This Project is focusing on advertising data in which we are using ML Logistics Regression for indicating whether or not particular internet user click on advertise. We will create a model in which we see whether or not user click on ad based on the age, ad and user features.

Importing Libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

%matplotlib inline

import seaborn as sns

Importing Advertising Data

ad\_data = pd.read\_csv('advertising.csv')

ad\_data.head()

ad\_data.info()

ad\_data.describe()

Data Analysis on Data

#Histplot for the age being visiting the advertise

sns.histplot(ad\_data['Age'])

#Jointplot for Age and income in particular Area

sns.jointplot(x='Age',y='Area Income',data=ad\_data)

#Jointplot for Age and time spent on site for calculating the age being visiting in website

sns.jointplot(x='Age',y='Daily Time Spent on Site',data=ad\_data,kind='kde',color='red')

sns.jointplot(x='Daily Time Spent on Site',y='Daily Internet Usage',data=ad\_data,color='green')

#Paiplot for visitors clicking on ad

sns.pairplot(ad\_data,hue='Clicked on Ad')

ML Libraries

from sklearn.model\_selection import train\_test\_split

Logistics Regression

from sklearn.linear\_model import LogisticRegression

logmodel = LogisticRegression()

ad\_data.columns

X = ad\_data[['Daily Time Spent on Site', 'Age', 'Area Income',

'Daily Internet Usage','Male']]

y = ad\_data['Clicked on Ad']

Train Test Split on Data for testing and training and fitting the model

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=101)

logmodel.fit(X\_train,y\_train)

Predictions for Testing Data

predictions = logmodel.predict(X\_test)

from sklearn.metrics import classification\_report, confusion\_matrix

Printing the classification report and confusion matrix for advertising data

print(classification\_report(y\_test, predictions))

print(confusion\_matrix(y\_test, predictions))