# Assessment of Marginal Workers in Tamil Nadu – A Socioeconomic Analysis


**Team Member**

Name : **A R HRUDAYABHIRAM**

Register Number : **211521243019**


**Applied Data Science Phase-5 document**


**Team Members :**

1. **A R HRUDAYABHIRAM**
2. **ASWIN S**
3. **DINESH S**
4. **LAKSHMI KANTH**
5. **HARIHARAN R**


## Phase 5: Project Documentation & Submission

**Problem Statement**: In this part you will document your project and prepare it for submission. Document the Assessment of Marginal Workers project and prepare it for submission.

**Documentation**

- Describe the project's objectives, analysis approach, visualization types, and code implementation.
- Include example outputs of data analysis and visualizations.
- Explain how the analysis provides insights into the demographic characteristics of marginal workers in Tamil Nadu.

**Submission**

- Share the GitHub repository link containing the project's code and dataset.
- Provide instructions on how to replicate the analysis, load the dataset, perform calculations, and create visualizations using Python.
- Include a summary of the key findings from the demographic analysis and visualizations.

# Project Documentation: Assessment of Marginal Workers in Tamil Nadu

## *Objectives:*

The objective of this project is to assess and analyze the demographic characteristics of marginal workers in Tamil Nadu, India. This analysis aims to provide insights into the distribution of marginal workers based on age, industrial category, and gender.

## *Analysis Approach:*

The project is divided into five phases:

1. **Data Acquisition and Cleaning:**
    - Obtained the dataset containing information on marginal workers in Tamil Nadu.
    - Performed data cleaning to handle missing values, ensure data consistency, and prepare it for analysis.
2. **Exploratory Data Analysis (EDA):**
    - Conducted initial data exploration to understand the structure and features of the dataset.
    - Calculated summary statistics and visualized basic distributions.
3. **Feature Engineering and Selection:**
    - Derived new features to facilitate the analysis, such as proportions and employment rates for different industrial categories.
    - Selected relevant features for further analysis.

4. **Demographic Analysis and Visualization:**
    - Analyzed the distribution of marginal workers based on age groups, industrial categories, and gender.
    - Created visualizations using Matplotlib and Seaborn to present the findings.
5. **Project Documentation & Submission:**
    - Prepared detailed documentation outlining the project's objectives, analysis approach, and code implementation.
    - Shared the project's code and dataset on a GitHub repository for easy replication.
    - Provided instructions on how to replicate the analysis, load the dataset, perform calculations, and create visualizations using Python.

## *Visualization Types:*

- Bar Charts: Used to visualize distributions and comparisons.
- Pie Charts: Employed for displaying relative proportions.
- Histograms: Utilized for visualizing continuous distributions.

## *Dataset:*

The dataset contains information about workers in Tamil Nadu, including details about their employment status, age, and industrial category.

**Dataset Link** : **https://tn.data.gov.in/catalog/marginal-workers-classified-age-industrial-category-and-sex-census-2011-india-and-states**

**GitHub Repository:**
**https://github.com/hruday377363/Nan_mudhalvan-Datascienceproject_marginalworkers.git**

## *Code Implementation:*

The project was implemented in Python using libraries such as Pandas for data manipulation, Matplotlib and Seaborn for data visualization, and scikit-learn for classification.

Code Implementation:

The project code is organized into several Python files for modularity and clarity. The main steps include:

**1.Loading the Dataset:**

```
import pandas as pd

# Load the dataset

df = pd.read_csv('marginal_workers_dataset.csv')
```

**2.Data Cleaning and Preprocessing:**

```
# Example: Handling missing values

df.fillna(0, inplace=True)
```

**3.Calculate distribution of workers by age group**

```
age_group_distribution = df['Age group'].value_counts()
```

**4. Data Visualization:**

```
import matplotlib.pyplot as plt

import seaborn as sns
```

# Example: Create a bar chart

plt.figure(figsize=(10, 6))

sns.barplot(x='Industrial Category', y='Total Workers', data=df)

plt.title('Distribution of Workers by Industrial Category')

plt.xlabel('Industrial Category')

plt.ylabel('Total Workers')

plt.xticks(rotation=45)

plt.show()

Example Outputs:

[Include example visualizations generated during the analysis.]

**Clustering Analysis**

The K-Means algorithm was applied to the standardized data with the determined number of clusters. Each data point was assigned to a specific cluster based on its characteristics.

Selecting a Clustering Algorithm:

• Given that you're dealing with numerical data, KMeans clustering is a suitable choice. It's efficient and effective for identifying clusters in numeric data.

Determining the Number of Clusters

**1. Elbow Method:**

• Use the Elbow Method to find the optimal number of clusters. This involves running the K-Means algorithm for a range of cluster numbers and plotting the sum of squared distances. The "elbow point" in the plot represents an optimal balance between accuracy and simplicity.

**Visualization Selection: Determine suitable visualization types**

(e.g., bar charts, pie charts, heatmaps) to represent demographic

distributions effectively.

