysis of multiple tourism-based datasets, and the results presented in the report are based on the data obtained from the datasets.

This certificate is issued to recognize the successful completion of the EDA project on Analyzing Foreign Tourist Arrivals and their Economic Impact on India, which demonstrates the analytical skills and knowledge of the students of Group 9 in the field of data analysis.

Signed,
Dr. Gopinath Panda,
IT 462 Course Instructor
Dhirubhai Ambani Institute of Information and Communication Technology
Gandhinagar, Gujarat, INDIA.

July 15, 2025

Contents

List of Figures

Abstract

Tourism plays a pivotal role in shaping India's economic landscape, with Foreign Tourist Arrivals (FTAs) serving as a critical indicator of its global appeal. This project focuses on analyzing continent-wise FTA data from the 21st century, uncovering regional trends and their contributions to India's tourism growth.

By performing exploratory data analysis (EDA) and forecasting FTAs for future years, we identify key growth regions and seasonal patterns. Additionally, we integrate insights from related datasets, including foreign exchange earnings (FEE), quarterly FTA data, age group demographics, and India's global tourism rank, to examine the broader economic impact of FTAs.

Our goal is to provide data-driven recommendations for optimizing India's tourism strategies, improving foreign exchange earnings, and enhancing the country's competitiveness in the global tourism sector.

Chapter 1. Introduction

1.1 Your Project idea

Foreign Tourist Arrivals (FTAs) significantly influence India's tourism-driven economy, contributing to foreign exchange earnings (FEE) and shaping the country's global tourism competitiveness. However, understanding and leveraging regional and demographic trends in FTAs remain a challenge. By integrating insights from related datasets such as quarterly FTA trends, age group distributions, and India's global tourism rank, the study seeks to provide reasonable and viable recommendations to optimize India's tourism strategies and enhance its contribution to the national economy.

1.2 Data Collection

We collected all our data from data.gov.in. In most of the datasets available on the website, most data was either incomplete or insufficient to make reasonable inferences. To counter this problem, we used multiple such datasets which combined together helped us perform a comprehensive analysis on this data.

An example is to analyse the growth of Foreign Exchange Earnings (FEE) due to to tourism, the data was in parts (2001-2019) and (2020-2023). Quarterly data for FTAs was also in a different dataset. We had to combine these filed for a meaningful analysis.

1.3 Dataset Description

The dataset 'FTA_2001to2003' provides insights into the yearly distribution of foreign tourist arrivals (FTA) across various nationalities and regions. It includes information organized by multiple factors, including categories, geographic distributions, and detailed yearly metrics. Each year is represented as a separate column, showing the number of foreign tourist arrivals for that specific year. The dataset captures trends over 23 years, enabling analysis of tourism patterns.

The dataset 'FTA_FEE_data' provides a comprehensive overview of foreign tourist arrivals (FTAs) in India from 2001 to 2023, along with insights into associated revenue, demographic distribution, and seasonal trends.

It captures the total FTAs annually, along with the percentage change over the previous year, and includes data on foreign exchange earnings (FEE) from tourism and their yearly growth rate. The dataset also offers a demographic breakdown of tourists by age groups (e.g., 0-14, 15-24, 25-34, etc.), as well as their distribution across the four quarters of the year. Additionally, it highlights tourism's contribution to India's economy as a percentage and tracks India's global rank in tourism over the years. This dataset is ideal for analyzing trends in tourism growth, demographic patterns, seasonal preferences, and the economic impact of tourism in India. Each column represents a key contributing factor to the amount of FTAs in India. This is the base of our entire analysis of the economic growth in India due to tourism.

1.4 Packages required

For a complete and comprehensive data analysis of any dataset, one will have to make use of multiple libraries of **python**. The libraries used in our exploratory analysis are:

- **NumPy Library**: Used for efficient numerical computations, including operations on arrays and matrices.
- **Pandas Library**: Essential for data reading and manipulation, offering tools like DataFrames for handling tabular data.
- **PyPlot from MatPlotLib Library**: Provides functionalities for creating static, animated, and interactive visualizations such as line graphs, bar charts, and scatter plots.
- **Seaborn Library**: A data visualization library based on Matplotlib, used for creating informative and attractive statistical graphics like heatmaps, boxplots, and violin plots.
- **Missingno Library**: Specialized in visualizing missing data patterns in datasets, helping identify and handle missing values effectively.
- **Plotly.express from Plotly Library**: A high-level interface for creating interactive visualizations like scatter plots, line graphs, and choropleth maps, with ease and customization.

  For the feature engineering component of our project, we used the **sklearn** library which facilitated a seamless feature selection and model training. The classes from the sklearn library used here are as follows;

  – `sklearn.model_selection`: Provides functions like `train_test_split` for splitting datasets into training and testing sets.

  – `sklearn.preprocessing`: Contains tools for data preprocessing, such as scaling features (`StandardScaler`) or generating polynomial features (`PolynomialFeatures`).

  – `sklearn.linear_model`: This module includes various linear models for regression and classification. Examples include:

    * `LinearRegression`: A class for performing simple or multiple linear regression.

  – `sklearn.ensemble`: This module implements ensemble methods, which combine multiple models for better predictive performance. Examples include:

    * `RandomForestRegressor`: A class for regression using random forest.
    * `GradientBoostingRegressor`: A class for gradient boosting regression.

  – `sklearn.metrics`: This module provides evaluation metrics for models. Examples include:

- * mean_squared_error and mean_absolute_error: Functions to measure error between predicted and actual values.
- * r2_score: A function to measure the goodness-of-fit of regression models.

```python
import numpy as np
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import missingno as msno
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor,
    GradientBoostingRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error,
    r2_score
```

Listing 1.1: Importing required libraries

Note that we have not installed any libraries as all libraries used in our analysis were already pre-installed on our devices. Hence, we have only imported them.

Chapter 2. Loading and Preprocessing of Data

After collecting our data, the next step is to load our data. Since, we are using the combination multiple data sets, we will have to individually observe each one's characteristics and merge them accordingly using `pd.merge`.

First, we upload our data into a repository (we have used github for easy access) and then read our data using `pd.read_csv(df)`. We have then observed a random sample of 5 rows in our data.

```
df1 = pd.read_csv('https://raw.githubusercontent.com/Shravan-0024/EDA_labAssignments/refs/heads/main/syb-18-chapter_26_tourism_table_26.1%20(1).csv')
df1.sample(5)

df2 = pd.read_csv('https://raw.githubusercontent.com/Shravan-0024/EDA_labAssignments/refs/heads/main/India-Tourism-Statistics-2022-Table-2.1.4.csv')
df2.sample(5)
```

Listing 2.1: Reading our Data

Here, df1 contains data up to 2016 and df2 contains data up until 2021 along with some other columns which we will drop during data cleaning. We now merge our data according to the correct columns.

```
result_df = pd.merge(
    df1,
    df2,
    left_on=['Category', 'Nationality'],
    right_on=['Region', 'Country of Nationality'],
    how='left'
)
result_df.head()
```

Listing 2.2: Merging the Datasets

Chapter 3. Data Cleaning

After collecting our data, the next step is to understand and clean our data. After observing a sample of our data in the previous section, we note that we will not be using some columns present in our dataset, specifically the percentage share columns. Hence, we will drop these columns and any duplicate columns entirely using `df.drop`. We will rename our columns more aptly so that it is easier for us or any other user to understand our data. We have dropped row 2 as it contains only null values and row 86 as the country is not mentioned for this row. Note that dropping these rows will not affect our analysis.

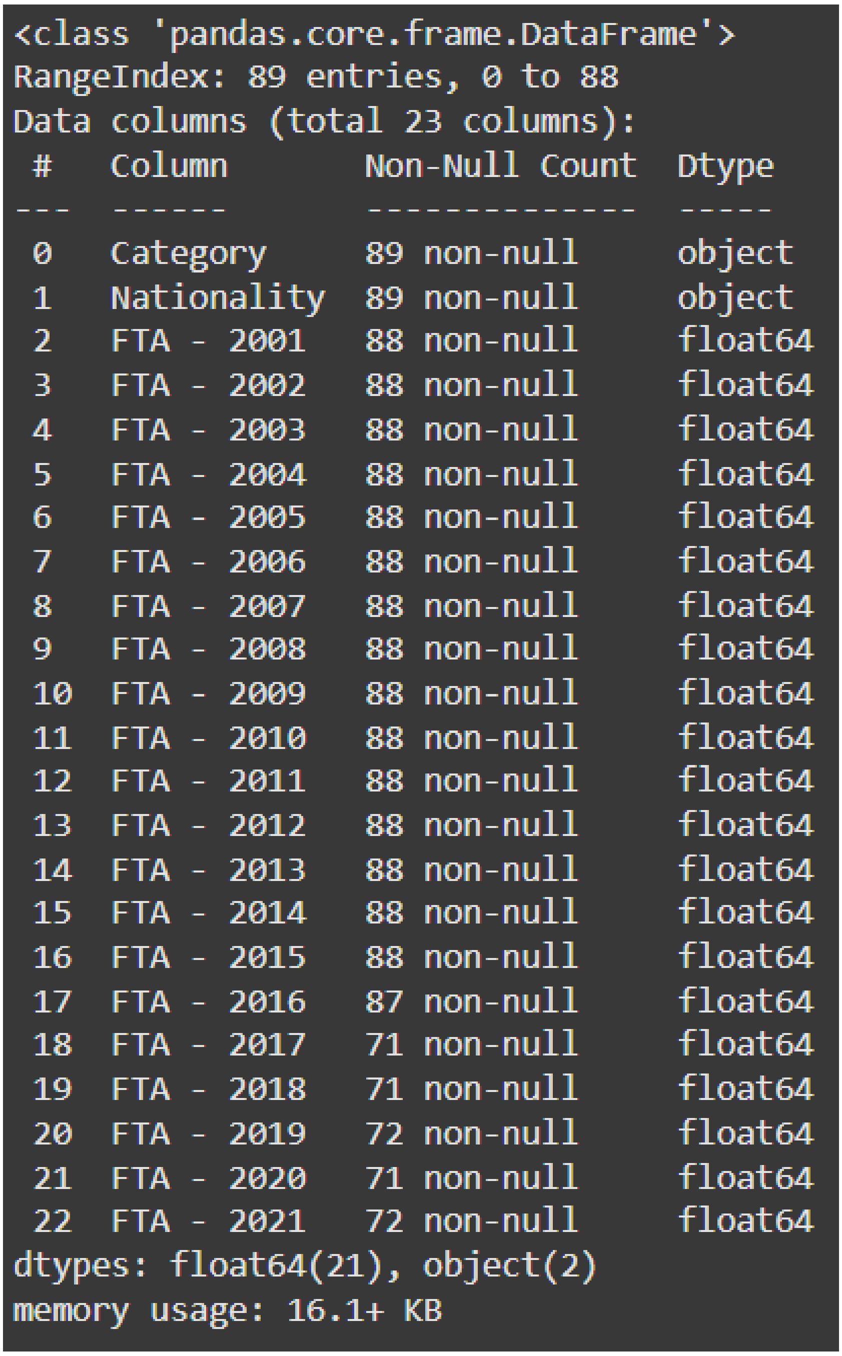
```python
columns_to_drop = [
    "Percentage Share  - 2018/17",
    "Percentage Share  - 2019/18",
    "Percentage Share  - 2020/19",
    "Percentage Change - 2021/20",
    "Percentage Change - 2021"
]
df2 = df2.drop(columns=columns_to_drop)

result_df.rename(columns={'2001': 'FTA - 2001', '2002': 'FTA - 2002', '2003': 'FTA - 2003', '2004': 'FTA - 2004', '2005': 'FTA - 2005', '2006': 'FTA - 2006', '2007': 'FTA - 2007', '2008': 'FTA - 2008', '2009': 'FTA - 2009', '2010': 'FTA - 2010'}, inplace=True)
result_df.rename(columns={'2011': 'FTA - 2011', '2012': 'FTA - 2012', '2013': 'FTA - 2013', '2014': 'FTA - 2014', '2015': 'FTA - 2015', '2016': 'FTA - 2016'}, inplace=True)
result_df.rename(columns={'Number of Arrivals-2017': 'FTA - 2017', 'Number of Arrivals-2018': 'FTA - 2018', 'Number of Arrivals-2019': 'FTA - 2019', 'Number of Arrivals-2020': 'FTA - 2020', 'Number of Arrivals-2021': 'FTA - 2021'}, inplace=True)

df.drop(index = 2, inplace=True)
df.drop(index = 86, inplace=True)
```

Listing 3.1: Reading our Data

For further analysis of our data such ac missing data analysis, visualisation, etc., we need to have an apt understanding of our dataset. This includes identifying what exactly is the data, its types, size, etc. To get the metadata of our dataset, we use `df.info()`. It provides essential details about the dataset columns, including the column names, the count of non-null values in each column, and the data type of each column.

```python
df.info()
```

