Data Collection and Preprocessing Phase

Date	June 2024
Team ID	739765
Project Title	Occupancy Rates and Demand in the Hospitality Industry
Maximum Marks	6 Marks

Preparation Template

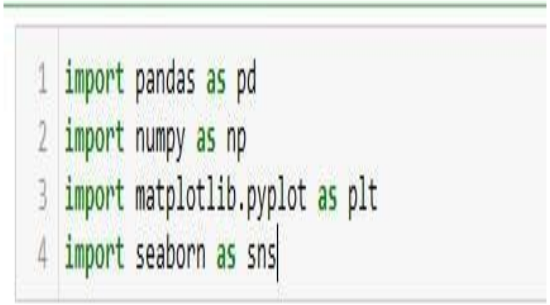
The images will be preprocessed by resizing, normalizing, augmenting, denoising, adjusting contrast, detecting edges, converting color space, cropping, batch normalizing, and whitening data. These steps will enhance data quality, promote model generalization, and improve convergence during neural network training, ensuring robust and efficient performance across various computer vision tasks.

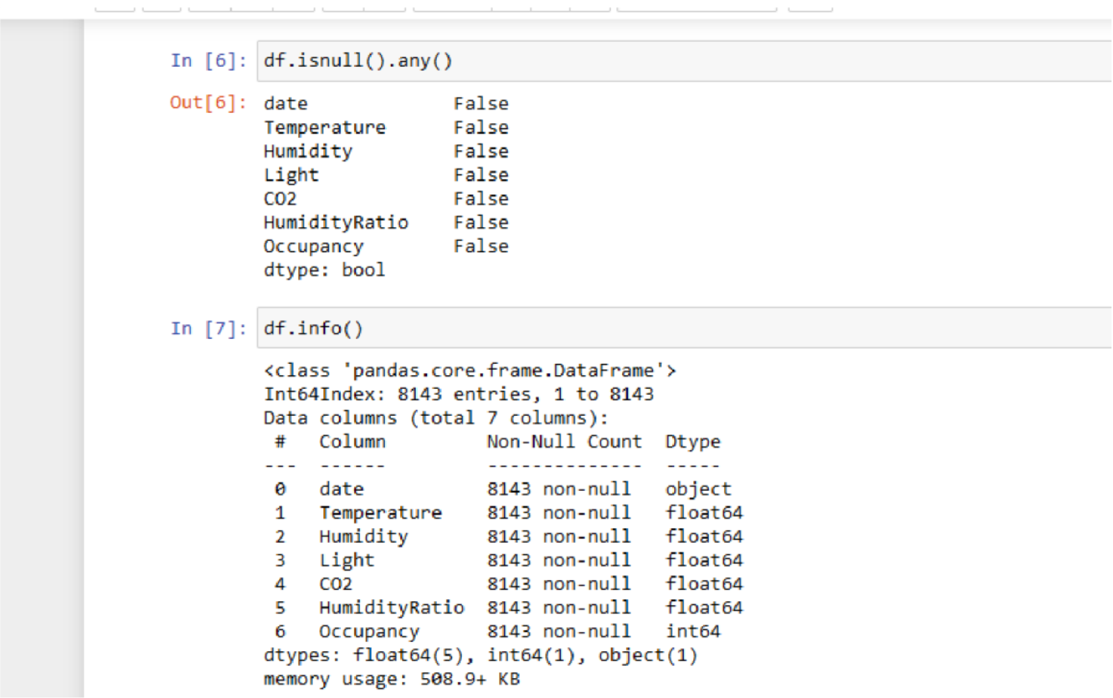
	Description