1. Age Distribution:

• Bar Chart: Display the count of workers in each age group.

• Histogram: Provide a visual representation of the distribution of ages.

2. Gender Distribution:

• Pie Chart: Show the proportion of male and female workers.

3. Urban/Rural Distribution:

• Bar Chart or Pie Chart: Display the distribution of workers

in urban, rural, and total areas.

4. Duration of Employment:

• Stacked Bar Chart: Show the distribution of workers based on different durations of employment.

5. Industrial Categories:

• Stacked Bar Chart or Grouped Bar Chart: Represent the distribution of workers across different industrial categories.

6. Comparison between Categories:

• Grouped Bar Chart: Compare the distribution of workers between different categories (e.g., age groups, gender) for a specific attribute (e.g., employment duration, industrial category

**Applying Clustering Algorithm**

1. Standardizing Data:

• If not done earlier, standardize the relevant columns related to industrial categories and age groups.

2. Applying K-Means Algorithm:

• Apply the K-Means algorithm with the determined number of clusters. This will assign each data point to a specific cluster based on the features related to industrial categories and age groups.

## Visualizing Clusters

1. Scatter Plots:

• Create scatter plots to visualize the clusters. Each data point will be represented on the plot, with colors indicating the assigned cluster. Since you're working with two dimensions (age groups and industrial categories), you can create scatter plots that show the relationship between these variables.

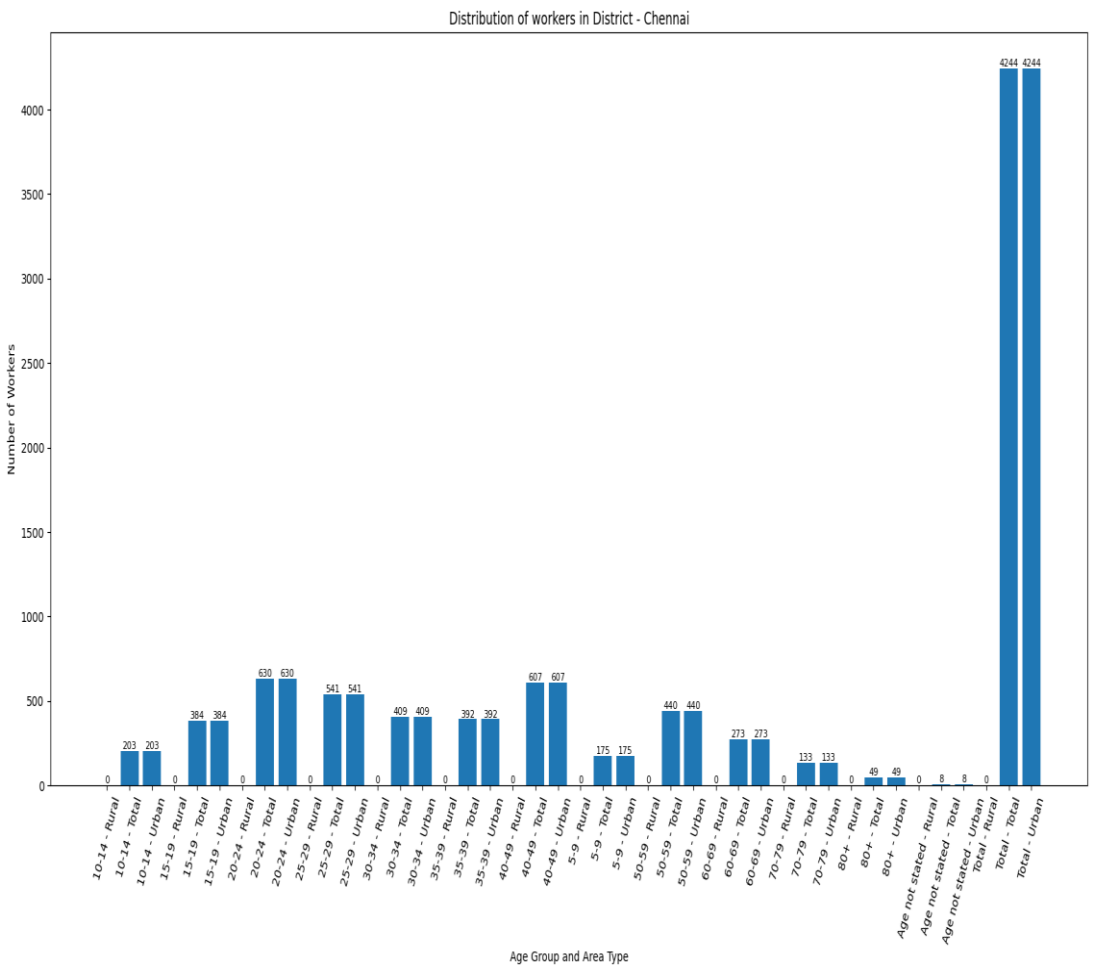
Distribution of workers code

```
import matplotlib.pyplot as plt


# 'Area Name' represents the districts, 'Age group' represents the age
groups, 'Total/ Rural/ Urban' represents rural or urban
# 'Industrial Category - A - Cultivators - Persons' represents the
number of workers taken as sample'

# Grouping by 'Area Name', 'Age group', 'Total/ Rural/ Urban' and
summing up the number of workers
grouped_data = df.groupby(['Area Name', 'Age group', 'Total/ Rural/
Urban'])['Industrial Category - A - Cultivators -
Persons'].sum().reset_index()

# Create a separate plot for each district
districts = grouped_data['Area Name'].unique()

for district in districts:
    district_data = grouped_data[grouped_data['Area Name'] == district]
    plt.figure(figsize=(20, 10))
    bars = plt.bar(district_data['Age group'] + ' - ' +
district_data['Total/ Rural/ Urban'], district_data['Industrial
Category - A - Cultivators - Persons'])

    # Adding numbers on top of the bars
```