Listing 3.2: Reading our Data



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 89 entries, 0 to 88
Data columns (total 23 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Category     89 non-null     object 
 1   Nationality  89 non-null     object 
 2   FTA - 2001   88 non-null     float64
 3   FTA - 2002   88 non-null     float64
 4   FTA - 2003   88 non-null     float64
 5   FTA - 2004   88 non-null     float64
 6   FTA - 2005   88 non-null     float64
 7   FTA - 2006   88 non-null     float64
 8   FTA - 2007   88 non-null     float64
 9   FTA - 2008   88 non-null     float64
 10  FTA - 2009   88 non-null     float64
 11  FTA - 2010   88 non-null     float64
 12  FTA - 2011   88 non-null     float64
 13  FTA - 2012   88 non-null     float64
 14  FTA - 2013   88 non-null     float64
 15  FTA - 2014   88 non-null     float64
 16  FTA - 2015   88 non-null     float64
 17  FTA - 2016   87 non-null     float64
 18  FTA - 2017   71 non-null     float64
 19  FTA - 2018   71 non-null     float64
 20  FTA - 2019   72 non-null     float64
 21  FTA - 2020   71 non-null     float64
 22  FTA - 2021   72 non-null     float64
dtypes: float64(21), object(2)
memory usage: 16.1+ KB
```

Figure 3.1: Output given by df.info()

Our dataset contains data as follows: Object (string) data types:

1. Category
2. Nationality

Float data types:

1. FTA - 2001
2. FTA - 2002
3. FTA - 2003
4. FTA - 2004
5. FTA - 2005
6. FTA - 2006
7. FTA - 2007
8. FTA - 2008
9. FTA - 2009
10. FTA - 2010
11. FTA - 2011
12. FTA - 2012
13. FTA - 2013
14. FTA - 2014
15. FTA - 2015
16. FTA - 2016
17. FTA - 2017
18. FTA - 2018
19. FTA - 2019
20. FTA - 2020
21. FTA - 2021

The `df.describe()` method generates descriptive statistics for the numeric columns in the dataset. These statistics are useful for displaying measures of central tendency (mean, median, or mode), measures of dispersion (variance, standard deviation, minimum, and maximum), and the overall shape of the distribution for numeric columns. We have not included a graphic of this function in our code as the number of numerical columns is too large. It can be directly viewed on the Google Colab file.

3.1 Missing data analysis

Missing data is a common issue in datasets and can significantly affect the quality of the analysis. Before any data processing, it is essential to determine both the quantity and proportion of missing data within the dataset.

We start our missing data analysis with identifying how much data is missing in our dataset. This can be done using `df.isnull().sum()`. This gives us a column-wise result of how many values are missing in each column. We have 16 missing values in each of the FTA columns of 2017-2021.

Another way of observing this visually is by using the `missingno` library in python. Ideally we want to identify what, where, and why the data is missing. The missingno library helps in that.

We used different plots from the missingno library to observe this. The figure 3.3 depicts the visual representation of how many values are missing in each column using the barplot from the missingno library. Fig 3.4 depicts the Matrix plot of our data. The blank cells in the matrix represent the missing data for each column. The key difference here is that the matrix plot tells us where exactly our data is missing. Here, we can see that whenever data is missing in the 2017 column, data is missing from the subsequent columns too. This positive correlation can be further confirmed by the heatmap from the missingno library. Fig 3.5 shows shows correlation between missingness of data. Correlation of +1 shows that whenever data is missing in one column, the same value is missing in the other column as well.

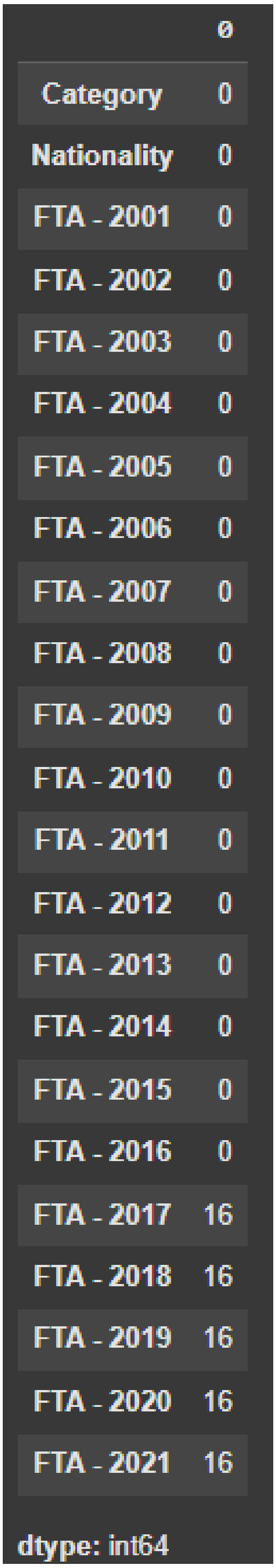
3.1.1 Types of Missingness present in our dataset

The next step in missing data analysis is to identify why our data is missing. There are three types of missing data. Missing completely at random, missing at random and missing not at random.

The three types are as follows:

- Missing Completely at Random (MCAR):
  - The probability of value missing is independent of the values of dataset.
  - In other words, there is no particular reason for the missing values.
- Missing at Random (MAR)
  - the probability of being missing is the same only within groups defined by the observed data.
  - MAR occurs when the missingness is not random, but where missingness can be fully accounted for by variables where there is complete information.
- Missing Not at Random (MNAR)
  - The probability of being missing varies for reasons that are unknown to us.
  - Missingness depends on unobserved data or the value of the missing data itself.

The missing data in our dataset, if we observe, is from our second dataset, which we merged with our initial dataset. So for some countries, the FTA data isn't available from 2017 onwards. The website has not mentioned any reason for this. This might be MCAR data. We can confirm this using little's



```
              0
Category      0
Nationality   0
FTA - 2001    0
FTA - 2002    0
FTA - 2003    0
FTA - 2004    0
FTA - 2005    0
FTA - 2006    0
FTA - 2007    0
FTA - 2008    0
FTA - 2009    0
FTA - 2010    0
FTA - 2011    0
FTA - 2012    0
FTA - 2013    0
FTA - 2014    0
FTA - 2015    0
FTA - 2016    0
FTA - 2017   16
FTA - 2018   16
FTA - 2019   16
FTA - 2020   16
FTA - 2021   16

dtype: int64
```

Figure 3.2: No. of missing values column-wise

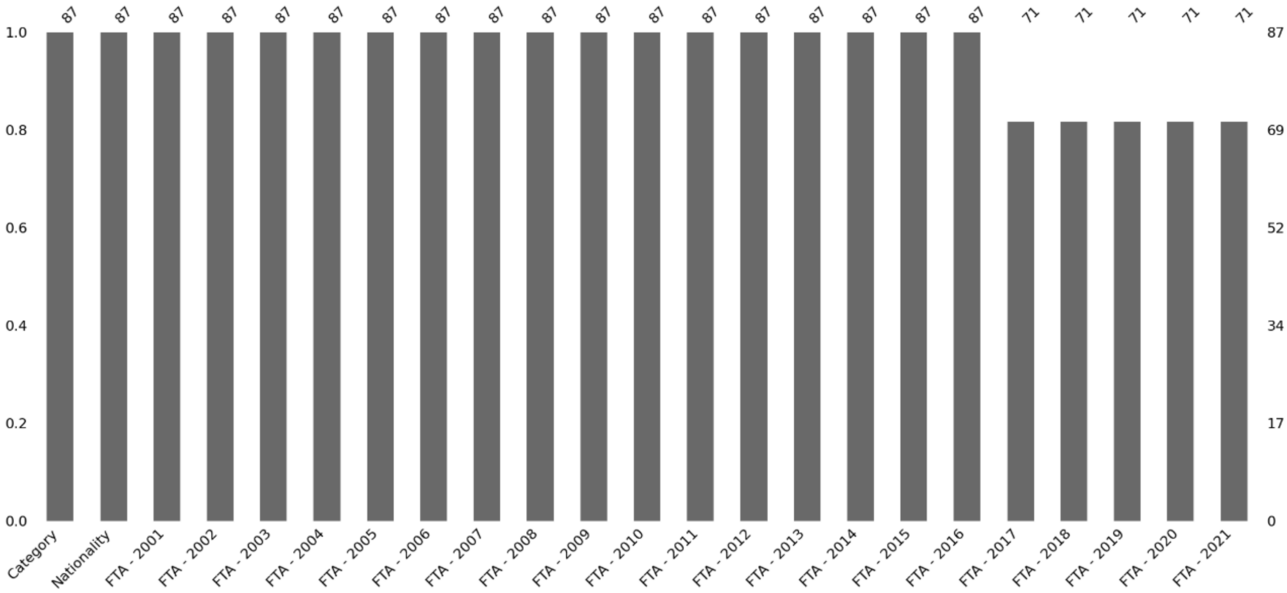


Figure 3.3: Barplot for visualising missing data

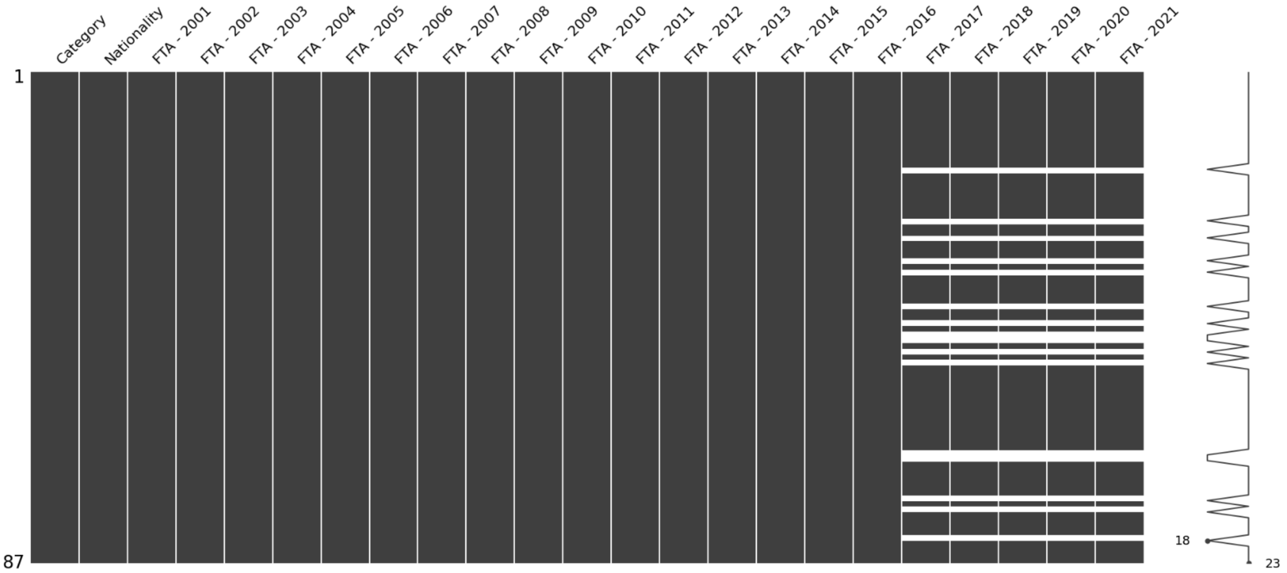


Figure 3.4: Matrix Plot for Missing Data

MCAR test.

3.1.2 Little's MCAR Test

It is a statistical test used to assess whether the missing data in a dataset are missing completely at random or if there is a systematic pattern to the missingness. Null Hypothesis (H0): The missing data are completely at random. Alternative Hypothesis (H1): The missing data are not completely at random; there is some systematic pattern or relationship with the observed data. The test is based on a chi-squared statistic, The p-value associated with this statistic is used to assess the significance of the

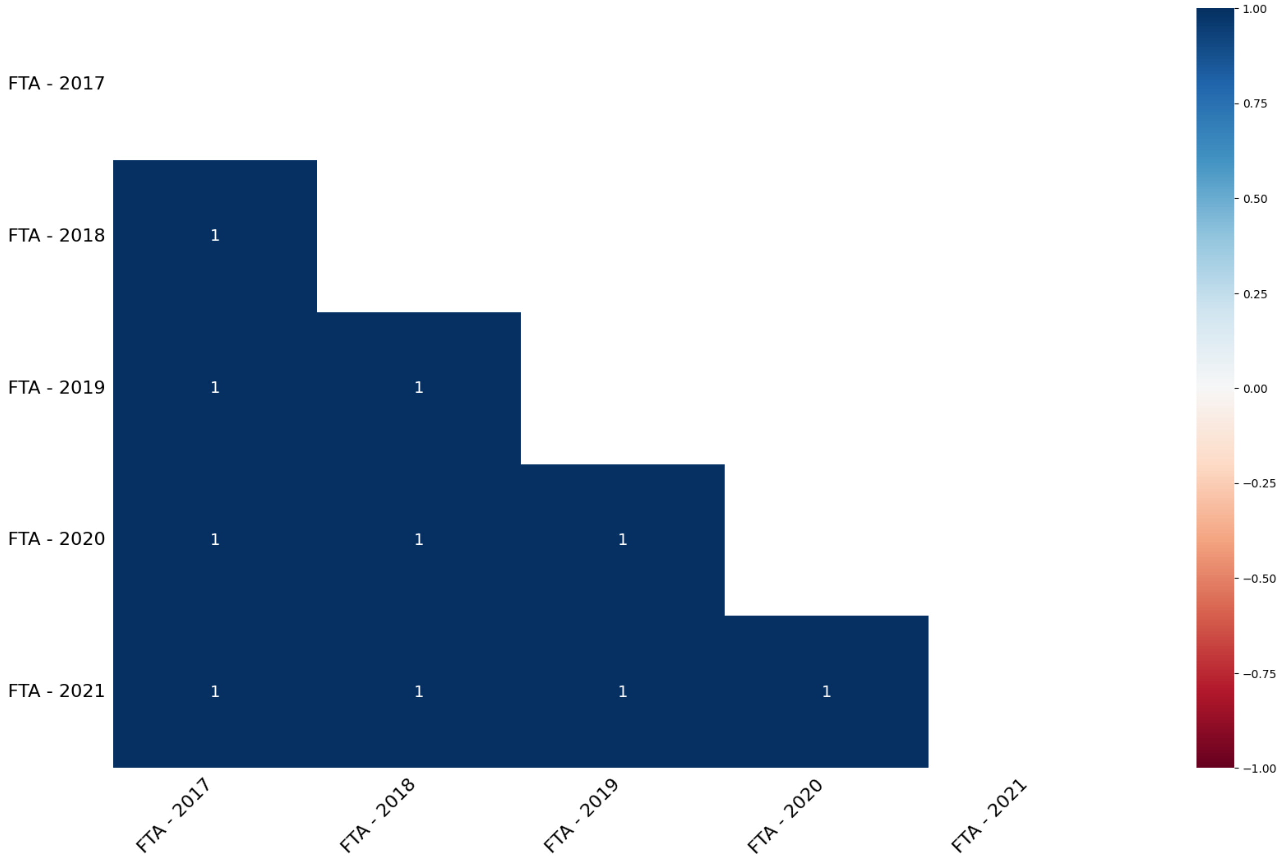


Figure 3.5: Correlation Heatmap between missingness of data.

test. If the p-value is below a chosen significance level , say 0.05, you may reject the null hypothesis and conclude that the missing data are not MCAR. Keep in mind that the MCAR test has limitations, and it might not be sensitive to certain types of non-random missingness. If data are not missing completely at random, other imputation methods or adjustments to the analysis might be necessary.

