Project 7: TN Marginal Workers Assessment

Phase 1: Problem Definition and Design Thinking

Problem Definition

The TN Marginal Workers Assessment project aims to perform a socioeconomic analysis of marginal workers in Tamil Nadu based on their demographic characteristics, including age, industrial category, and sex. The primary objective is to gain insights into the distribution of marginal workers across different categories and to represent this information using effective visualizations. This project involves defining clear objectives, designing the analysis approach, selecting appropriate visualization types, and conducting the analysis using Python and data visualization libraries.

Design Thinking

Project Objectives

Objective 1: Analyze the demographic characteristics of marginal workers, with a focus on age and gender distribution.

Objective 2: Explore the distribution of marginal workers across various industrial categories.

Analysis Approach

Step 1: Data Extraction

Access the dataset from the provided link (Dataset Link: Marginal Workers Classified by Age, Industrial Category, and Sex – Census 2011).

Verify data integrity and completeness.

Step 2: Data Preprocessing

Clean and preprocess the dataset, handling any missing or inconsistent data.

Organize the data into suitable data structures for analysis.

Step 3: Demographic Analysis

Perform an analysis of age and gender distribution among marginal workers.

Calculate summary statistics such as mean age, gender ratios, etc.

Step 4: Industrial Category Analysis

# Import necessary libraries

Import pandas as pd

Import matplotlib.pyplot as plt

Import seaborn as sns

# Step 1: Load and preprocess the dataset

Data\_url = <https://tn.data.gov.in/catalog/marginal-workers-classified-age-industrial-category-and-sex-census-2011-india-and-states>

Df = pd.read\_csv(data\_url)

# Step 2: Data Cleaning (if needed)

# Step 3: Data Analysis

# Example: Calculate the distribution of marginal workers based on age

Age\_distribution = df[‘Age’].value\_counts()

# Step 4: Data Visualization

# Example: Create a bar chart to visualize age distribution

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

Sns.barplot(x=age\_distribution.index, y=age\_distribution.values)

Plt.title(“Age Distribution of Marginal Workers”)

Plt.xlabel(“Age”)

Plt.ylabel(“Count”)

Plt.xticks(rotation=90)

Plt.show()

# Step 5: Project Documentation (Explain the steps, objectives, and findings)

# Submission details (GitHub link, instructions, key findings)

# Save figures if needed

# plt.savefig(“age\_distribution.png”)

# End of the program

En

Explore the distribution of marginal workers across different industrial categories.

Identify the most common industries and any significant variations.

Visualization Selection

Select appropriate visualization types to effectively represent the demographic distributions:

For age and gender distribution: Bar charts, pie charts, or histograms.

For industrial category distribution: Bar charts, stacked bar charts, or heatmaps. Example program.

# Import necessary libraries

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Step 1: Load and preprocess the dataset

data\_url = "https://tn.data.gov.in/catalog/marginal-workers-classified-age-industrial-category-and-sex-census-2011-india-and-states"

df = pd.read\_csv(data\_url)

# Step 2: Data Cleaning (if needed)

# Step 3: Data Analysis

# Example: Calculate the distribution of marginal workers based on age

age\_distribution = df['Age'].value\_counts()

# Step 4: Data Visualization

# Example: Create a bar chart to visualize age distribution

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

sns.barplot(x=age\_distribution.index, y=age\_distribution.values)

plt.title("Age Distribution of Marginal Workers")

plt.xlabel("Age")

plt.ylabel("Count")

plt.xticks(rotation=90)

plt.show()

# Step 5: Project Documentation (Explain the steps, objectives, and findings)

# Submission details (GitHub link, instructions, key findings)

# Save figures if needed

# plt.savefig("age\_distribution.png")

# End of the program