# INTRODUCTION MARGINAL-WORKERS-CLASSIFIED-AGE-INDUSTRIAL

# INTRODUCTION

This table gives the break-up of the population by their economic activity status as 'main workers', 'marginal workers', 'non-workers' and 'marginal and non-worker' seeking/available for work cross classified with educational level and sex. This table gives the data for India/States/ UTs./Districts and City. This table is separate for SCs upto District level. It allows organizations to make informed decisions related to inventory levels, procurement, pricing, and marketing strategies.

# PROBLEM STATEMENT

The problem is to implement data science techniques to MARGINAL-WORKERS based on their behavior, preferences, and demographic attributes. This project involves data collection, data preprocessing, feature engineering, clustering algorithms, visualization, and interpretation of results.

MARGINAL-WORKERS is the process of dividing a company's customers into groups based on common characteristics so companies can market to each group effectively and appropriately.

# PREREQUISITES FOR BUILDING A MARGINAL-wORKERS-CLASSIFIED-AGE-INDUSTRIAL-MODEL.

* The data is obtained from <https://tn.data.gov.in/catalog/marginal-workers-classified-age-industrial-category-and-sex-census-2011-india-and-states>

# TOOLS & SOFTWARE USED

* Programming language
* Machine Learning Libraries
* Integrated Development Environment
* Data Visualization tools
* Data Preprocessing tools
* Data Collection & Storage
* Notebooks & Documentation
* External Data Source
* Data Labelling tools
* Version tools

# PROCESS OVERVIEW

1. Design thinking
2. Design into innovation
3. Build Load and Preprocessing of dataset
4. Additional Evaluation

# DESIGN THINKING AND PRESENT IN FORMOF DOCUMENT

## Empathize:

* Understand the annual income and spending score of the customers.
* Conduct interviews and surveys to gather insights on customers and what information is used for critical thinking.

## Define:

* Clearly articulate the problem statement and identify the key goals and success criteria for the project and improve customer’s trust in the evaluation process

## Ideate:

* Creative solutions and data sources that can enhance the transparency of a MARGINAL-wORKERS.
* Encourage interdisciplinary collaboration to generate a wide range of ideas, including the use of alternative data, new algorithms, or improved visualization techniques.

## Prototype:

* Create prototype machine learning models based on the ideas generated during the ideation phase.
* Test and iterate on these prototypes to determine which approaches are most promising in terms of accuracy and usability.

## Test & Train:

* There are many different machine learning algorithms that can be used for house price prediction. Some popular choices include linear regression, random forests, and gradient boosting machines.

## Implement:

* Implement transparency measures, such as model interpretability tools, to ensure customers understand how too A MARGINAL-WORKERS are generated.

## Evaluate:

* Continuously monitor the performance of the machine learning model after implementation to ensure it remains accurate and relevant.
* Gather feedback and insights from customers to identify improvement.

## Iterate:

* Apply an iterative approach to refine the machine learning model based on ongoing feedback and changing user needs.

Before starting any data science project, it is vital to explore the dataset and understand each variable.

* 1. BUILD **LOADING AND PREPROCESSING THE DATASET**

## Dataset Loading:

* Import relevant libraries, such as pandas for data manipulation and numpy for numerical operations. Load the dataset into a pandas Data Frame for easy data handling. You can use pd.read\_csv() for CSV files or other appropriate functions for different file formats.

## df=pd.read\_csv(https://tn.data.gov.in/catalog/marginal-workers-classified-age-industrial-category-and-sex-census-2011-india-and-states)

## Data Exploration:

* Explore the dataset to understand its structure and contents. Check for the presence of missing values, outliers, and data types of each feature.

## Data Cleaning:

* Handle missing values by either removing rows with missing data or imputing values based on the nature of the data.

## Data Encoding:

* Convert categorical variables into numerical format using techniques like one-hot encoding.

## Features Selection:

* Use techniques like feature importance scores or recursive feature elimination to identify the most relevant features.

## Data Preprocessing:

* The raw data we downloaded is complex and in a format that cannot be easily ingested by customer MARGINAL-WORKERS models. We need to do some preliminary data preparation to make this data interpretable.

## Model Building:

* We are going to create a K-Means clustering algorithm to perform MARGINAL-WORKERS . The goal of a K-Means clustering model is to segment all the data available into non-overlapping sub-groups that are distinct from each other.
* These includes various aspects like Cluster of customers, Kmeans clusters, dropping, counterplotting, scatterplotting, describing the values.