```python
from scipy.stats import chi2_contingency

# Creating a missingness indicator (1 for missing, 0 for not missing)
missing_indicator = df.iloc[:, 2:].isnull().astype(int)

# Now let's create the observed data, which will be the data where values
    are not missing
observed_data = df.iloc[:, 2:].notnull().astype(int)

# Creating a contingency table (observed vs. missing data)
contingency_table = pd.concat([observed_data.sum(axis=1), missing_indicator.
    sum(axis=1)], axis=1)
contingency_table.columns = ['Observed', 'Missing']


# Checking for rows with zero observed or missing values and remove them
contingency_table = contingency_table[
    (contingency_table['Observed'] != 0) & (contingency_table['Missing'] !=
    0)
]

# Performing Chi-Square Test for MCAR hypothesis if contingency table is not
    empty
if not contingency_table.empty:
    chi2, p, _, _ = chi2_contingency(contingency_table)

    # Check the result
    print("Chi-Square Test Result:")
    print(f"Chi-Square Value: {chi2}")
    print(f"P-Value: {p}")

    # If p-value is greater than 0.05, we fail to reject the null hypothesis
    (MCAR)
    if p > 0.05:
        print("The missing data is likely MCAR.")
    else:
        print("The missing data is not MCAR.")
else:
    print("Contingency table is empty after removing rows with zero values.
    Cannot perform Chi-Square test.")
```

Listing 3.3: Code for Little's MCAR Test

Here, we got our p-value as 1. Hence, our missing data is of MCAR type. Now we move on to handling missing values. There are many methods of handling missing values, such as deleting, filling, imputing, etc. We will not be deleting any data such as rows or columns as each row and column plays an integral contribution to the overall FTA count per year.

3.2 Imputation

Imputation is a technique that involves replacing missing data with estimated values based on the observed data. There are several methods for imputation, such as Linear, Mean, Mode, Hot-deck, KNN imputation etc.

We have imputed our values by using the factor of total FTAs from the concerned year, which is available in our second dataset, and then proportionally assigned the FTA details for each country accordingly. This in a way, can be interpreted as mean imputation. This data will be used in the future to predict FTAs for future years. This gives us an accurate idea of how many tourists would have come to India in that particular year taking into account the previous 5 years.

We did not use any other method of imputation such as filling or interpolation as it can cause discrepancies as our data not only depends on the previous year, but also on the other total proportion of tourist arrivals from other countries to preserve the correctness of our data.

```python
# Total FTA values for 2022 and 2023
total_fta = {
    2022: 6440000,
    2023: 9240000
}
# Start by calculating the proportions for 2021
df['Proportion_2021'] = df['FTA - 2021'] / df['FTA - 2021'].sum()

# Generate FTA values year by year, updating proportions dynamically
previous_year = 2021
for year, total in total_fta.items():
    # Calculate proportions based on the previous year's data
    proportion_column = f'Proportion_{previous_year}'
    fta_column_previous = f'FTA - {previous_year}'

    # Add a new column for the current year's FTA
    df[f'FTA - {year}'] = (df[fta_column_previous] / df[fta_column_previous].sum()) * total

    # Update the proportions for the current year
    df[f'Proportion_{year}'] = df[f'FTA - {year}'] / df[f'FTA - {year}'].sum()
    # Update the previous year
    previous_year = year

