Assessment of Marginal Workers in TamilNadu – phase2 submission document

Project: Marginal workers in Tamilnadu

**Content of project phase2:**

Consider conducting clustering analysis to identify patterns among different industrial categories and age groups.

**Data source:** Good source for categoring the age group of the workers.

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

**Cluster analysis:**

Cluster analysis is a [statistical method](https://www.qualtrics.com/experience-management/research/survey-analysis-types/) for processing data. It works by organizing items into groups – or clusters . Clustering analysis can definitely help identify patterns among different industrial categories and age groups. It can provide valuable insights into the relationships and similarities between these variables. You can use tools like Python’s scikit-learn or R’s cluster package to perform the analysis on the dataset you provided.

**Title:** Clustering Analysis for Identifying Patterns in Marginal Workers by Age and Industrial Category

**Introduction:**

* Briefly introduce the problem and the dataset.
* Explain the objective of the clustering analysis.

**Dataset Description:**

* Provide an overview of the dataset, including its source, structure, and the variables it contains.

**Data Preprocessing:**

* Describe the steps taken to prepare the data for clustering. This may include data cleaning, handling missing values, and scaling.

**Clustering Method:**

* Explain the clustering method you intend to use (e.g., K-Means, Hierarchical Clustering).
* Justify your choice of clustering method.

**Feature Selection:**

* Discuss which features (variables) from the dataset will be used for clustering.
* Explain the reasons for selecting these features.

**Clustering Analysis:**

* Present the results of the clustering analysis.
* Include visualizations, such as scatter plots or dendrogram diagrams, to illustrate the clusters.

**Interpretation:**

* Discuss the insights gained from the clustering analysis.
* Identify any patterns or trends among different industrial categories and age groups.

1. **Data Preparation**:
   * Download the dataset from the provided link.
   * Load the dataset into a data analysis tool, such as Python with libraries like Pandas.
2. **Data Cleaning**:
   * Handle missing data, if any.
   * Preprocess the data, ensuring it's in a suitable format for clustering.
3. **Feature Selection**:
   * Decide which features (columns) you want to include in your clustering analysis. In your case, this might be industrial categories and age groups.
4. **Feature Scaling**:
   * Normalize or standardize the features to ensure they have the same scale.
5. **Clustering Algorithm Selection**:
   * Choose a clustering algorithm that suits your data. Common choices include K-Means, Hierarchical Clustering, and DBSCAN.
6. **Clustering**:
   * Apply the chosen clustering algorithm to your data.
   * Determine the number of clusters (K) using methods like the elbow method or silhouette analysis.
7. **Visualization**:
   * Visualize the clusters to better understand the patterns. You can use tools like Matplotlib or Seaborn for this.
8. **Interpretation**:
   * Interpret the results and analyze the patterns that emerge within different industrial categories and age groups.
9. **Example Code**:
   * Here's an example of how to perform K-Means clustering using Python and the Scikit-Learn library:

Import pandas as pd

From sklearn.cluster import KMeans

Import matplotlib.pyplot as plt

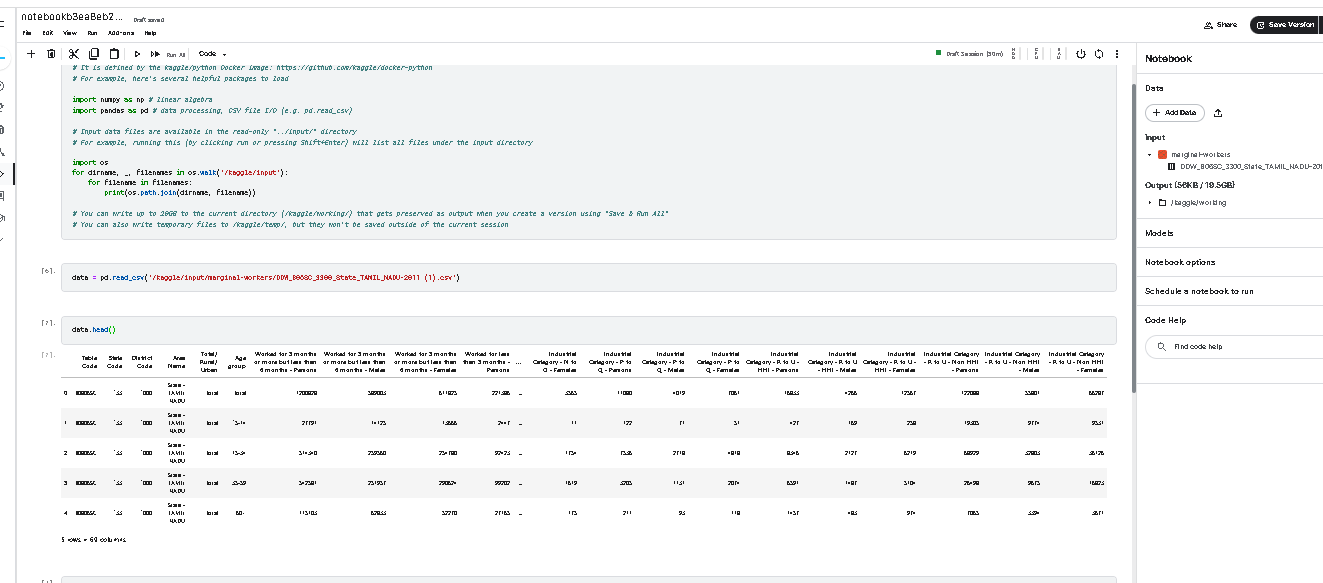
# Load the dataset

data = pd.read\_csv(“/kaggle/input/marginal-workers/DDW\_B06SC\_3300\_State\_TAMIL\_NADU-2011 .csv”)

In :

data.head()

Out :

*In:* 

*#Plotting the clusters*

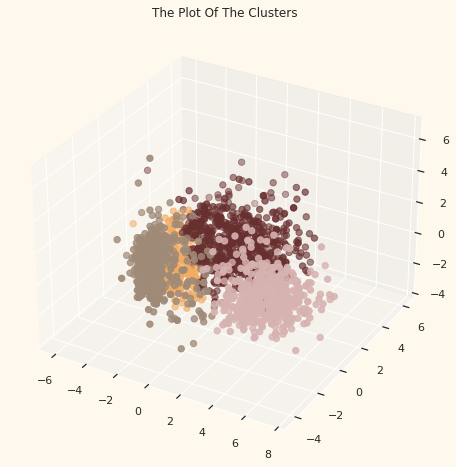
fig = plt.figure(figsize=(10,8))

ax = plt.subplot(111, projection='3d', label="bla")

ax.scatter(x, y, z, s=40, c=PCA\_ds["Clusters"], marker='o', cmap = cmap )

ax.set\_title("The Plot Of The Clusters")

plt.show()



Now one of the example program for employee dataset let’s read the data and do some exploratory data analysis to understand this dataset properly:

1

attrition = pd.read\_csv('Employee-Attrition.csv')

Usually one of the first steps in data exploration is getting a rough idea of how the features are distributed among them. To do this, I’ll use the kdeplot function in the seaborn library in Python:

|  |  |
| --- | --- |
|  | f, axes = plt.subplots(3, 3, figsize=(10, 8), |
|  | sharex=False, sharey=False) |
|  |  |
|  | # Defining our colormap scheme |
|  | s = np.linspace(0, 3, 10) |
|  | cmap = sns.cubehelix\_palette(start=0.0, light=1, as\_cmap=True) |
|  |  |
|  | # Generate and plot |
|  | x = attrition['Age'].values |
|  | y = attrition['TotalWorkingYears'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, cut=5, ax=axes[0,0]) |
|  | axes[0,0].set( title = 'Age against Total working years') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=0.333333333333, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['Age'].values |
|  | y = attrition['DailyRate'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[0,1]) |
|  | axes[0,1].set( title = 'Age against Daily Rate') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=0.666666666667, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['YearsInCurrentRole'].values |
|  | y = attrition['Age'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[0,2]) |
|  | axes[0,2].set( title = 'Years in role against Age') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=1.0, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['DailyRate'].values |
|  | y = attrition['DistanceFromHome'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[1,0]) |
|  | axes[1,0].set( title = 'Daily Rate against DistancefromHome') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=1.333333333333, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['DailyRate'].values |
|  | y = attrition['JobSatisfaction'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[1,1]) |
|  | axes[1,1].set( title = 'Daily Rate against Job satisfaction') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=1.666666666667, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['YearsAtCompany'].values |
|  | y = attrition['JobSatisfaction'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[1,2]) |
|  | axes[1,2].set( title = 'Daily Rate against distance') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=2.0, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['YearsAtCompany'].values |
|  | y = attrition['DailyRate'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[2,0]) |
|  | axes[2,0].set( title = 'Years at company against Daily Rate') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=2.333333333333, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['RelationshipSatisfaction'].values |
|  | y = attrition['YearsWithCurrManager'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[2,1]) |
|  | axes[2,1].set( title = 'Relationship Satisfaction vs years with manager') |
|  |  |
|  | cmap = sns.cubehelix\_palette(start=2.666666666667, light=1, as\_cmap=True) |
|  | # Generate and plot |
|  | x = attrition['WorkLifeBalance'].values |
|  | y = attrition['JobSatisfaction'].values |
|  | sns.kdeplot(x, y, cmap=cmap, shade=True, ax=axes[2,2]) |
|  | axes[2,2].set( title = 'WorklifeBalance against Satisfaction') |
|  |  |
|  | f.tight\_layout()  out:  https://i0.wp.com/thecleverprogrammer.com/wp-content/uploads/2020/11/1-employee-attrition.png?w=734&ssl=1 |

*Conclusion (Phase 2):*

*Project Conclusion:*

* In the Phase 2 conclusion, we will summarize the key findings and insights from the cluster analysis.*