## Missing Values:

* By using isnull() or notnull() we can identify the missing values in the data.

# (3) ADDITIONAL EVALUATION

## Evaluation Process:

* This includes various aspects starting from the preparation to deployment.

## Data Preparation:

* This includes cleaning the data, removing outliers, and handling missing values.

## Feature selection:

* This can be done using a variety of methods, such as correlation analysis, information gain, and recursive feature elimination.

## Model Training:

* There are many different machine learning algorithms that can be used for house price prediction. Some popular choices include linear regression, random forests, and gradient boosting machines.
* Model Training is the process of teaching a machine learning model to do customer segmentation. It involves feeding the model historical data and features. The model then learns the relationships between these customers and spending scores.

## Model Evaluation:

* This can be done by calculating the mean squared error (MSE) or the root mean squared error (RMSE) of the model's predictions on the held-out test set.
* Model Evaluation is the process of assessing the performance of a machine learning model on unseen data. This is important to ensure that the model will generalize to the new data.

## Model Deployment:

* Once the model has been evaluated and found to be performing well, it can be deployed to production.

## Model Comparison:

* Model Comparison is the tool enhanced to provide systematic representation on the relationship between annual income and spending score.

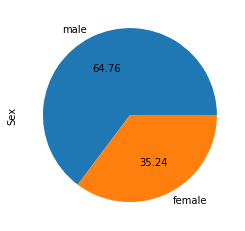
## Model Analysis:

* Once evaluation of model gets completes, the analysis of model’s predictions can be started to identify any patterns or biases. This will help us to understand the strengths and weaknesses of the model and to improve it.
* **Pie Chart.**

The pie chart is also the same as the countplot, only gives you additional information about the percentage presence of each category in data means which category is getting how much weightage in data. let us check about the Sex column, what is a percentage of Male and Female members traveling.

data['Sex'].value\_counts().plot(kind="pie", autopct="%.2f")

plt.show()



#### Histogram

A histogram is a value distribution plot of numerical columns. It basically creates bins in various ranges in values and plots it where we can visualize how values are distributed. We can have a look where more values lie like in positive, negative, or at the center(mean). Let’s have a look at the Age column.

plt.hist(**data**['Age'], bins=5)

plt.show()

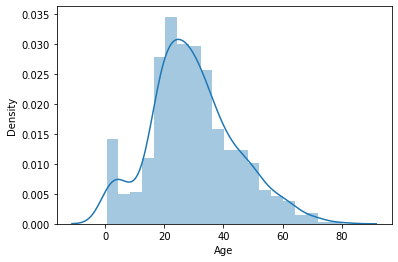


#### Distplot.

Distplot is also known as the second Histogram because it is a slight improvement version of the Histogram. Distplot gives us a KDE(Kernel Density Estimation) over histogram which explains PDF(Probability Density Function) which means what is the probability of each value occurring in this column. If you have study statistics before then definitely you should know about PDF function.

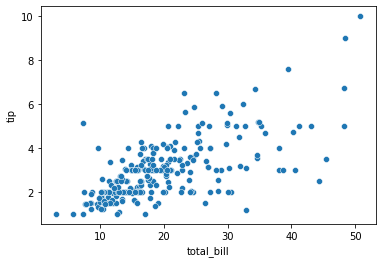
sns.distplot(data['Age'])

plt.show()



#### Scatter Plot

To plot the relationship between two numerical variables scatter plot is a simple plot to do. Let us see the relationship between the total bill and tip provided using a scatter plot.

sns.scatterplot(tips["total\_bill"], tips["tip"])

#### Multivariate analysis with scatter plot.

we can also plot 3 variable or 4 variable relationships with scatter plot. suppose we want to find the separate ratio of male and female with total bill and tip provided.

