In [5]: ad\_data.info() <class 'pandas.core.frame.DataFrame'> RangeIndex: 1000 entries, 0 to 999 Data columns (total 10 columns): # Column Non-Null Count Dtype \_\_\_\_\_ ---O Daily Time Spent on Site 1000 non-null float64 1 Age 1000 non-null int64 2 Area Income 1000 non-null float64 1000 non-null float64 3 Daily Internet Usage 1000 non-null float64 4 Ad Topic Line 1000 non-null object
5 City 1000 non-null object
6 Male 1000 non-null int64
7 Country 1000 non-null object
8 Timestamp 1000 non-null object
9 Clicked on Ad 1000 non-null int64 dtypes: float64(3), int64(3), object(4) memory usage: 78.2+ KB In [6]: ad\_data.describe() Out[6]: **Daily Time Spent on Site** Age Area Income Daily Internet Usage Male Clicked on Ad 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000 1000.00000 count 65.000200 36.009000 55000.000080 180.000100 0.481000 0.50000 mean 15.853615 8.785562 13414.634022 43.902339 0.499889 0.50025 std 32.600000 19.000000 13996.500000 104.780000 0.000000 0.00000 min 25% 51.360000 29.000000 47031.802500 138.830000 0.000000 0.00000 **50**% 68.215000 35.000000 57012.300000 183.130000 0.000000 0.50000 **75**% 42.000000 65470.635000 1.000000 1.00000 78.547500 218.792500 91.430000 61.000000 79484.800000 269.960000 1.000000 1.00000 max In [12]: sns.distplot(ad\_data["Age"], kde=False, bins=30) Out[12]: <AxesSubplot:xlabel='Age'> 80 60 40 -Age In [14]: sns.jointplot(y="Area Income", x="Age", data=ad\_data) <seaborn.axisgrid.JointGrid at 0x1e8eadcf130> 80000 70000 60000 50000 30000 20000 20 In [18]: sns.jointplot(data=ad\_data,y="Daily Time Spent on Site",x="Age",kind="kde",color="red") Out[18]: <seaborn.axisgrid.JointGrid at 0x1e8ec50bee0> 100 90 -70 60 50 40 30 -20 60 20 50 In [20]: sns.jointplot(y="Daily Time Spent on Site", x="Daily Internet Usage", data=ad\_data, color="green") <seaborn.axisgrid.JointGrid at 0x1e8ec9a5eb0> Out[20]: 90 40 150 175 200 225 250 Daily Internet Usage In [22]: sns.pairplot(data=ad\_data,hue="Clicked on Ad") Out[22]: <seaborn.axisgrid.PairGrid at 0x1e8f15237f0> 90 Site 80 e Spent o <u>=</u> 50 Daily 40 60 20 80000 70000 60000 50000 Clicked on Ad 40000 30000 20000 275 250 225 귤 200 혈 175 150 125 0.8 0.6 0.2 0.0

City Male

Wrightburgh

West Jodi

Davidton

Country

Tunisia 2016-03-27 00:53:11

Nauru 2016-04-04 01:39:02

Italy 2016-01-10 02:31:19

Iceland 2016-06-03 03:36:18

0 San Marino 2016-03-13 20:35:42

Timestamp Clicked on Ad

0

**Ad Topic Line** 

Robust logistical utilization South Manuel

Cloned 5thgeneration orchestration

Monitored national standardization

Organic bottom-line service-desk

245.89 Triple-buffered reciprocal time-frame West Terrifurt

0.86 0.96 0.91 162 0.96 0.85 0.90 168 0.91 330 accuracy 0.91 0.91 0.91 330 macro avg 0.91 0.91 330 weighted avg 0.91

precision recall f1-score support

40 60 80 100

68.95 35

80.23 31

74.15 29

68.37 35

In [40]: from sklearn.model\_selection import train\_test\_split

In [42]: **from** sklearn.linear\_model **import** LogisticRegression

In [46]: from sklearn.metrics import classification\_report

In [48]: print(classification\_report(y\_test,predictions))

26

69.47

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

In [43]: lm=LogisticRegression()

In [44]: lm.fit(X\_train,y\_train)

LogisticRegression()

In [45]: predictions=lm.predict(X test)

Daily Time Spent on Site Age Area Income Daily Internet Usage

61833.90

68441.85

59785.94

73889.99

Daily Time Spent on Site

In [32]: ad\_data.head()

2

4

Out[32]:

Out[44]:

40

20000400006000080000

Area Income

236.50

225.58

In [39]: X=ad\_data[['Daily Time Spent on Site', 'Age', 'Area Income', 'Daily Internet Usage', 'Male']]

In [41]: X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.33, random\_state=42)

100 150 200 250 300-0.5 0.0

Wrightburgh

West Jodi

Davidton

City Male

Daily Internet Usage

**Ad Topic Line** 

Robust logistical utilization South Manuel

Cloned 5thgeneration orchestration

Monitored national standardization

Organic bottom-line service-desk

245.89 Triple-buffered reciprocal time-frame West Terrifurt

0.5 1.0 1.5

Tunisia 2016-03-27 00:53:11

Nauru 2016-04-04 01:39:02

Italy 2016-01-10 02:31:19

Iceland 2016-06-03 03:36:18

0 San Marino 2016-03-13 20:35:42

Country

Timestamp Clicked on Ad

In [1]: import numpy as np

import pandas as pd

%matplotlib inline
import seaborn as sns

ad data.head()

Out[4]:

2

3

import matplotlib.pyplot as plt

In [4]: ad\_data=pd.read\_csv("advertising.csv")

Daily Time Spent on Site Age Area Income Daily Internet Usage

61833.90

68441.85

59785.94

54806.18

73889.99

256.09

193.77

236.50

225.58

68.95 35

80.23 31

69.47 26

74.15 29

68.37 35