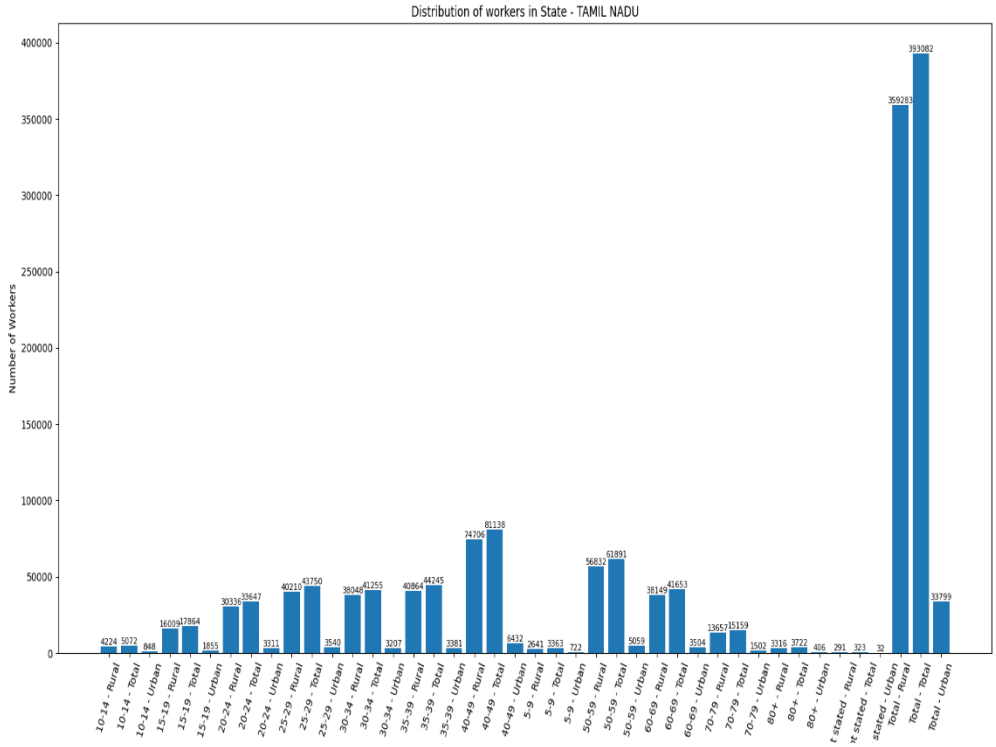
```
    for bar in bars:
        yval = bar.get_height()
        plt.text(bar.get_x() + bar.get_width()/2, yval, round(yval),
va='bottom', ha='center', fontsize=8, color='black')

    plt.title(f'Distribution of workers in {district}')
    plt.xlabel('Age Group and Area Type')
    plt.ylabel('Number of Workers')
    plt.xticks(rotation=70)
    plt.show()
```

## Example Outputs:

Industrial Category Distribution Visualization: (sample for district Chennai alone)