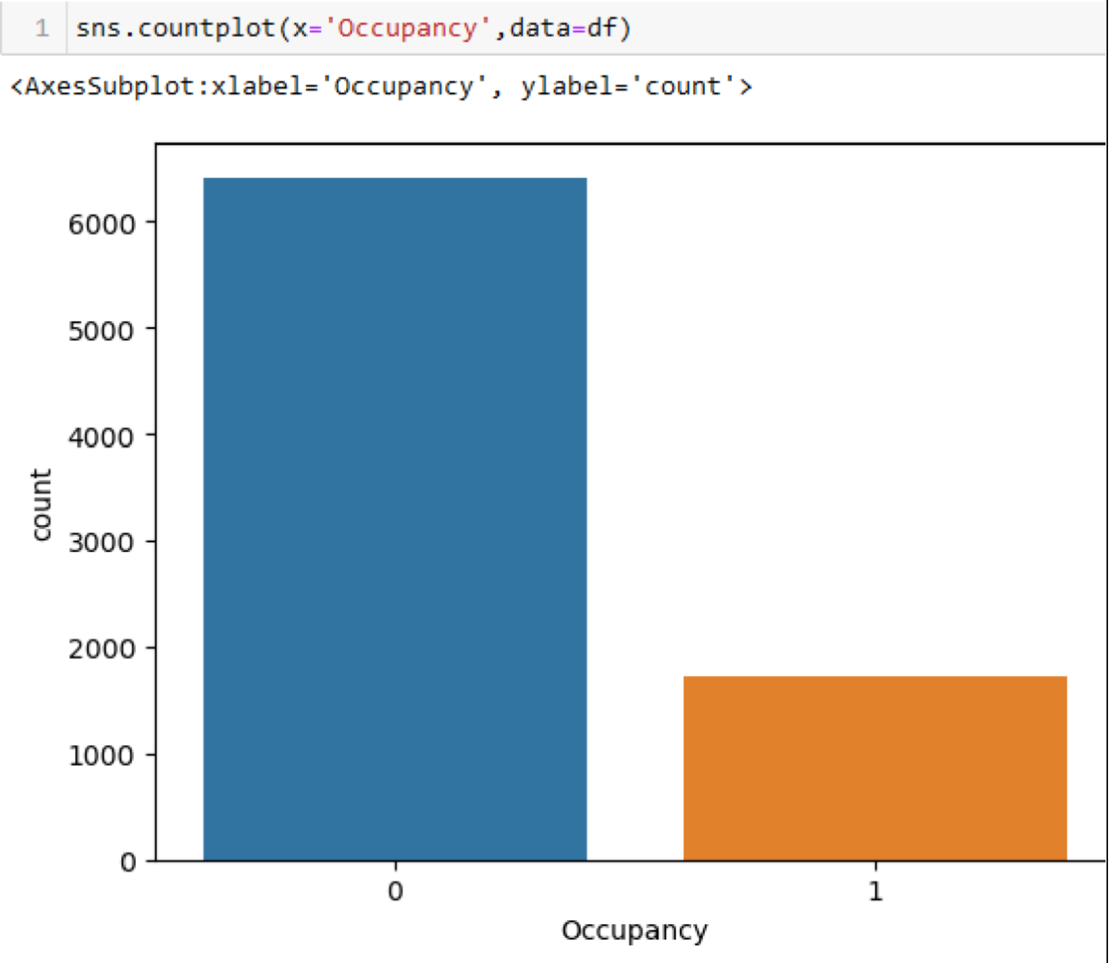
Section	
Data Overview	There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc. In this project we have used .csv data.
Data Preparation	These are the general steps of pre-processing the data before using it for machine learning
Handling missing values	We use Handling missing values For checking the null values
Handling categorical data	As we can see our dataset has categorical data we must convert the categorical data to integer encoding or binary encoding
Handling Outliers in Data	With the help of boxplot, outliers are visualized. And here we are going to find upper bound and lower bound of numerical features with some mathematical formula.