# Display the updated DataFrame for first 5 rows...
columns_to_display = ['Category', 'Nationality'] + [f'FTA - {year}' for year in range(2016, 2024)]
```

Listing 3.4: Imputing Approximate values for 2022 and 2023

Chapter 4. Outlier Analysis

Outliers are observations that are significantly different from other data points. They are rare, distinct and exceptionally far away from the mainstream of data. There can be many reasons for the presence of outliers in our data.

- Sometimes the outlier maybe genuine (arising due to extreme circumstances).
- Sometimes they maybe due to incorrect data entry.

Outliers can be identified in boxplots using the Inter-quartile range. In our first dataset containing country-wise FTAs, the outliers are present in each year almost always for the same country. This is not a case of incorrect data as these extreme values belongs to countries with a high number of FTAs to India compared to other countries. These values maybe high due to those countries having easy access (no visa requirement) or cheap travel, etc.

Hence, **we will not be removing or capping the outliers** in our country-wise dataset.

In our dataset involving FEE and other demographic data, our key observations are outliers present in the columns which show the change in no. of FTAs and change in the foreign exchange earnings through tourism. The outliers are present in the years 2020 and 2021 mainly. This is due to the COVID-19 pandemic that broke out at the start of 2020. Hence, we observed a drastic reduction in the FTAs in those years. These outliers were due to extreme and unforeseen circumstances of the pandemic. However this data is accurate and is critical to predicting and analysing FTA and FEE trends for future years.

Hence, **we will not be removing or capping the outliers** in our FTA-FEE demographic dataset.

Below are the boxplots for the different data in our data.

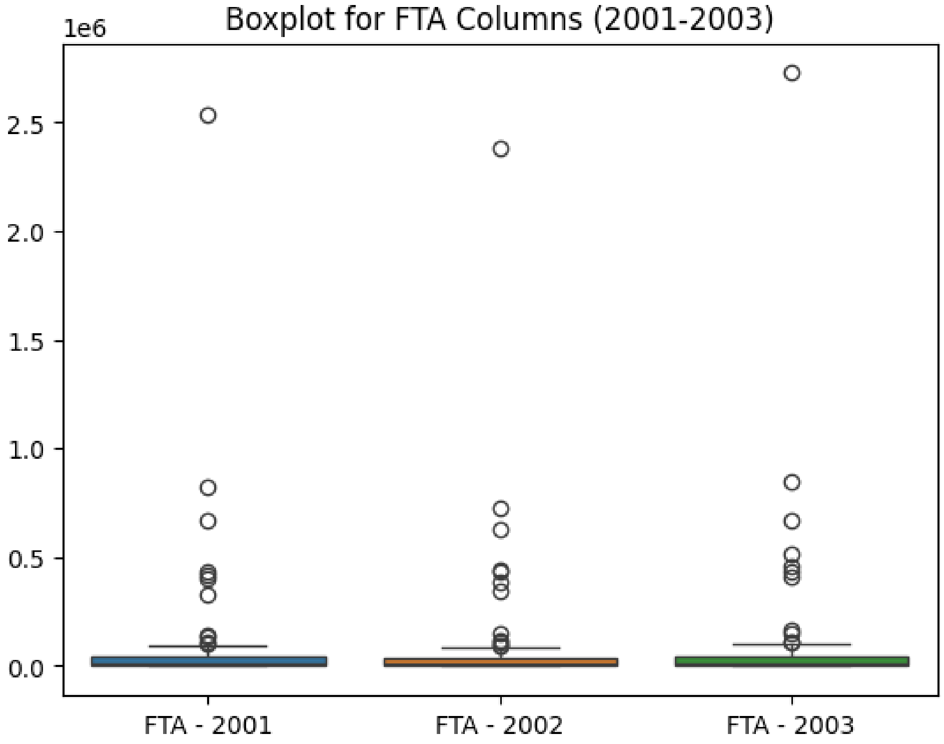


Figure 4.1: FTAs Outliers in 2001-03

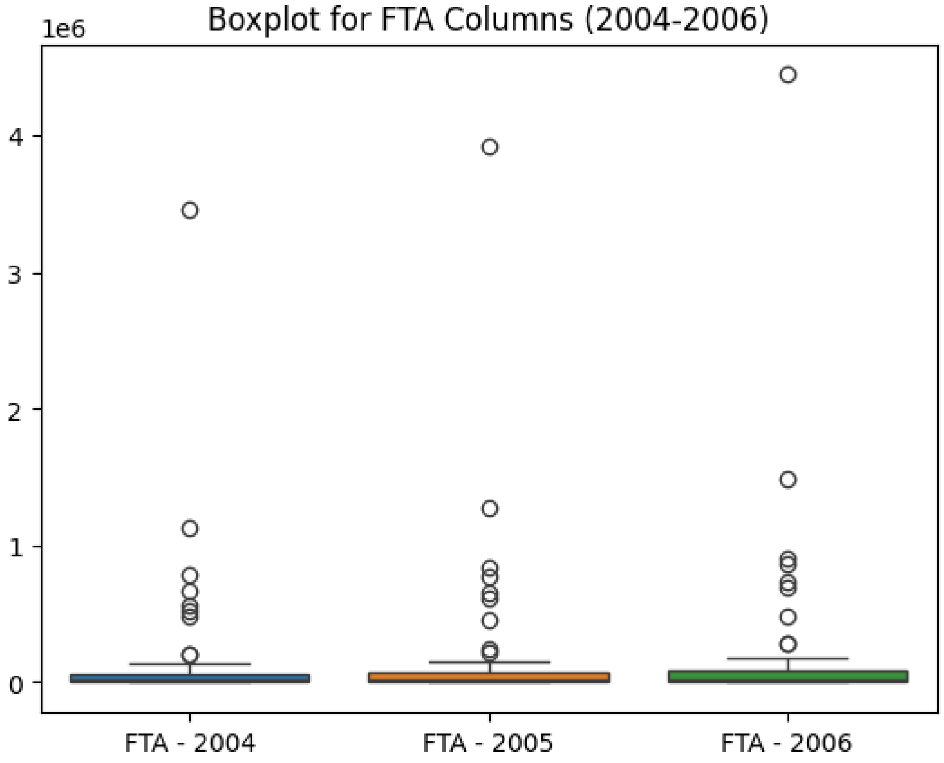


Figure 4.2: FTAs Outliers in 2004-06

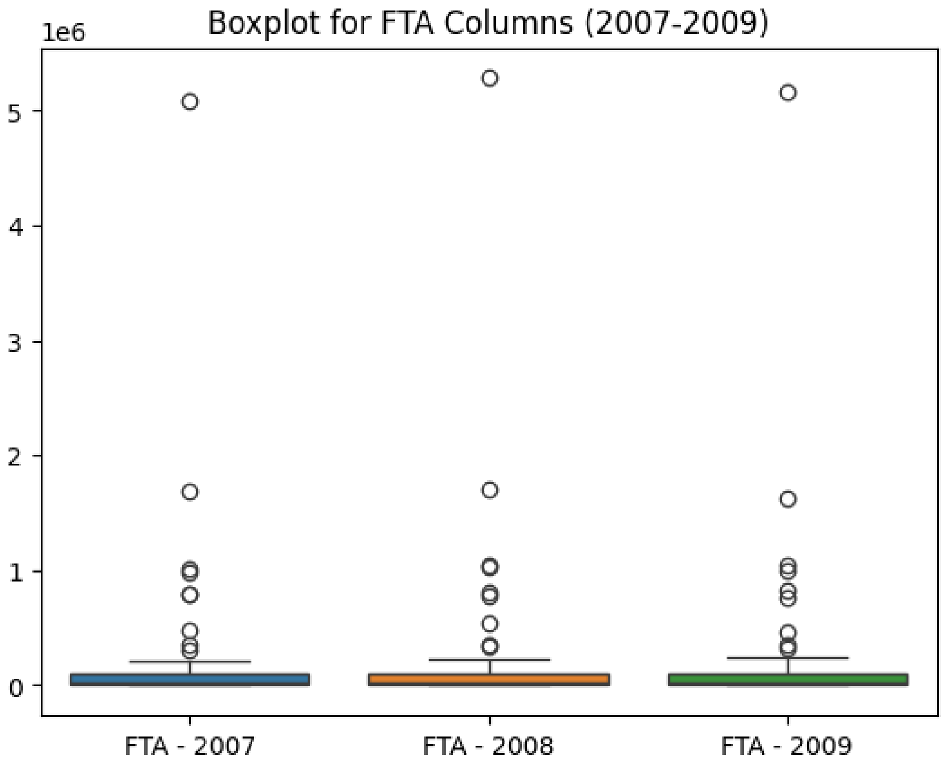


Figure 4.3: FTAs Outliers in 2007-09

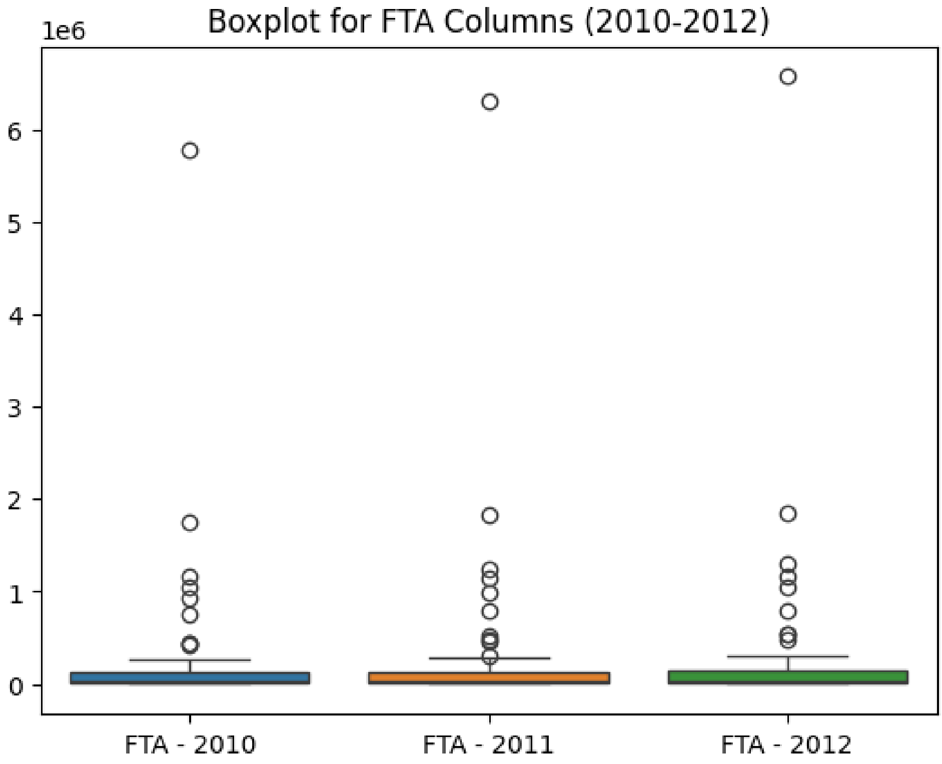


Figure 4.4: FTAs Outliers in 2010-12

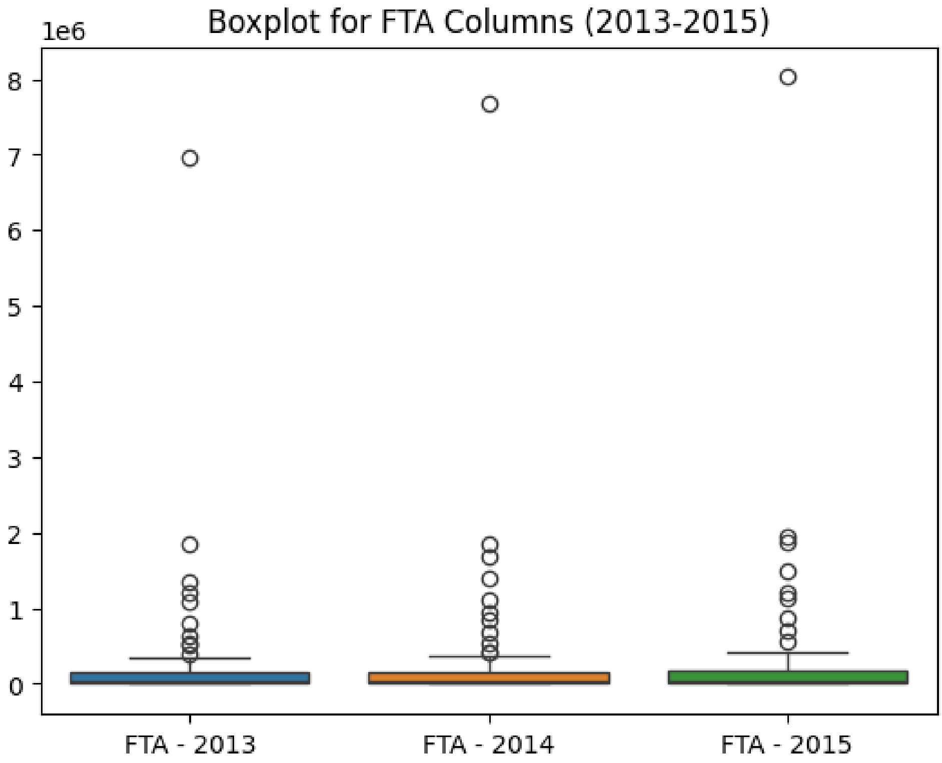


Figure 4.5: FTAs Outliers in 2013-15

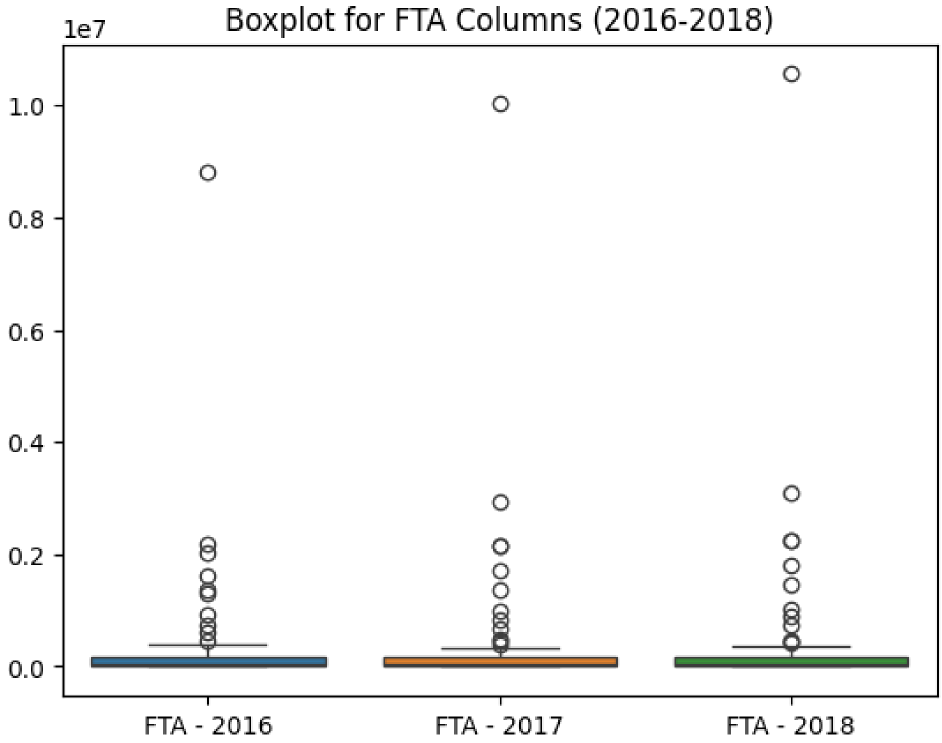


Figure 4.6: FTAs Outliers in 2016-18

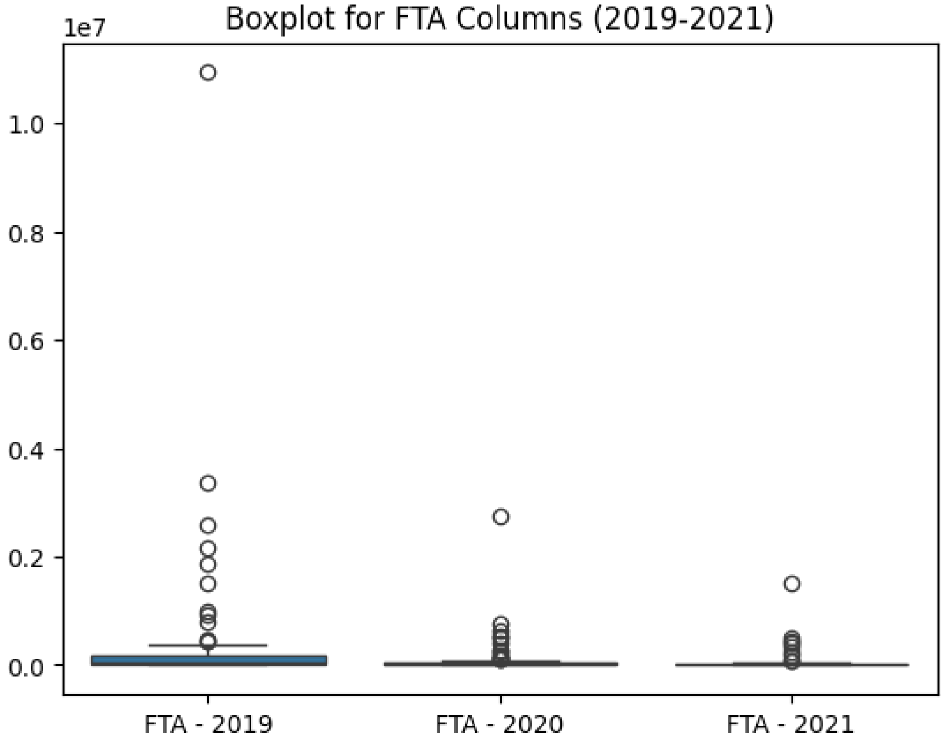


Figure 4.7: FTAs Outliers in 2019-21

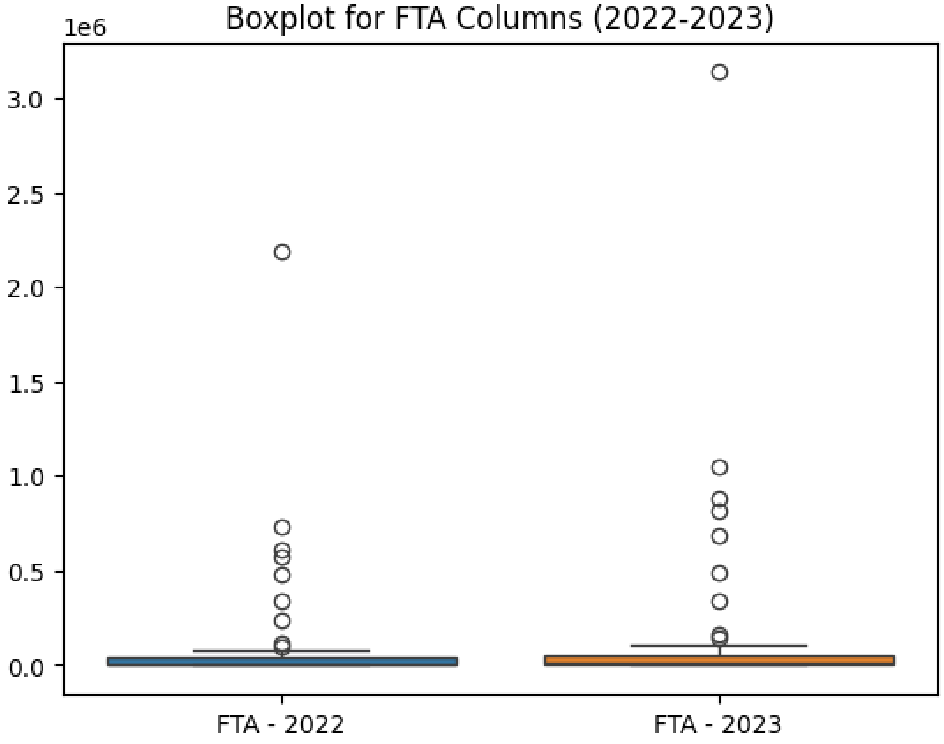


Figure 4.8: FTAs Outliers in 2022-23

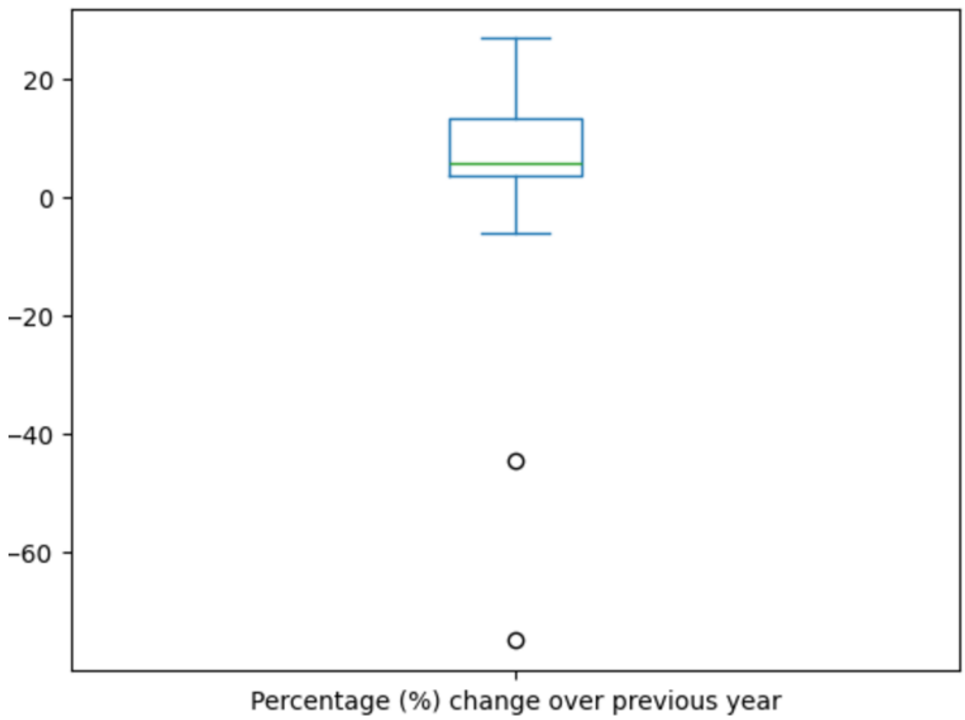


Figure 4.9: Outliers in FTAs

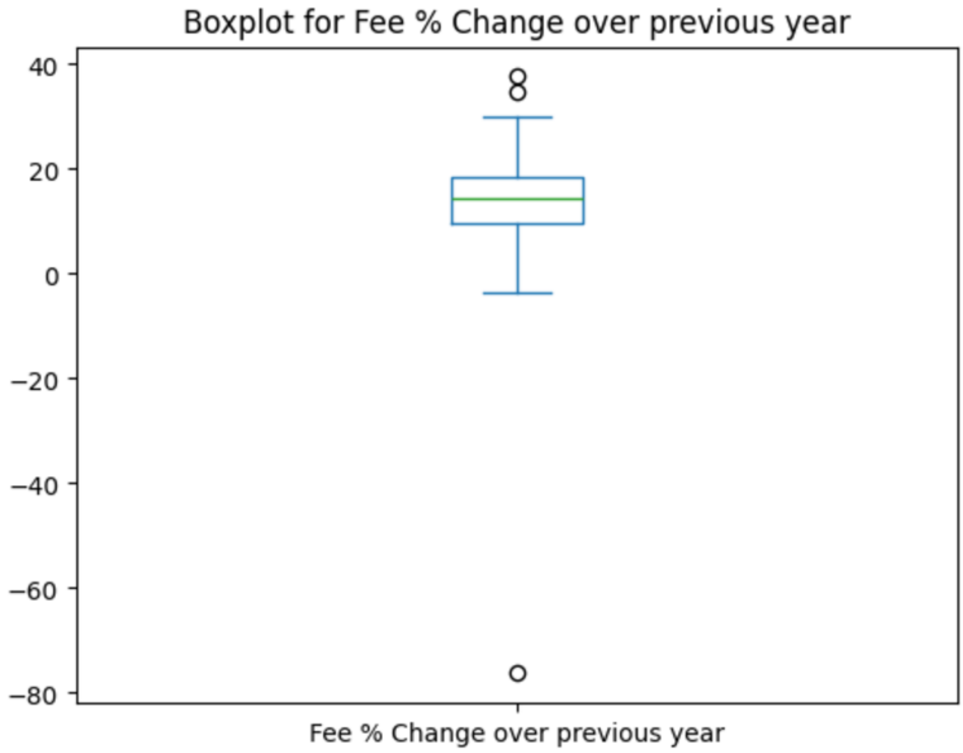


Figure 4.10: Outliers in FEEs

Chapter 5. Visualization

5.1 Univariate analysis

Univariate analysis involves analysing each variable independently in the dataset. The goal of univariate analysis is to understand the distribution of each variable and examine their statistical properties to make meaningful inferences. Some techniques and plots involved in univariate analysis are:

- Histogram
- Box Plot
- Count Plot
- Bar Plot
- Pie Chart
- Stem Leaf
- Violin Plots

- Fig 5.1 gives us the distribution of Regions from where the foreign tourists travel to India.
- Fig 5.2 gives us the distribution of FTAs by age group. The major contributors by age to FTAs in India are middle-aged adults (ages 25-34, 35,-44, and 45-54).
- Fig 5.3 gives us the distribution of FTAs in the 21st century quarter-wise.
- This bar plot gives us the distribution of different regions and how much they contribute to our data set. The most common tourists being from western Europe followed by west asia.
- Fig 5.5 illustrates a steady rise in the number of tourists visiting India up until 2020, when COVID-19 restrictions dealt a severe blow to the tourism sector. The pandemic not only disrupted the sector's growth trajectory but also caused a significant economic downturn, from which recovery will take time. Now we review Fig 5.6

  1. **Q1 (January - March):**
     This quarter witnesses a significant influx of tourists to India. A few plausible reasons include India's cool and pleasant weather during these months, political or cultural events attracting international attention, or winter vacations in tourists' home countries that align with their travel plans.

2. **Q2 (April - June):**
   During this period, the number of tourists visiting India is noticeably lower. The scorching summer heat in many parts of India may act as a deterrent for travelers. Additionally, the lack of holidays for working individuals and school-going children in many countries could further contribute to the decline.
3. **Q3 (July - September):**
   Tourist arrivals during this quarter are moderate, reflecting a balanced inflow. The monsoon season, with its lush greenery and rejuvenating rains, could appeal to certain visitors. However, the absence of significant holidays in this timeframe might prevent a larger influx of tourists.
4. **Q4 (October - December):**
   This quarter records the highest number of tourist visits to India. Factors such as the festive season in India, along with year-end holidays in many countries, make it an attractive time for travelers to explore the country.

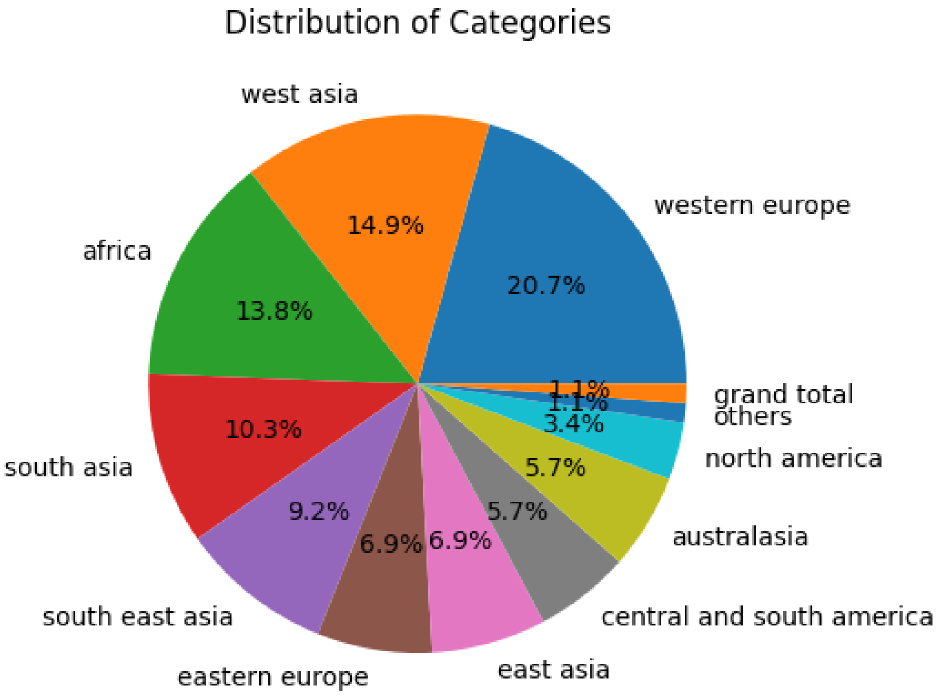


Figure 5.1: Distribuiton of Regions of Origins for FTAs

5.2 Multivariate analysis

Multivariate analysis is used to study and analyze multiple variables simultaneously. It helps to understand how variables interact, identify patterns, relationships, and dependencies, and make predictions by considering the combined influence of several factors. In our analysis, we used multiple plots, from the `seaborn and matplotlib` library to identify such relationships, if any.