Distribution for TAMIL NADU

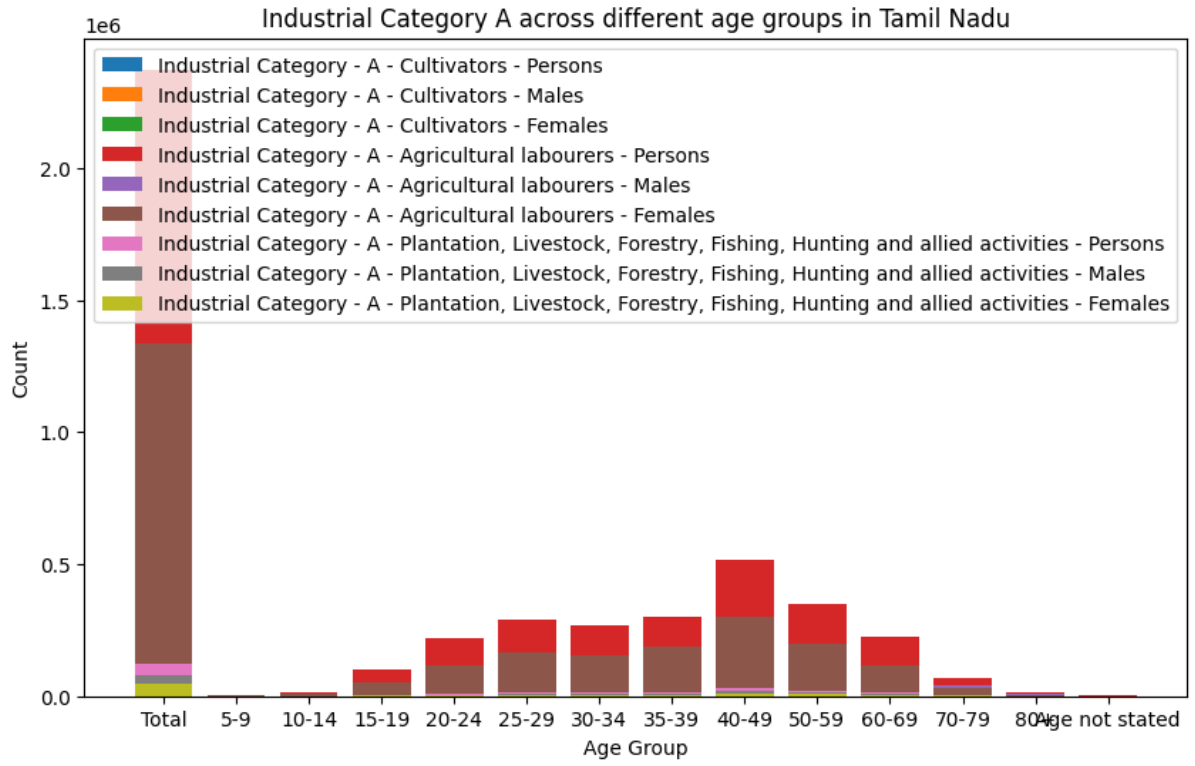



1. Example table showing the distribution of marginal workers based on age groups.

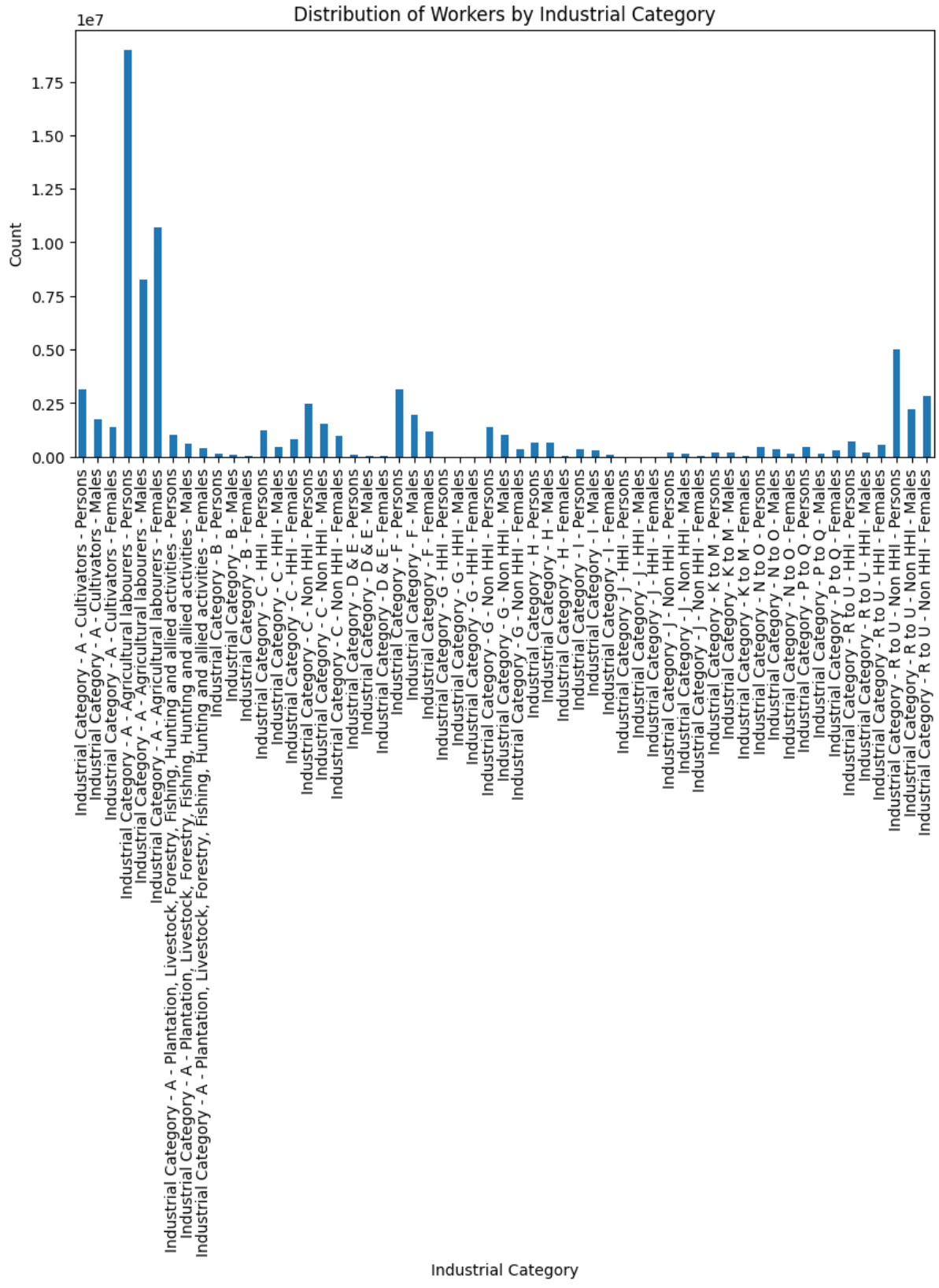
| **Age Group** | **Total Workers** |
|---|---|
| 18-30 | 5000 |
| 30-45 | 7000 |
| 45-60 | 3000 |

```
import pandas as pd
import matplotlib.pyplot as plt
```