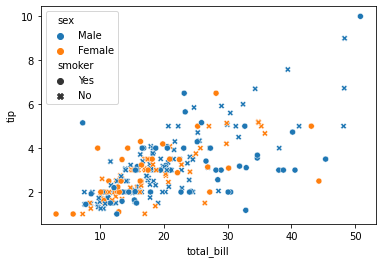
sns.scatterplot(tips["total\_bill"], tips["tip"], hue=tips["sex"])

plt.show()



sns.scatterplot(tips["total\_bill"], tips["tip"], hue=tips["sex"], style=tips['smoker'])

plt.show()

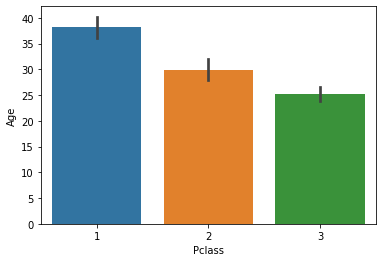


#### Bar Plot.

Bar plot is a simple plot which we can use to plot categorical variable on the x-axis and numerical variable on y-axis and explore the relationship between both variables. The blacktip on top of each bar shows the confidence Interval. let us explore P-Class with age.

sns.barplot(data['Pclass'], data['Age'])

plt.show()

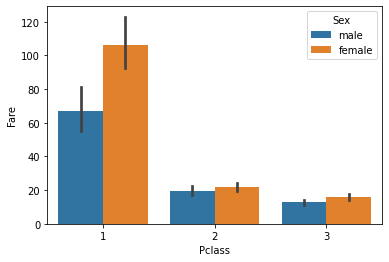


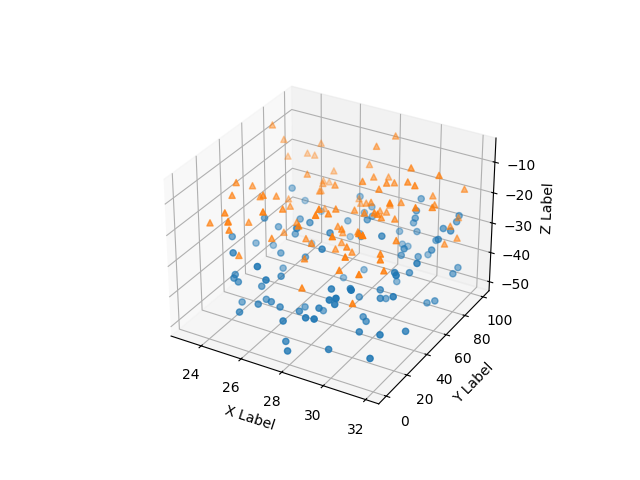
#### Multivariate analysis using Bar plot.

Hue’s argument is very useful which helps to analyze more than 2 variables. Now along with the above relationship we want to see with gender.

sns.barplot(**data**['Pclass'], **data**['Fare'], hue = **data**["Sex"])

plt.show()

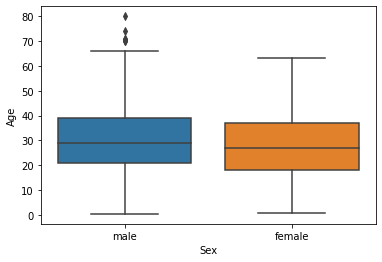


* 3D plot:
* 

#### Boxplot.

We have already study about boxplots in the Univariate analysis above. we can draw a separate boxplot for both the variable. let us explore gender with age using a boxplot.

sns.boxplot(data['Sex'], data["Age"])

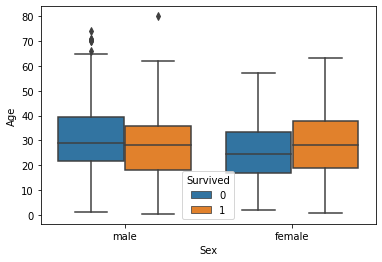


#### Multivariate analysis with boxplot.

Along with age and gender let’s see who has survived and who has not.

sns.boxplot(data['Sex'], data["Age"], data["Survived"])

plt.show()



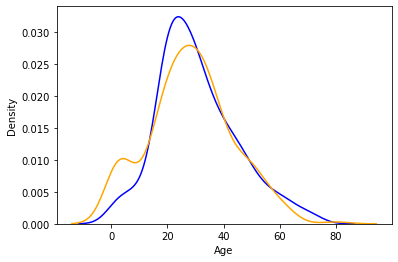
#### Distplot.