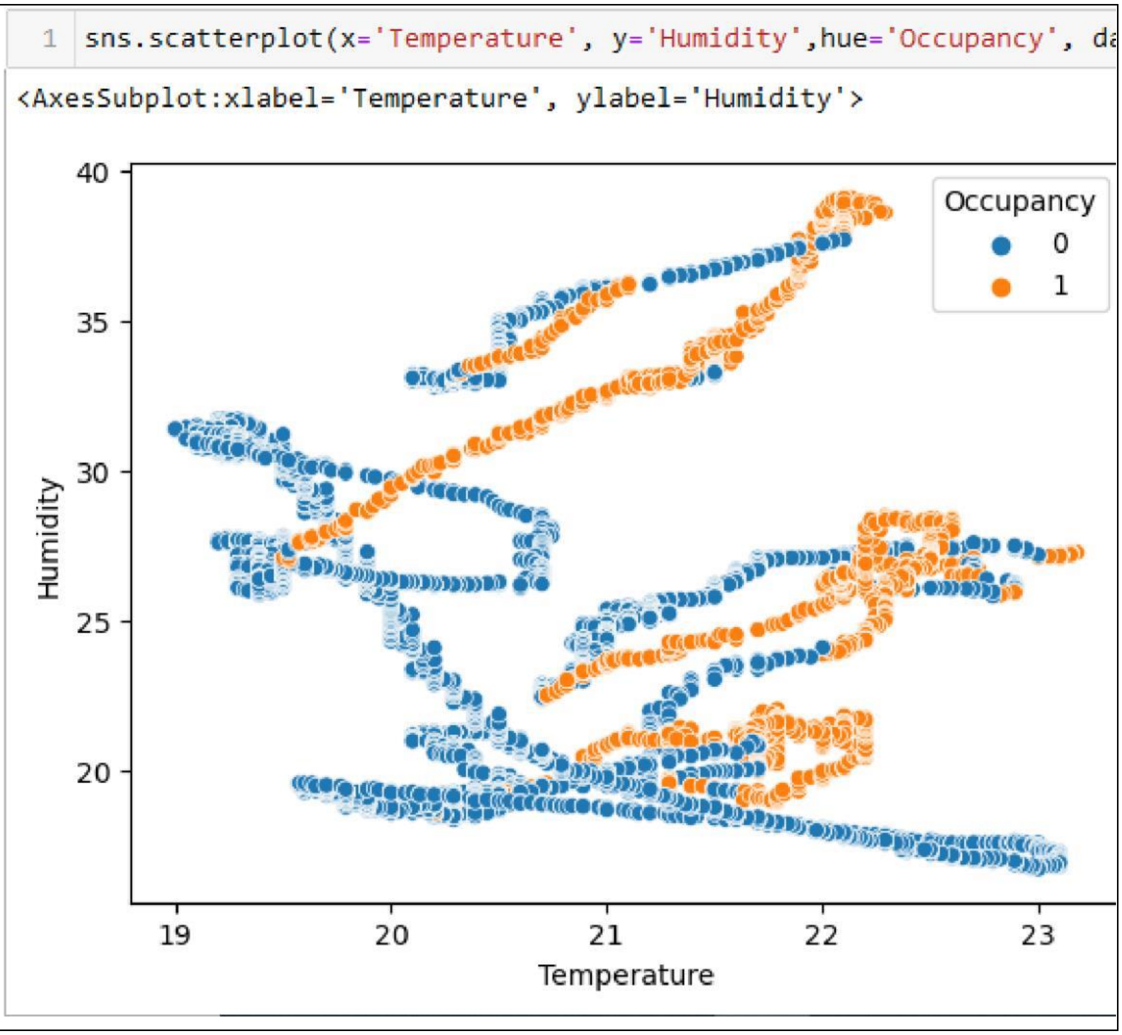
Preparation

Collect the dataset	Please refer to the link given below to download the dataset. https://www.kaggle.com/datasets/robmarkcole/occupancy-detection-data-set-uci https://www.kaggle.com/code/turksoyomer/hvac-occupancy-detection-with-mlanddl-methods

Importing the libraries	 import pandas as pd import numpy as np import matplotlib.pyplot as plt import seaborn as sns
Loading Data	We use the code Data =pd.read_csv(‘datatraining.csv’) For reading the dataset

Handling missing values	 In [6]: df.isnull().any() Out[6]: date False Temperature False Humidity False Light False CO2 False HumidityRatio False Occupancy False dtype: bool In [7]: df.info() <class 'pandas.core.frame.DataFrame'> Int64Index: 8143 entries, 1 to 8143 Data columns (total 7 columns): # Column Non-Null Count Dtype --- ------ -------------- ----- 0 date 8143 non-null object 1 Temperature 8143 non-null float64 2 Humidity 8143 non-null float64 3 Light 8143 non-null float64 4 CO2 8143 non-null float64 5 HumidityRatio 8143 non-null float64 6 Occupancy 8143 non-null int64 dtypes: float64(5), int64(1), object(1) memory usage: 508.9+ KB

Handling Outliers	
	`sns.countplot(x='Occupancy',data=df)`
	`<AxesSubplot:xlabel='Occupancy', ylabel='count'>`
	 count
	6000
	5000
	4000
	3000
	2000
	1000
	0
	0
	1
	Occupancy



```
sns.scatterplot(x='Temperature', y='Humidity',hue='Occupancy', da
```

```
<AxesSubplot:xlabel='Temperature', ylabel='Humidity'>
```