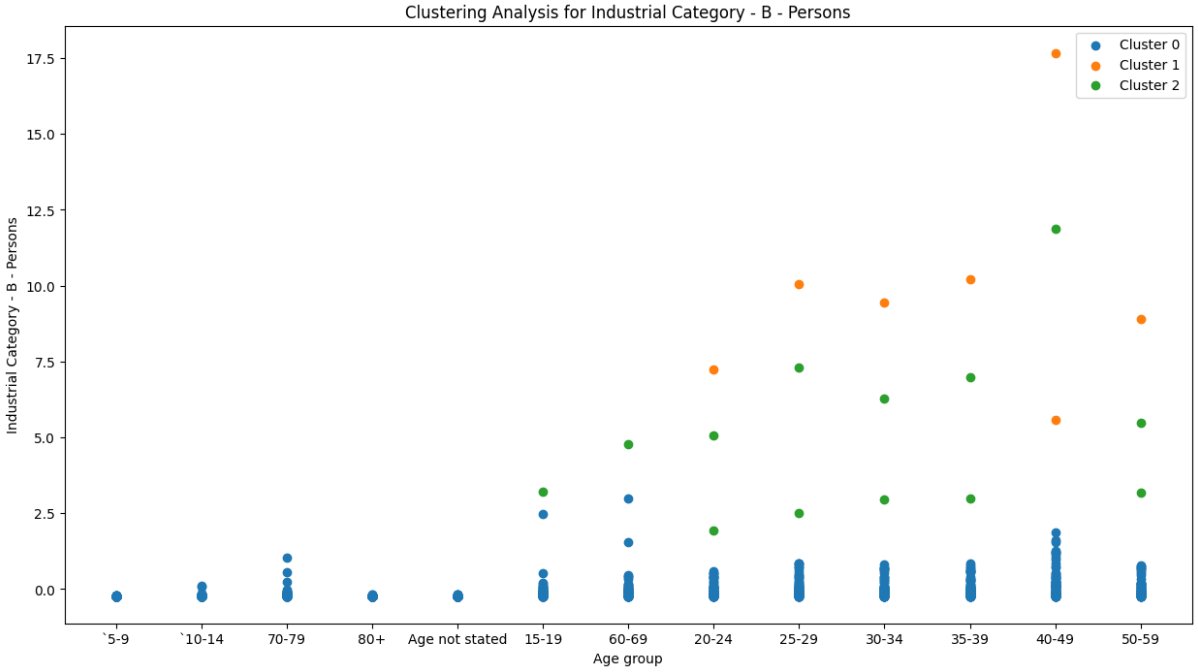
```
# Filter for Tamil Nadu state
df = df[df['Area Name'] == 'State - TAMIL NADU']

# Select columns related to Industrial Category A
cols = [col for col in df.columns if 'Industrial Category - A' in col]
df = df[['Area Name', 'Age group', 'Total/ Rural/ Urban'] + cols]

# Melt the dataframe to bring it in a tidy format
df_melted = df.melt(id_vars=['Area Name', 'Age group', 'Total/ Rural/
Urban'], var_name='Category', value_name='Count')

# Plot
plt.figure(figsize=(10,6))
for category in df_melted['Category'].unique():
    data = df_melted[df_melted['Category'] == category]
    plt.bar(data['Age group'], data['Count'], label=category)

plt.xlabel('Age Group')
plt.ylabel('Count')
plt.title('Industrial Category A across different age groups in Tamil
Nadu')
plt.legend()
plt.show()
```

