Assessment of Marginal Workers in TN

Phase 3 - Project

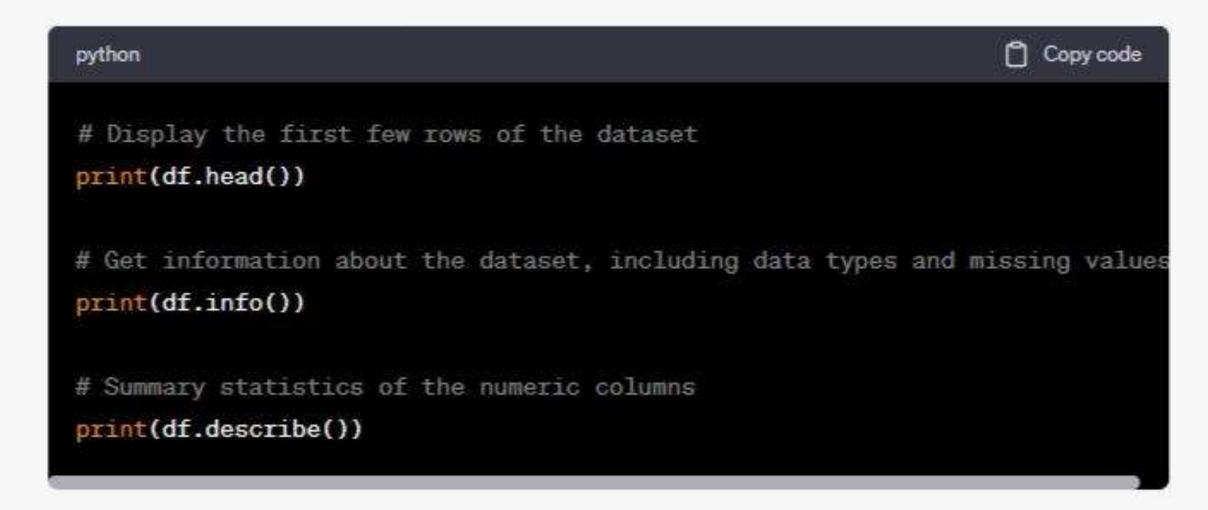
 Import Necessary Libraries: First, you need to import the necessary libraries, including Pandas, which is commonly used for data manipulation.



 Load the Dataset: Use Pandas to load your dataset. You can load data from various formats, such as CSV, Excel, JSON, etc. For example, if you have a CSV file, you can load it like this:

```
# Replace 'your_dataset.csv' with the path to your dataset file
df = pd.read_csv('your_dataset.csv')
```

 Explore the Dataset: After loading the dataset, it's a good practice to explore it. You can use various methods and attributes to do this, such as `head()`, `info()`, and `describe()`.
 For example:



Data Preprocessing: Depending on your project's requirements and the dataset's
characteristics, you may need to perform preprocessing tasks. This can include handling
missing data, encoding categorical variables, scaling/normalizing data, and more. Here are
some common preprocessing steps:

```
python
                                                                    Copy code
# Dealing with missing values (e.g., fill with mean or median)
df.fillna(df.mean(), inplace=True)
# Encoding categorical variables (if needed)
df = pd.get_dummies(df, columns=['categorical_column'])
# Scaling/normalizing numeric features (if needed)
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
df[['numeric_feature1', 'numeric_feature2']] = scaler.fit_transform(df[['num
```

```
import pandas as pd
# Load the dataset
# Replace 'your_dataset_path.csv' with the actual path to your dataset
dataset_path = 'your_dataset_path.csv'
# Load the dataset into a pandas DataFrame
df = pd.read_csv(dataset_path)
# Display the first few rows of the dataset to have an initial look
print(df.head())
```

Step 1: Import Necessary Libraries

Start by importing the required libraries, including pandas for data manipulation and other libraries as needed for visualization and analysis.

```
import pandas as pd
import numpy as np # For numerical operations
import matplotlib.pyplot as plt # For data visualization
```

Step 2: Load the Dataset

Load your dataset into a pandas DataFrame as explained in my previous response. Ensure that you replace ''your_dataset_path.csv' with the actual path to your dataset file.

```
python

dataset_path = 'your_dataset_path.csv'

df = pd.read_csv(dataset_path)
```

Step 3: Initial Data Exploration

It's essential to understand your data. Start by performing some initial data exploration tasks to get a sense of what your dataset contains:

View the first few rows of the dataset to get an idea of the data's structure.



Get information about the dataset such as data types, null values, and column names.



Descriptive statistics provide a summary of the dataset.



Step 4: Handle Missing Data

Identify and handle missing data in your dataset. Depending on the amount of missing data, you can either remove or impute missing values.

Remove rows with missing values (not recommended if there are many missing values):



. Impute missing values with the mean, median, or a specific value.

```
# Replace missing values in a specific column with the mean

df['column_name'].fillna(df['column_name'].mean(), inplace=True)
```

Step 5: Data Cleaning

Clean the data by addressing any issues with data quality or consistency. This may involve tasks like removing duplicates, correcting data types, and standardizing values.

Remove duplicates:



Step 6: Feature Engineering

If needed, create new features or transform existing ones to better represent the information in your dataset. This can include one-hot encoding categorical variables, creating new features based on existing ones, and more.

One-hot encoding categorical variables:

```
python

df = pd.get_dummies(df, columns=['categorical_column'])
```