Distplot explains the PDF function using kernel density estimation. Distplot does not have a hue parameter but we can create it. suppose we want to see the probability of people with an age range that of survival probability and find out whose survival probability is high to the age range of death ratio.

sns.distplot(**data**[**data**['Survived'] == 0]['Age'], hist=False, color="blue")

sns.distplot(**data**[**data**['Survived'] == 1]['Age'], hist=False, color="orange")

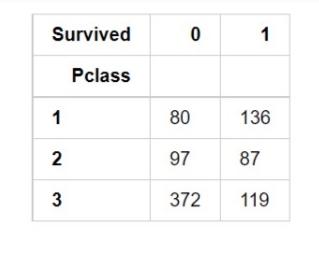
plt.show()



### Heatmap.

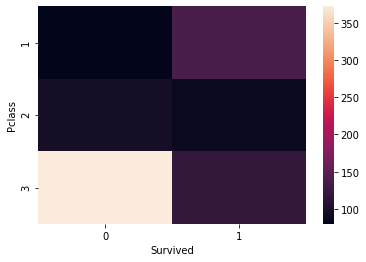
If you have ever used a crosstab function of pandas then Heatmap is a similar visual representation of that only. It basically shows that how much presence of one category concerning another category is present in the dataset. let me show first with crosstab and then with heatmap.

pd.crosstab(data['Pclass'], data['Survived'])



Now with heatmap, we have to find how many people survived and died.

sns.heatmap(pd.crosstab(data['Pclass'], data['Survived']))

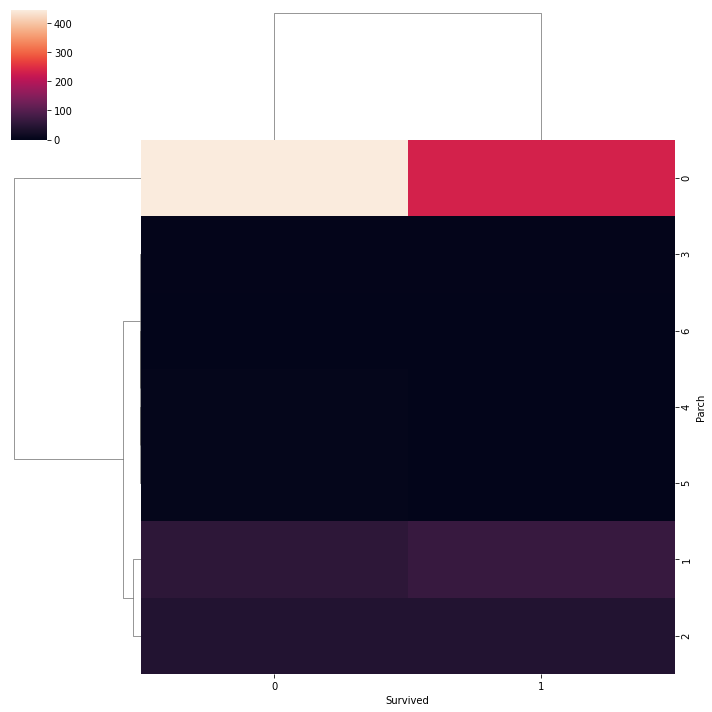


#### Cluster map.

we can also use a cluster map to understand the relationship between two categorical variables. A cluster map basically plots a dendrogram that shows the categories of similar behavior together.

sns.clustermap(pd.crosstab(data['Parch'], data['Survived']))

plt.show()



* **data code.**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset

data = pd.read\_csv('marginal\_workers\_data.csv')

# Data preprocessing (e.g., handle missing values)

# Descriptive statistics

age\_mean = data['age'].mean()

gender\_distribution = data['gender'].value\_counts()

education\_distribution = data['education'].value\_counts()

# Data visualization

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

plt.subplot(2, 2, 1)

sns.histplot(data['age'], kde=True)

plt.title('Age Distribution')

plt.subplot(2, 2, 2)

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

plt.title('Gender Distribution')

plt.subplot(2, 2, 3)

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

plt.title('Education Distribution')

plt.subplot(2, 2, 4)

sns.countplot(data=data, x='employment\_status', hue='gender')

plt.title('Employment Status by Gender')

plt.tight\_layout()

plt.show()

# Interpretation: Provide insights based on the visualizations

print(f"Average age of marginal workers: {age\_mean}")

print("Gender Distribution:")

print(gender\_distribution)

print("Education Distribution:")

print(education\_distribution)

# (4) DATA FLOW OF CUSTOMER MODEL.

* Physical Flow :