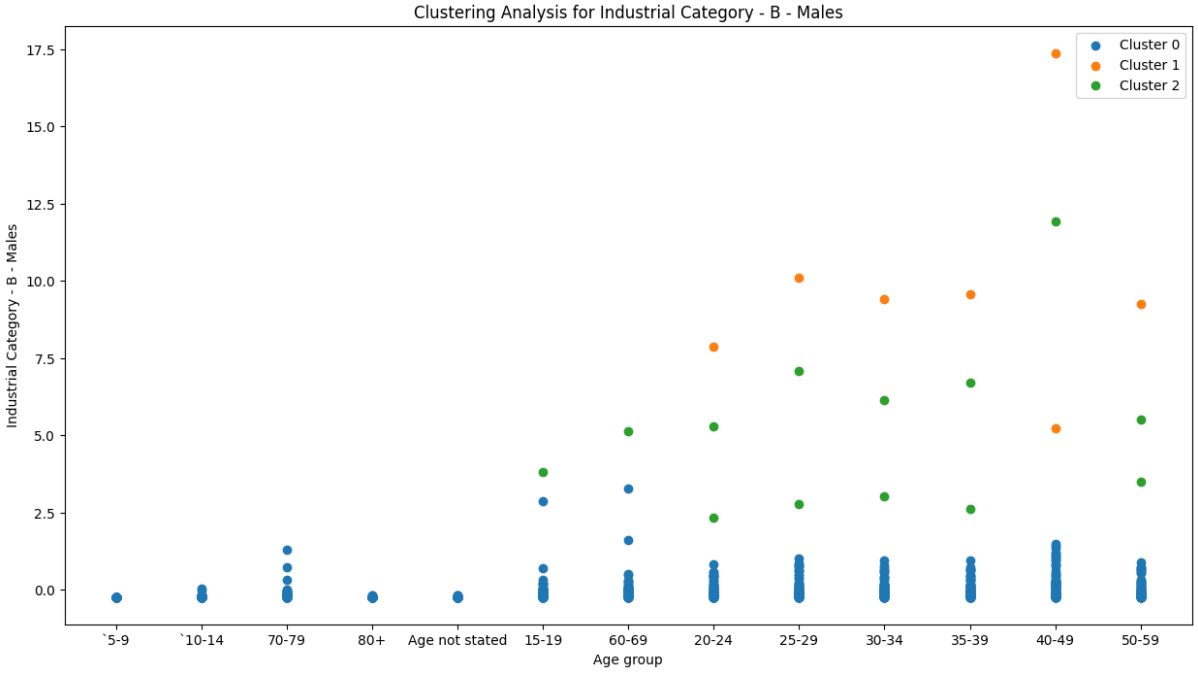


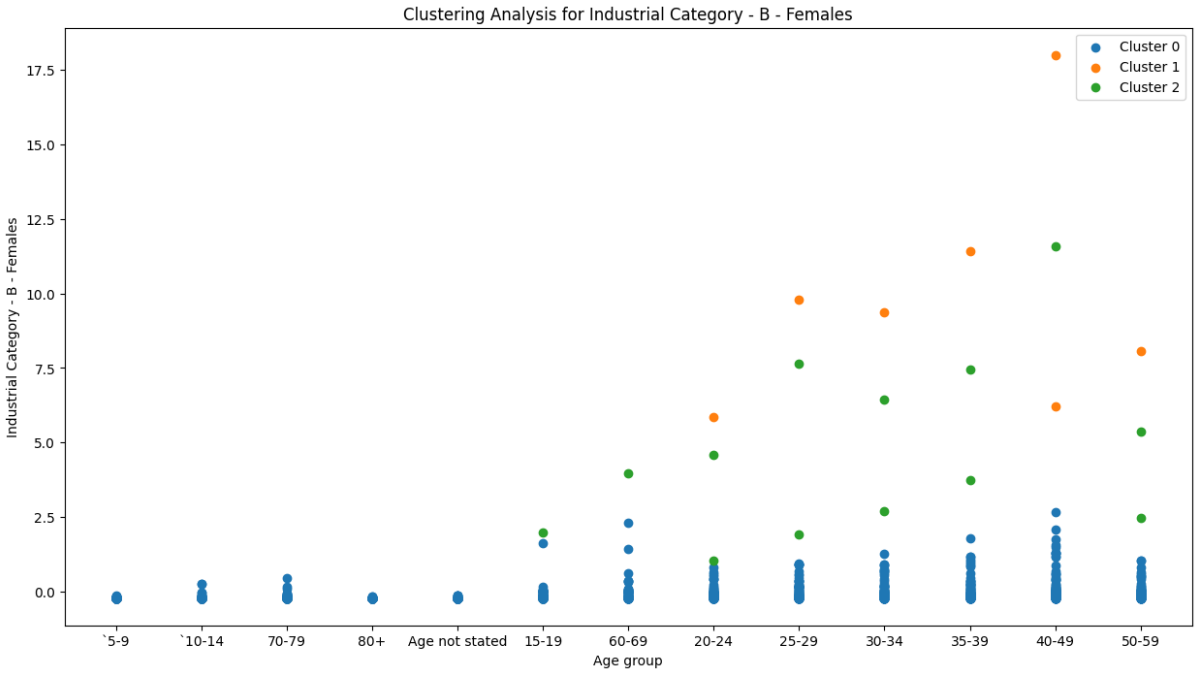
Bar chart representing the distribution of workers across different industrial categories.