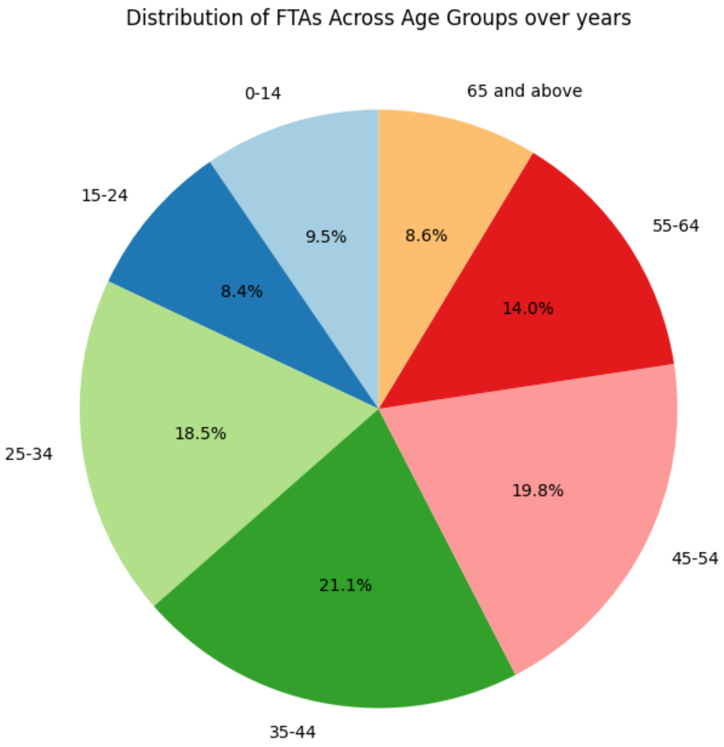


Figure 5.2: Distribution of FTAs by age groups

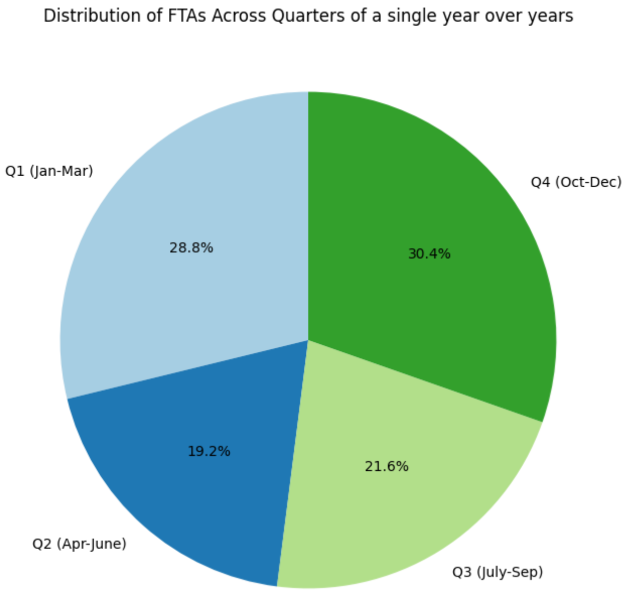


Figure 5.3: Quarterly Distribution of 21st century FTAs

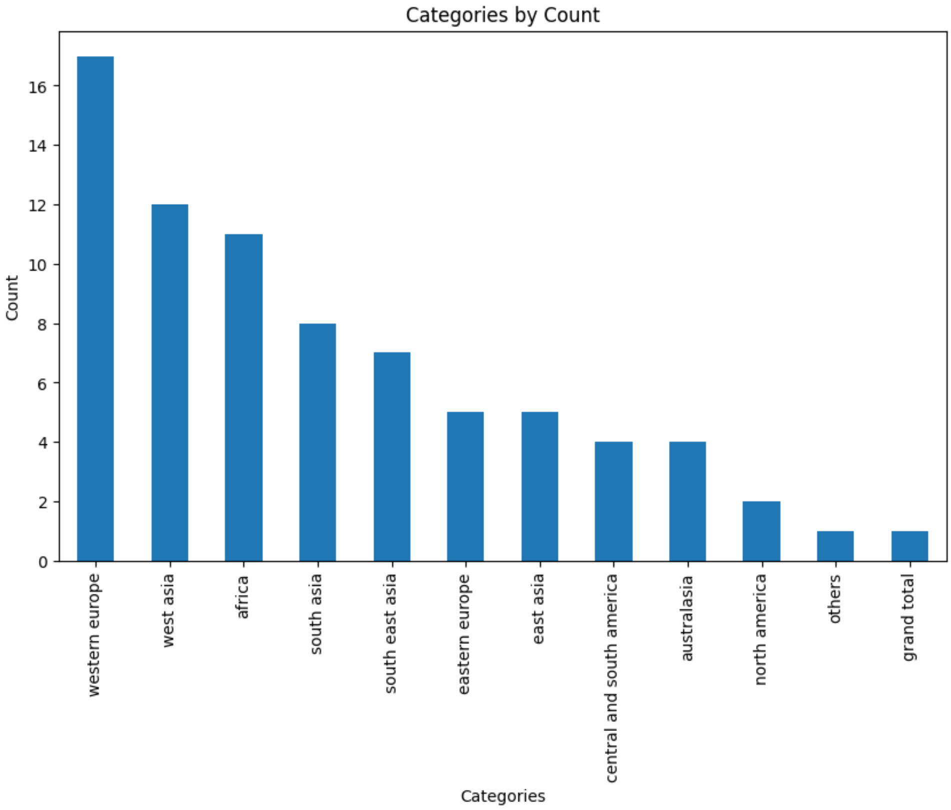


Figure 5.4: Regions by count

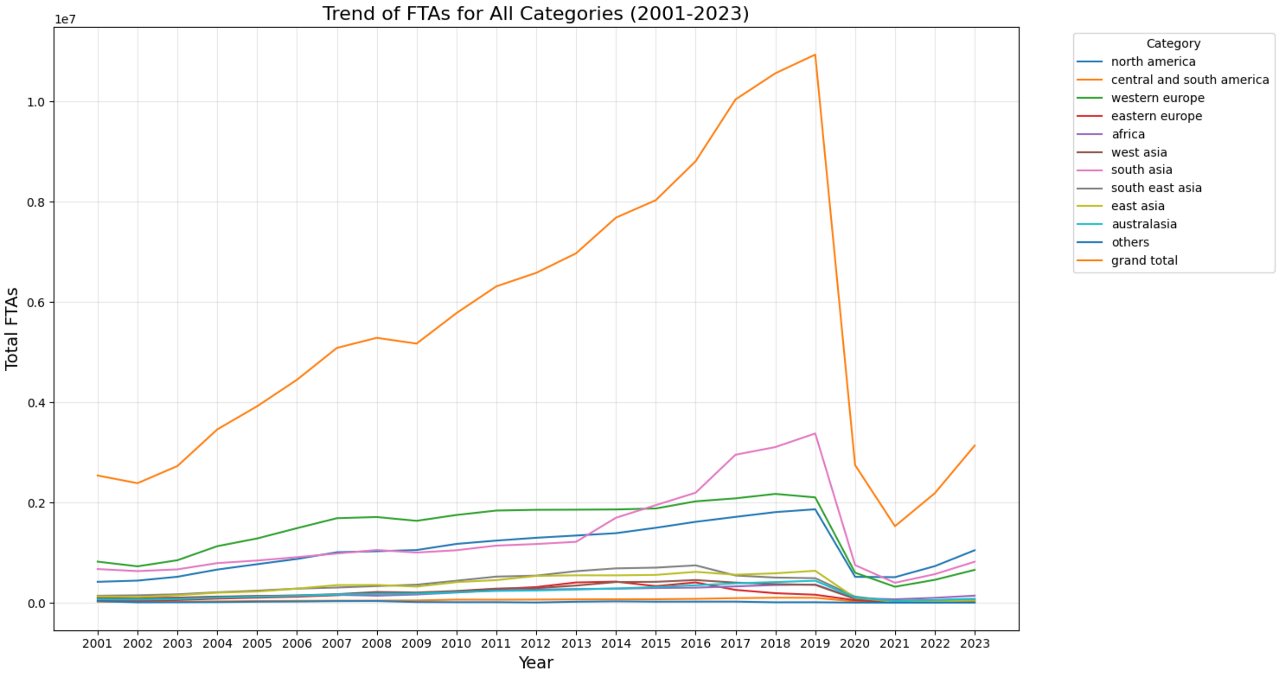


Figure 5.5: FTA Trends by region

For Bi-variate analysis, we have used seaborn library where we can analyse how the FTA columns of different years are related with each other and whether have any correlation or not. Generally we use one column as our Target feature for which we have to predict values so here as our target feature is FTA values so we have used here only FTA values only.

The figures along with their description and inference are given below.

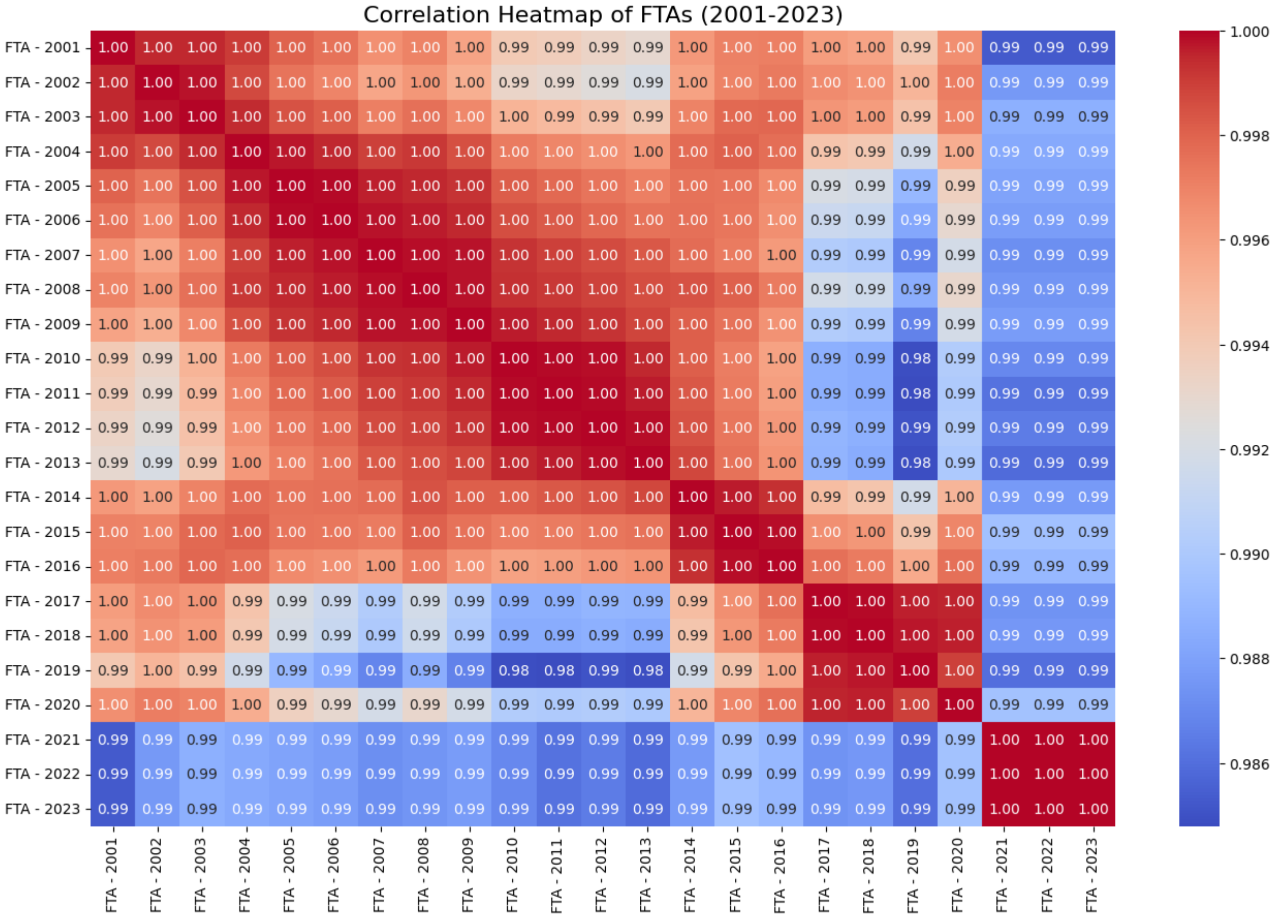
- Fig 5.7 Shows the correlation heatmap of the FTA data in India. The reason for the correlation matrix having most values close to one is because of our data being time-series data. Time-series data often exhibits a common trend, such as growth or decline over time. For example, if all variables (e.g., FTA values from different years) show an increasing trend, they will naturally have high positive correlations. For our dataset with columns like FTA-2001 to FTA-2023, the FTA values from one year are likely influenced by trends or factors that persist across years, such as economic growth or tourism policies. This shared dependency results in high correlations between these columns.

- Fig 5.8 shows the relation between FTAs in India and its share of world tourism. As FTAs increase, India takes up more of the world's tourism market.

- Fig 5.9 shows the relation between FTAs in India and India's rank in world tourism in that year. From this, we can infer that as FTAs increase, India achieves a higher rank in world tourism.

- Fig 5.10 Shows the scatter plot between FEE due to tourism vs India's share of the market in the tourism industry. We observe that as FEE increases, India achieves a higher share in the tourism market.

- Fig 5.11 shows the relation between FEE due to tourism and India's Rank in world tourism. It can be interpreted as when FEE increases, India achieves a better rank in world tourism.



	FTA - 2001	FTA - 2002	FTA - 2003	FTA - 2004	FTA - 2005	FTA - 2006	FTA - 2007	FTA - 2008	FTA - 2009	FTA - 2010	FTA - 2011	FTA - 2012	FTA - 2013	FTA - 2014	FTA - 2015	FTA - 2016	FTA - 2017	FTA - 2018	FTA - 2019	FTA - 2020	FTA - 2021	FTA - 2022	FTA - 2023
FTA - 2001	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	0.99
FTA - 2002	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99
FTA - 2003	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	0.99
FTA - 2004	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	1.00	0.99	0.99	0.99
FTA - 2005	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99
FTA - 2006	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99
FTA - 2007	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99
FTA - 2008	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99
FTA - 2009	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99
FTA - 2010	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.98	0.99	0.99	0.99	0.99
FTA - 2011	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.98	0.99	0.99	0.99	0.99
FTA - 2012	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99
FTA - 2013	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.98	0.99	0.99	0.99	0.99
FTA - 2014	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	1.00	0.99	0.99	0.99
FTA - 2015	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	0.99
FTA - 2016	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99
FTA - 2017	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99
FTA - 2018	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99
FTA - 2019	0.99	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.98	0.99	0.98	0.99	0.99	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99
FTA - 2020	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99
FTA - 2021	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00
FTA - 2022	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00
FTA - 2023	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00

Figure 5.6: Correlation Heatmap of FTA data

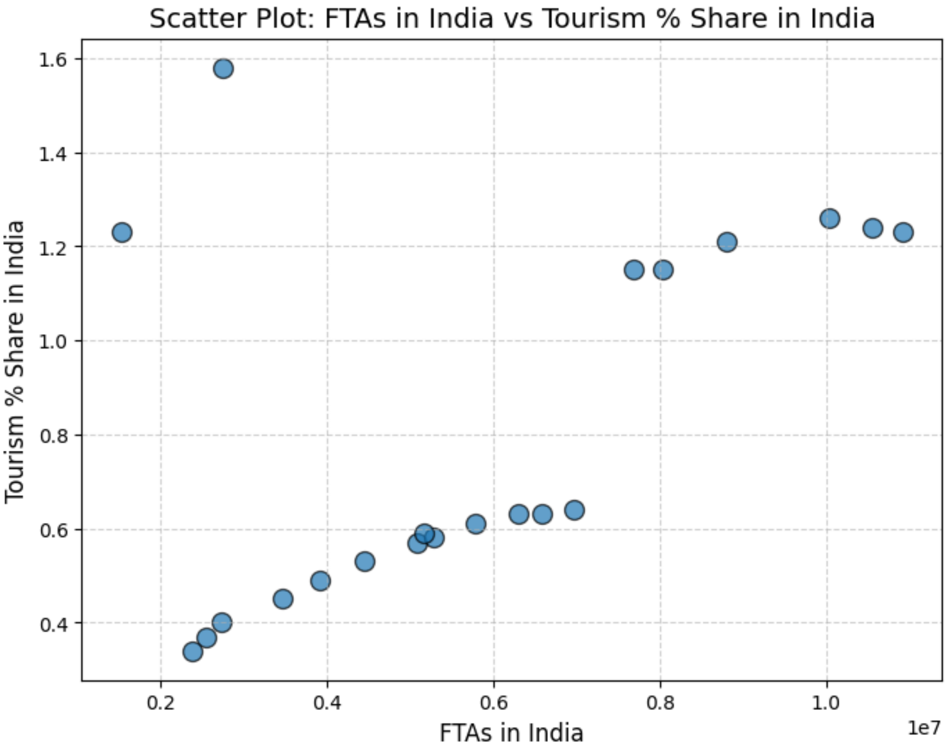


Figure 5.7: FTAs in India vs Tourism % share

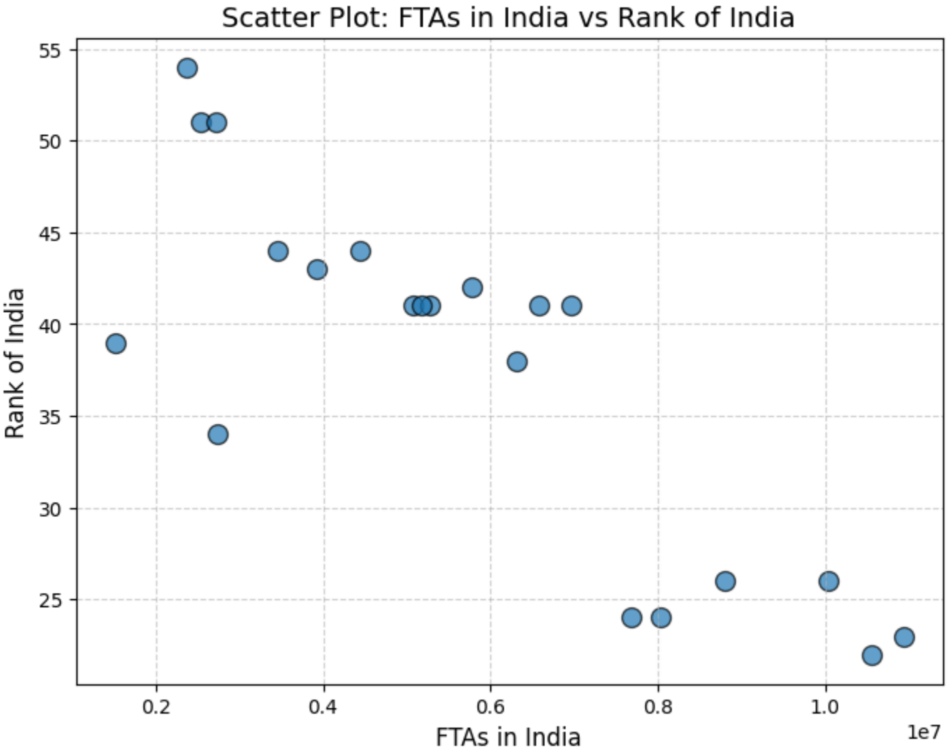


Figure 5.8: FTAs vs Rank of India in world tourism

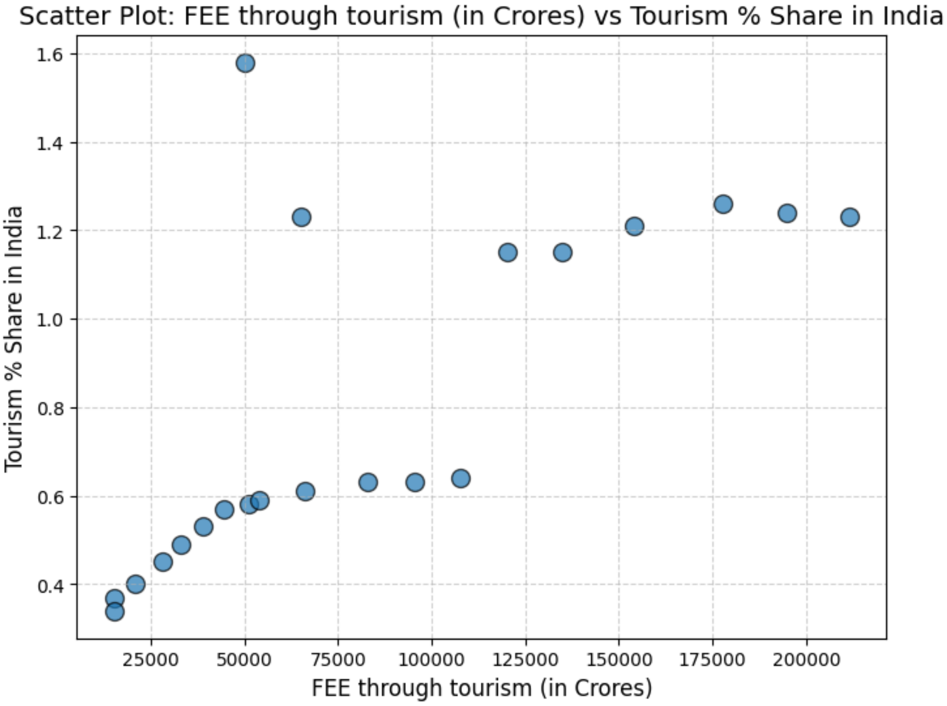


Figure 5.9: FEE vs % Tourism Share

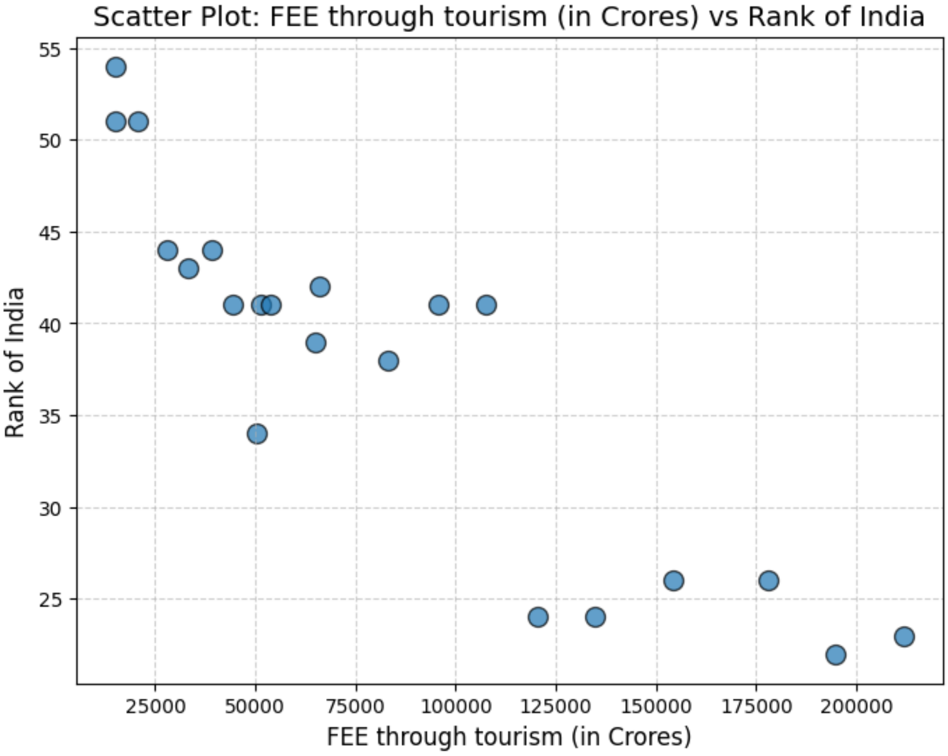


Figure 5.10: FEE vs Rank of India in world tourism

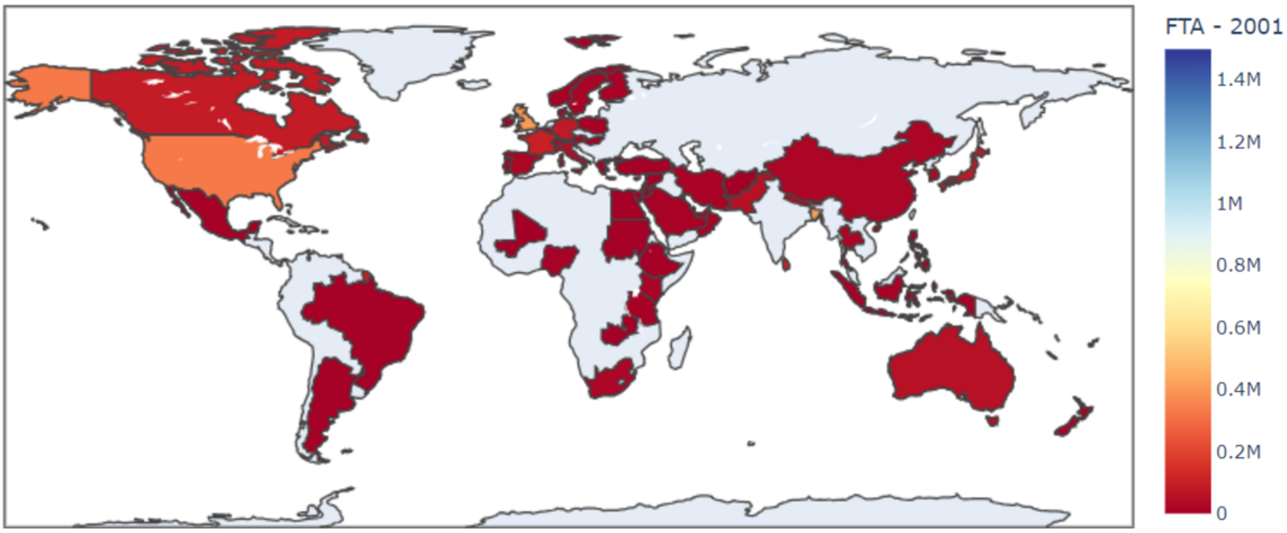


Figure 5.11: World Map Chloropleth showing region-based density of FTAs into India

Chapter 6. Feature Engineering

6.1 Feature Selection

After completing the visualizations on the dataset, we moved forward with feature engineering. The dataset contains FTA values as columns (e.g., `FTA-2001` to `FTA-2023`). This structure highlights the potential to predict future FTA values based on nationalities and categories, as the dataset also includes columns for *Category* and *Nationality*.

Given the year-wise FTA values and the objective of forecasting future FTA values, it is evident that the dataset has a temporal structure, categorizing it as time-series data. To prepare the data for time-series analysis and prediction, the dataframe was reshaped by converting year values into rows. This transformation ensures compatibility with the requirements of time-series analysis.

Code for Reshaping the Dataframe

Below is the code snippet for reshaping the dataframe:

```
# Melt the dataframe to convert years to a single column(to make into time-series data)
    melted_df = _df_.melt(id_vars=[’Category’, ’Nationality’],
                         var_name=’Year’,
                         value_name=’FTA’)
```

Listing 6.1: Reshaping the data

Since the dataset contains both *Category* and *Nationality* information, it enables us to predict future FTA values either category-wise or nationality-wise. In this analysis, we focus on predicting future FTA values using the `Category` column, i.e., predicting category-wise FTA values.

For this prediction, we used libraries from sklearn, specifically the **LinearRegression**, **RandomForestRegressor** and **GradientBoostingRegressor** models to predict the FTA values effectively.

Since the dataset is time-series data, we created additional features based on current values to assist in future value predictions. First, we performed the ”prepare dataset” step, where we prepared the dataframe to make it compatible for input. During this process, we added three additional features that are crucial for model predictions and feature selection. These features are designed to capture time-based trends and improve model performance.

The features selected are as follows:

Year Squared

- **Purpose:** Adds a quadratic term for the `Year` column.

- **Why:** It helps capture non-linear trends in time-series data. Many time series patterns evolve in non-linear ways over time, and this feature can improve prediction accuracy.

Lag Feature

- **Purpose:** Creates a feature that represents the FTA value of the previous year for each region (`Category/Nationality`).
- **Why:** Lag features capture the influence of past values on the current value, which is common in time-series data (autocorrelation).

Rolling Mean (3-Year)

- **Purpose:** Computes a 3-year rolling mean of FTA values for each region (`Category`).
- **Why:** Smoothens the data to reduce short-term fluctuations and highlights longer-term trends. Rolling features are often predictive in time-series forecasting.

Below is the code for adding these additional features:

```
# Creating additional required features
    continent_yearly['Year_Squared'] = continent_yearly['Year'] ** 2
    continent_yearly['Lag_1_Year'] = continent_yearly.groupby('Category')['FTA'].shift(1)
    continent_yearly['Rolling_Mean_3Year'] = continent_yearly.groupby('Category')['FTA'].rolling(window=3, min_periods=1).mean().reset_index(0, drop=True)
    # Selected features
    features = ['Year', 'Year_Squared', 'Lag_1_Year', 'Rolling_Mean_3Year']
```

Listing 6.2: Feature selection

Chapter 7. Model fitting

For model fitting, we used **preprocessed data**, where we performed a train-test split and trained the model on the prepared data. Since feature selection was already completed, we split the data into training and testing sets and then proceeded with model fitting. Before fitting the model, we transformed the data using **StandardScaler**.

The **StandardScaler** transformation standardizes the Foreign Tourist Arrival (FTA) values by scaling them to have a mean of 0 and a variance of 1. This transformation ensures that the data is scaled properly, helping the model optimize predictions by concentrating most values near the mean.

We used three different models for prediction. After fitting the models to the training data, we evaluated their performance using key metrics that measure prediction accuracy:

- **Mean Squared Error (MSE):** MSE calculates the average of the squared differences between the actual and predicted values. Larger errors result in significantly higher MSE, making it sensitive to outliers.

- **Mean Absolute Error (MAE):** MAE computes the average of the absolute differences between actual and predicted values. Unlike MSE, MAE does not square the error terms, making it less sensitive to outliers.

- **R^2 Score**: The R^2 score evaluates how well the model's predictions match the original data. It represents the proportion of variance in the target variable that is explained by the model. R^2 ranges from 0 to 1, with 1 indicating a perfect fit.

Below is the code snippet for model fitting:

```python
# Train-test split
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

        # Scaling the features for proper predictions
        scaler = StandardScaler()
        X_train_scaled = scaler.fit_transform(X_train)
        X_test_scaled = scaler.transform(X_test)

        results[continent] = {
            'X_train': X_train_scaled,
            'X_test': X_test_scaled,
            'y_train': y_train,
            'y_test': y_test,
            'scaler': scaler
        }
    # Evaluate model using mae, mse, r2 metrics
```

```
        mse = mean_squared_error(data[’y_test’], y_pred)
        mae = mean_absolute_error(data[’y_test’], y_pred)
        r2 = r2_score(data[’y_test’], y_pred)
```

Listing 7.1: Model fitting

7.1 Regression

To predict future Foreign Tourist Arrival (FTA) values, which are numeric, we utilized regression models. Specifically, we implemented **Linear Regression**, **Random Forest Regressor**, and **Gradient Boosting Regressor**. Below are the reasons for selecting these models:

- **Linear Regression:** This model assumes a linear relationship between the input features and the target variable. If the FTA values exhibit a linear trend across the years, this model can provide accurate predictions.

- **Random Forest Regressor**: Random Forest is a powerful prediction model based on an ensemble of decision trees. Unlike linear regression, it does not assume a linear relationship between input and output features. This makes it particularly suitable for datasets where the relationship is non-linear, such as in our case.

- **Gradient Boosting Regressor**: Gradient Boosting is a robust ensemble machine learning algorithm that builds a sequence of weak prediction models, often decision trees, in a sequential manner. It is highly effective for time-series data, as it captures complex relationships and trends within the data.

Using the training and testing datasets, we applied these machine learning algorithms to predict future FTA values and analyze their performance.

7.2 Machine Learning Algorithms

Future Feature Engineering

For each year in *future_years*, new features are generated to align with the input requirements of the model:

- **Year:** Represents the target year for prediction.

- **YearSquared:** A non-linear term introduced to capture quadratic trends in the data.

- **Lag1Year:** Refers to the previous year’s predicted Foreign Tourist Arrival (FTA) value.

- **RollingMean3Year:** A rolling average computed over the last three years’ FTA values.

Scaling Features

The newly created features are transformed using the same `StandardScaler` (`data[’scaler’]`) that was applied during the training phase. This ensures consistency between the training and prediction data.

Iterative Predictions

- The model predicts the next year's FTA based on the scaled feature set.
- The predicted value is then used as the lag feature (*Lag1Year*) for subsequent years.
- The rolling mean is updated iteratively to incorporate the most recent predictions, enabling predictions for multiple years into the future.

Final Predictions

The predicted FTA values for all future years are organized into a dictionary (*predictions*), grouped by continent.

Visualizing Predictions

The visualization utility is designed to display the predicted Foreign Tourist Arrivals (FTAs) for various continents on a unified graph. This visualization helps in comparing trends across continents and provides an intuitive understanding of the prediction results.

Code for Predictions

Below is the code snippet for the machine learning algorithm used to generate these predictions:

```python
# Preparing for future predictions
    future_X = []
    last_fta = y_full.iloc[-1]  # Initialize with last FTA value
    future_y = list(y_full[-2:])  # Storing the last 2 values for rolling mean calculation
    rolling_mean = np.mean(future_y)  # Initializing rolling mean

    for year in future_years:
        # Create a future row
        future_row = pd.DataFrame(
            [[year, year**2, last_fta, rolling_mean]],
            columns=['Year', 'Year_Squared', 'Lag_1_Year', 'Rolling_Mean_3Year']
        )
        #transform the data using standardScaler
        future_row_scaled = data['scaler'].transform(future_row)

        # Predicting next year's FTA
        last_fta = model.predict(future_row_scaled)[0]
        future_X.append([year, year**2, last_fta, rolling_mean])

        # Updating rolling mean with new prediction
        future_y.append(last_fta)
        rolling_mean = np.mean(future_y[-3:])  # Updating rolling mean with the last 3 values
```

Listing 7.2: ML Predictions

Chapter 8. Conclusion & future scope

8.1 Findings/observations

Tourism, as a vital contributor to India's economy, offers profound insights through the study of Foreign Tourist Arrivals (FTAs) and their related economic metrics. In our analysis, we undertook an extensive examination of tourism trends spanning over two decades, integrating diverse datasets to provide a comprehensive view of India's global tourism footprint.

Our investigation highlighted several pivotal patterns and key findings:

- **Age Group Distribution:** Middle-aged adults (25-54 years) emerged as the predominant contributors to FTAs, reflecting their higher travel frequency and disposable incomes. This demographic insight is crucial for tailoring tourism services to this segment.

- **Quarterly Trends:** The quarterly analysis underscored seasonal peaks in FTAs, particularly during cultural festivals and winter months, indicating the impact of climate and cultural appeal on tourist inflows.

- **Regional Analysis:** Western Europe and West Asia dominated as primary sources of foreign tourists, emphasizing the need for sustained diplomatic and cultural outreach in these regions.

- **Impact of COVID-19:** The pandemic significantly disrupted the tourism sector, with sharp declines in FTAs during 2020-2021. This anomaly underscored the sector's vulnerability to global crises.

- **Economic Contribution:** Analyzing the correlation between FTAs and Foreign Exchange Earnings (FEE) affirmed tourism's critical role in boosting India's economy, with clear implications for national economic policy.

8.2 Challenges

1. **Regional Disparities in Tourist Inflows:**
   While India attracts a significant number of tourists from regions like Western Europe and West Asia, the over-reliance on these areas highlights a critical challenge. This limited diversity in tourist origins makes the sector highly vulnerable to global geopolitical or economic crises affecting these regions. Expanding outreach to underrepresented areas such as Africa, South America, and East Asia remains an ongoing struggle.

2. **Seasonality of Tourism:**
   Tourism in India heavily depends on specific seasons, with peaks during winter and cultural festivals. While this creates vibrant tourist experiences, it poses challenges for sustaining consistent demand throughout the year. The off-peak periods often lead to underutilized infrastructure and employment issues in the sector, highlighting the need for strategies to promote year-round tourism.

3. **Age Group-Specific Preferences:**
   Our analysis revealed that middle-aged adults (25-54 years) are the predominant contributors to foreign tourist arrivals. This focus on a single demographic group indicates a missed opportunity to engage younger (15-24) and senior (55+) travelers. India must design inclusive and diverse offerings to cater to all age groups, ensuring that its tourism appeal is both broad and balanced.

4. **Impact of Global Crises:**
   The COVID-19 pandemic underscored the vulnerability of India's tourism industry to global crises. The sharp decline in foreign tourist arrivals during this period disrupted livelihoods and strained the economy. Strengthening the sector's resilience through better crisis management and financial support mechanisms is crucial for ensuring stability in the face of future disruptions.

5. **Insufficient Infrastructure:**
   While India's cultural and natural attractions are globally renowned, the supporting infrastructure often falls short. Poor accessibility, inadequate transportation, and limited accommodation in rural or heritage sites deter many potential visitors. Investing in world-class infrastructure, especially in lesser-known regions, is essential for elevating India's status as a leading global tourist destination.

8.3 Future plan

While our analysis has provided valuable insights, it also opens avenues for further exploration and improvement in both data handling and strategic application:

1. **Enhanced Data Integration:**

   - Incorporating granular datasets, such as tourist satisfaction surveys or regional expenditure reports, can provide a multidimensional view of tourism's impact.
   - Expanding the analysis to include domestic tourism trends can bridge the understanding of India's overall tourism ecosystem.

2. **Policy Implications:**

   - Insights from our study can aid policymakers in enhancing tourism infrastructure, particularly targeting middle-aged and high-spending tourists.
   - Developing targeted campaigns for underrepresented regions, such as Africa or South America, could diversify India's tourist demographics.

3. **Predictive Modeling and Forecasting:** Advanced machine learning techniques, such as LSTM or ARIMA models, could further improve the accuracy of FTA forecasting, accounting for macroeconomic factors like currency fluctuations or geopolitical events.

4. **Sustainability Focus:** Future studies could incorporate environmental metrics, evaluating tourism's ecological footprint and proposing sustainable practices to mitigate adverse effects.

5. **Real-Time Analytics:** Integration with real-time data sources, such as travel bookings or social media trends, can provide dynamic insights, enabling more responsive strategies.

In conclusion, this analysis underscores the transformative power of data-driven insights in optimizing tourism strategies and enhancing India's global competitiveness. By extending the scope of research and leveraging advanced analytics, we can ensure a robust and sustainable future for India's tourism sector.

Group Contribution

Shravan Kakadiya

- Data Collection for tourism related datasets from data.gov.in .
- Data creation for dataset 1 using different datasets that we got from government website.
- Data Pre-Processing, Cleaning, outlier detections, Visualisations, Model Developments on dataset 1 using EDA steps.
- Made Google Colab runnable IPYNB file on dataset 1 with proper source codes and comments and proper Markdown contents.
- Made some part of report compilation like Feature engineering, Feature selection, Model fitting, Regression and ML algorithms.

Dishant Patel

- Data Collection for tourism related datasets from data.gov.in .
- Majorly Worked on Dataset 2. Performed all EDA steps on dataset 2, Pre-processing, cleaning, outliers detection, visualisation using EDA steps.
- Made Google Colab runnable IPYNB file on dataset 2 with proper source codes and comments and proper Markdown contents.

Madhav

- Data Collection for tourism related datasets from data.gov.in .
- Worked on the problem statement for the project.
- Data creation for dataset 2 using different datasets that we got from government website.
- Data Pre-processing, cleaning, outlier detection for making dataset 2 using necessary EDA.
- Majorly worked on Report compilation.
- Compiled IPYNB file for Dataset 2 and conducted major data visualisation for univariate and bivariate analysis

Short Bio

1. **Shravan Kakadiya** is a 3rd year ICT student at DA-IICT, where he focuses on his interests in physics and web development. His passion for physics drives his curiosity about the natural world, while his skills in web development allow him to explore and create innovative digital projects. Outside of academics, Shravan enjoys watching motorsports, appreciating the excitement and precision of the sport. He is also a vlogger, using this platform to share experiences, insights, and engage with a wider audience. Shravan's diverse interests reflect his blend of scientific curiosity, technical skills, and creative expression.

2. **Dishant Patel** is a 3rd year ICT student at DA-IICT, where he has developed a strong interest in database management systems (DBMS), advanced mathematics, and computational theories. His academic pursuits reflect his passion for understanding complex systems and solving intricate problems. Beyond his studies, Dishant is an avid cricket player, enjoying the sport both for its competitive nature and as a means to stay active. He is also drawn to adventurous activities, seeking out new experiences that challenge him and add excitement to his life. Dishant's well-rounded interests highlight his commitment to both academic excellence and a balanced lifestyle.

3. **Madhav Kanjilimadom** is a dedicated 3rd year Mathematics and Computing student at DA-IICT, where he has developed a strong passion for probability, statistics, and data analytics. His academic interests are complemented by his enthusiasm for applying analytical skills to solve real-world problems. Beyond his studies, Madhav is an avid speedcuber, showcasing his quick-thinking and problem-solving abilities. He is also the captain of the DA-IICT football team, demonstrating his leadership skills, teamwork, and commitment to both sports and the student community. With a well-rounded blend of technical expertise and extracurricular involvement, Madhav continues to pursue excellence in all his endeavors.

Link to Google Colab Notebooks:

1. Colab Notebook for Analysis on Country-wise FTAs into India in the 21st Century.
2. Colab Notebook for Analysis on Foreign Exchange Earnings due to Tourism in India in the 21st Century.

References

[1] Open Government Data (OGD) Platform India *URL:* data.gov.in

[2] FTA Infographics of India *URL:* https://community.data.gov.in/foreign-tourist-arrivals-in-india/

[3] Python Documentation *URL:* https://docs.python.org/