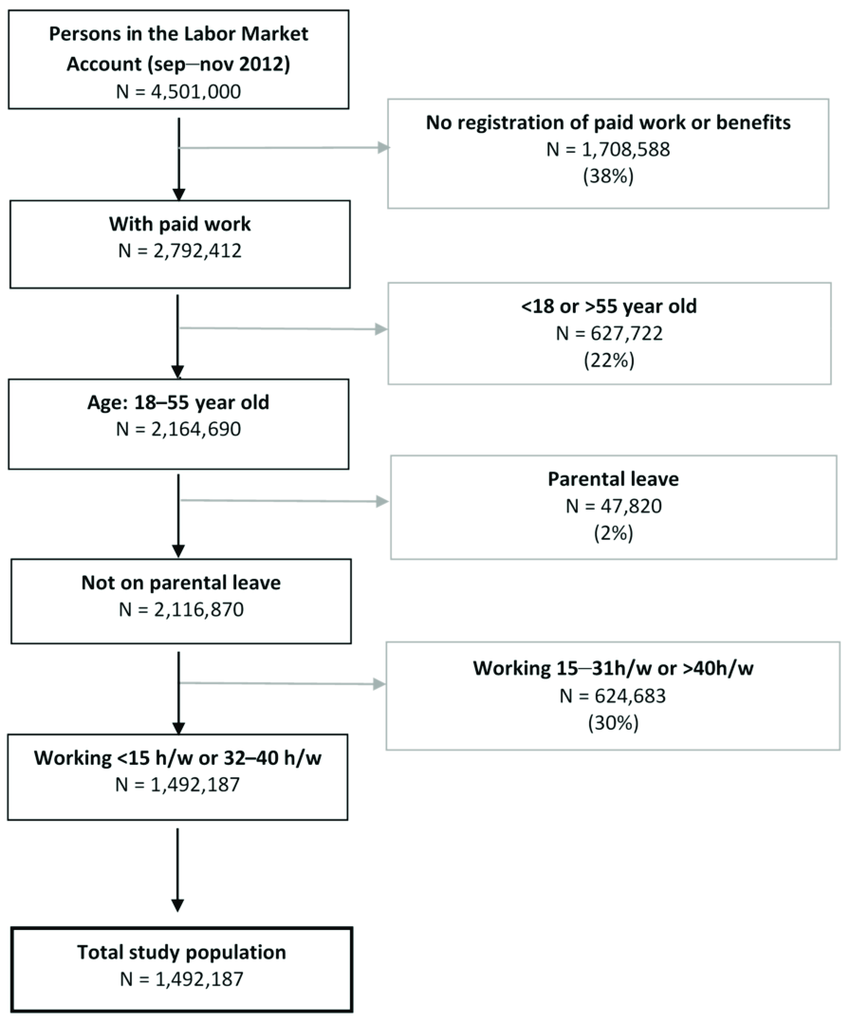
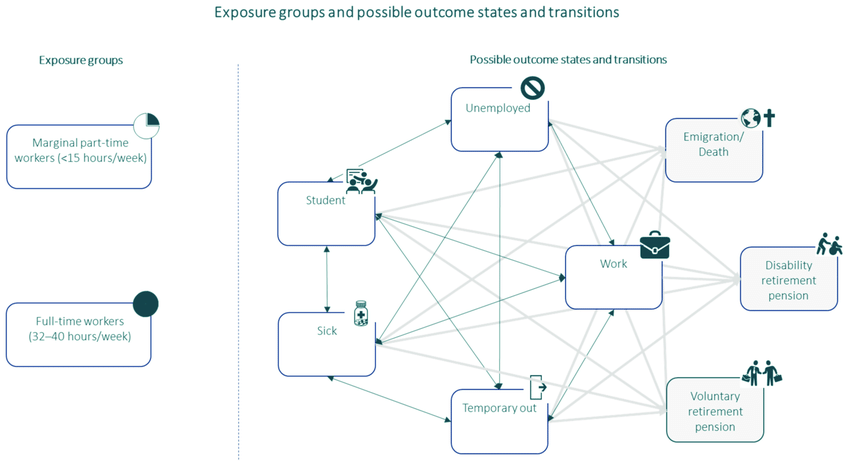


Figure 1

* Logical Flow:



THANK YOU