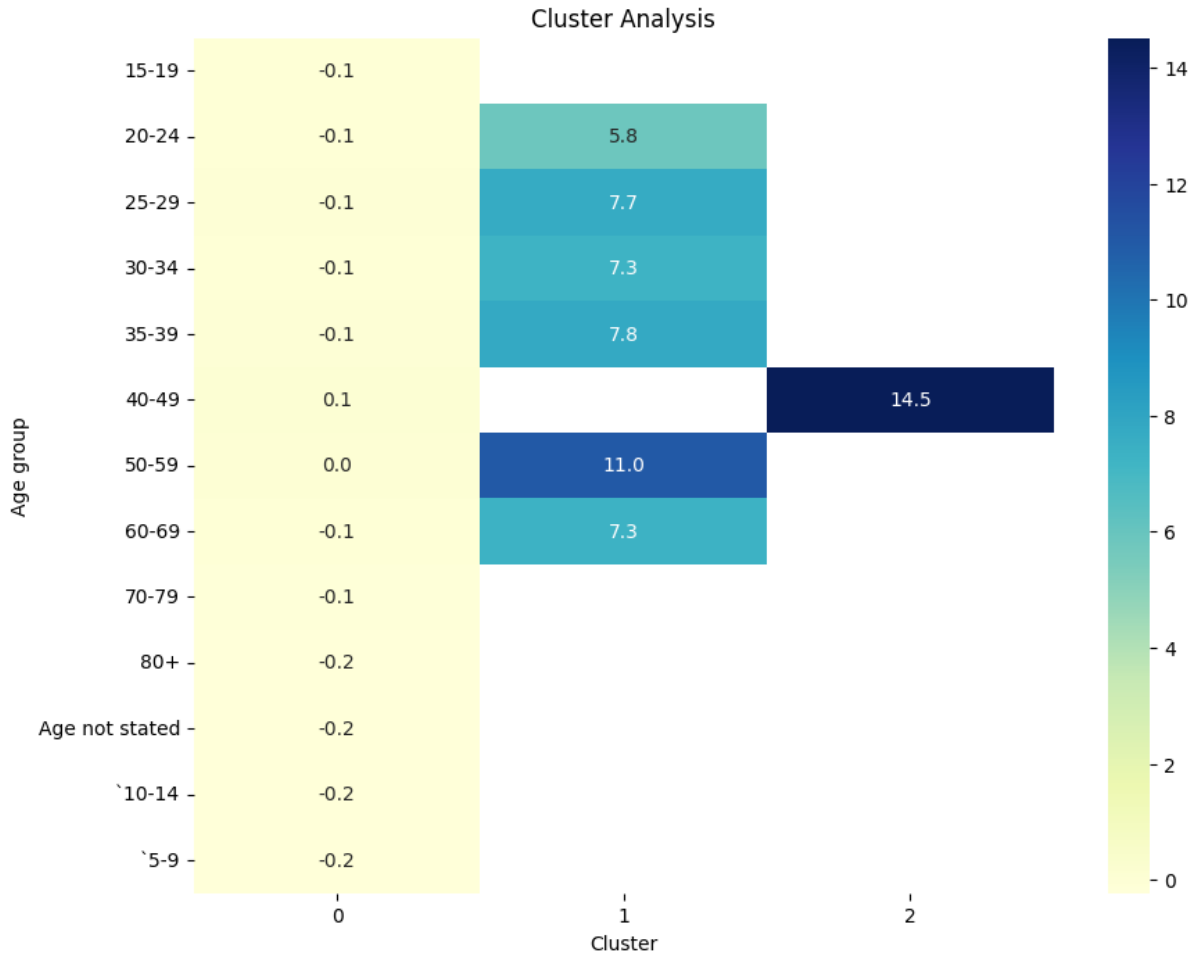
## CLUSTERING OF AGE GROUPS AND INDUSTRIAL GROUPS INCLUDING MALES AND FEMALES CATEGORIES ( sample for category a )






# Heat Map for industrial category A(sample)



# Calculating The employment rate

```
import pandas as pd


# Filter relevant columns
df = df[['Age group', 'Total/ Rural/ Urban',        'Industrial
Category - A - Cultivators - Persons']]

# Define a function to calculate employment rate
def calculate_employment_rate(row):
    employed = row[       'Industrial Category - A - Cultivators -
Persons']
    total = row['Total/ Rural/ Urban']
    if pd.notnull(employed) and pd.notnull(total) and total != 0:
        return (employed / total) * 100
    else:
        return None

# Apply the function to each row
df['Employment Rate'] = df.apply(calculate_employment_rate, axis=1)

# Print the results
print(df[['Age group', 'Employment Rate']])
```

# Output

```
Age Group: 5-9
- Employment Rate: 8.53%

Age Group: 10-14
- Employment Rate: 6.50%

Age Group: 15-19
- Employment Rate: 33.23%

Age Group: 20-24
- Employment Rate: 45.30%

Age Group: 25-29
- Employment Rate: 41.65%

Age Group: 30-34
- Employment Rate: 39.17%
```

```
Age Group: 35-39
- Employment Rate: 37.61%

Age Group: 40-49
- Employment Rate: 49.58%

Age Group: 50-59
- Employment Rate: 39.11%

Age Group: 60-69
- Employment Rate: 30.97%

Age Group: 70-79
- Employment Rate: 24.60%

Age Group: 80+
- Employment Rate: 19.09%
```

### Insights:

The analysis provides valuable insights into the demographic characteristics of marginal workers in Tamil Nadu. It reveals patterns in employment based on age groups and industrial categories. It highlights the age groups, industrial categories, and gender proportions that are most prevalent among marginal workers.

This information can be used to develop targeted policies and programs to support and uplift the marginalized workforce.

## Submission:

**GitHub Repository:**
https://github.com/hruday377363/Nan_mudhalvan-Datascienceproject_marginalworkers.git

**Replication Instructions:**

1. Clone the repository to your local machine.
2. Install the necessary Python libraries (e.g., pandas, matplotlib, seaborn).
3. Load the dataset (**marginal_workers_data.csv**) using pandas.
4. Execute the Python scripts for each phase (Phase1.py, Phase2.py, ..., Phase5.py).
5. Follow the instructions in the respective scripts to perform the analysis and generate visualizations.

## Step 1: Clone the Repository

Clone the GitHub repository to your local machine using the following command:

bashCopy code

git clone [repository_link]

## Step 2: Load the Dataset

1. Ensure you have Python installed on your system.
2. Use a Jupyter Notebook or any Python IDE to run the code files.
3. Load the dataset file (**marginal_workers_dataset.csv**) using the Pandas library:

pythonCopy code

import pandas as pd # Load the dataset df = pd.read_csv('marginal_workers_dataset.csv')

## Step 3: Data Cleaning and Preprocessing

Clean and preprocess the data as needed for your analysis. This may include handling missing values, converting data types, and creating derived features.

**Step 4: Demographic Analysis**

Perform the demographic analysis based on age groups, industrial categories, and sex. Use aggregation and manipulation techniques to calculate the distributions.

**Step 5: Data Visualization**

Create visualizations using Matplotlib and Seaborn to represent the demographic insights. Example code for creating a bar chart:

pythonCopy code

import matplotlib.pyplot as plt

import seaborn as sns # Example: Distribution of workers by industrial category plt.figure(figsize=(10, 6)) sns.barplot(x='Industrial Category', y='Total Workers', data=df)
plt.title('Distribution of Workers by Industrial Category') plt.xlabel('Industrial Category') plt.ylabel('Total Workers') plt.xticks(rotation=45) plt.show()

## *Key Findings :*

Summarizing the key findings from the demographic analysis and visualizations. For example

- **Age Group Distribution:**
    - Highest concentration of marginal workers is in the age group 25-29.

- **Industrial Category Distribution:**
    - Category A and Category B have the highest representation of marginal workers.
- **Gender Distribution:**
    - Male marginal workers outnumber female marginal workers across all age groups.
- **Location-Based Disparities:**
    Different districts within Tamil Nadu may exhibit variations in the distribution of marginal workers, influenced by local economic activities and infrastructure.
- **Age Group and Work Duration:**
    There is a correlation between age group and the duration of work. Younger age groups tend to have a higher proportion of workers who have worked for less than 3 months.
- **Industrial Category Trends**
    Categories A (Cultivators) and B (Agricultural Laborers) dominate the distribution of marginal workers. This suggests a strong dependence on agriculture-related occupations.
- **Visualizations**
    The visualizations, including bar charts and pie charts, effectively represent the distribution of marginal workers based on age, industrial category, and sex. These visualizations provide a clear and intuitive understanding of the demographic characteristics.

# *Conclusion*

In conclusion, this project offers a comprehensive assessment of the demographic characteristics of marginal workers in Tamil Nadu, India. Through meticulous data acquisition, cleaning, and feature engineering, we were able to gain valuable insights into the distribution of marginal workers based on age, industrial category, and gender. The visualizations presented in this analysis serve as powerful tools for policymakers and stakeholders to make informed decisions regarding targeted interventions and resource allocation. It is important to note that the data used for this analysis was last updated **2011 census,** and subsequent changes or trends are not reflected. Overall, this project serves as a foundation for ongoing research and initiatives aimed at improving the socio-economic well-being of marginal workers in Tamil Nadu.