In [1]: **import** numpy **as** np **import** pandas **as** pd

In [2]: df **=** pd**.**read\_csv("C:\\Users\\Hp\\Desktop\\New folder (2)\\click.csv") df**.**head()

Out[2]: **Daily**

**Daily**

**Time Area Clicked**

**Age Internet Ad Topic Line City Gender Country Timestamp**

**Spent on Income on Ad**

**Usage**

**Site**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 62.26 | 32.0 | 69481.85 | 172.83 | Decentralized real-time circuit | Lisafort | Male | Svalbard &  Jan Mayen Islands | 2016-06-09 21:43:05 | 0 |
| **1** | | 41.73 | 31.0 | 61840.26 | 207.17 | Optional fullrange projection | West  Angelabury | Male | Singapore | 2016-01-16 17:56:05 | 0 | |
| **2** | | 44.40 | 30.0 | 57877.15 | 172.83 | Total  5thgeneration standardization | Reyesfurt | Female | Guadeloupe | 2016-06-29 10:50:45 | 0 | |
| **3** | | 59.88 | 28.0 | 56180.93 | 207.17 | Balanced empowering success | New  Michael | Female | Zambia | 2016-06-21 14:32:32 | 0 | |
| **4** | | 49.21 | 30.0 | 54324.73 | 201.58 | Total  5thgeneration standardization | West  Richard | Female | Qatar | 2016-07-21 10:54:35 | 1 | |

In [3]: **import** plotly.graph\_objects **as** go **import** plotly.express **as** px **import** plotly.io **as** pio

pio**.**templates**.**default **=** "plotly\_white"

|  |
| --- |
| df['Clicked on Ad'] **=**df['Clicked on Ad']**.**map({0 :'No', *# Map se replace kiya click*  1:'Yes'}) df**.**head() |

In [4]:

Out[4]: **Daily**

**Daily**

**Time Area Clicked**

**Age Internet Ad Topic Line City Gender Country Timestamp**

**Spent on Income on Ad**

**Usage**

**Site**

Svalbard &

Decentralized 2016-06-09

**0** 62.26 32.0 69481.85 172.83 Lisafort Male Jan Mayen No real-time circuit 21:43:05

Islands

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | 41.73 | 31.0 | 61840.26 | 207.17 | Optional fullrange projection | West  Angelabury | Male | Singapore | 2016-01-16 17:56:05 | No |
| **2** | 44.40 | 30.0 | 57877.15 | 172.83 | Total  5thgeneration standardization | Reyesfurt | Female | Guadeloupe | 2016-06-29 10:50:45 | No |
| **3** | 59.88 | 28.0 | 56180.93 | 207.17 | Balanced empowering success | New  Michael | Female | Zambia | 2016-06-21 14:32:32 | No |
| **4** | 49.21 | 30.0 | 54324.73 | 201.58 | Total  5thgeneration standardization | West  Richard | Female | Qatar | 2016-07-21 10:54:35 | Yes |

Now lets analyze the clickt-hrough rate based on the time spent bythe users on the website

|  |
| --- |
| fig **=** px**.**box(df, x **=**"Daily Time Spent on Site", color **=**"Clicked on Ad", title **=** "Clickt-Through Rate based on The Time Spent on Site ", color\_discrete\_map **=** {'Yes' :'blue', 'No':'red'}) fig**.**show() |

In [5]:

# Clickt-Through Rate based on The Time Spent on Site

Now lets analyze the clickt-hrough rate based on the Daily Internet Usage of the users

|  |
| --- |
| fig **=** px**.**box(df, x **=**"Daily Internet Usage", color **=**"Clicked on Ad", title **=** "Clickt-Through Rate based on The Daily Internet Usage ", color\_discrete\_map **=** {'Yes' :'blue', 'No':'red'}) fig**.**show() |

In [6]:

# Clickt-Through Rate based on The Daily Internet Usage

Now lets analyze the clickt-hrough rate based on the age of the users

|  |
| --- |
| fig **=** px**.**box(df, x **=**"Age",  color **=**"Clicked on Ad", title **=** "Clickt-Through Rate based on Age ", color\_discrete\_map **=** {'Yes' :'black', 'No':'red'}) fig**.**show() |

In [7]:

# Clickt-Through Rate based on Age

Now lets analyze the clickt-hrough rate based on the Income of the users

|  |
| --- |
| fig **=** px**.**box(df, x **=**"Area Income", color **=**"Clicked on Ad", title **=** "Clickt-Through Rate based on The Income ", color\_discrete\_map **=** {'Yes' :'blue',  'No':'red'}) fig**.**show() |

In [8]:

|  |
| --- |
| df['Clicked on Ad']**.**value\_counts() *# valuecount ek column ki jitni bhi subcategory hai u* |

# Clickt-Through Rate based on The Income

Calculating click Through rate of Ads

In [9]:

Out[9]: No 5083 Yes 4917

Name: Clicked on Ad, dtype: int64

In [10]: df**.**shape

Out[10]: (10000, 10)

In [11]: click\_through\_rate\_no **=** 5083**/**10000 *#ye humne value nikalni hai*  click\_through\_rate\_no

Out[11]: 0.5083

In [12]: click\_through\_rate\_Yes **=** 4917**/**10000 click\_through\_rate\_Yes

Out[12]: 0.4917

Click through rate prediction model (Start machine learning )

In [13]: *# yaha per maping karenge convert karenge sare data ko numbrers me* df['Gender'] **=** df['Gender']**.**map({'Male':1,

|  |
| --- |
| print("Ads Click Through Rate Prediction ") a **=**float(input("Daily Time Spent on Site:")) b **=**float(input("Age:")) c **=**float(input("Area of Income:")) d **=**float(input("Daily Internet Usage:")) e **=**input("Gender (Male = 1, Female = 0):") |

'Female':2,})

In [14]: x **=** df**.**iloc[: ,0:7] *#helps us to select a specific row or column from the data set. Usin* x **=** x**.**drop(columns **=**['Ad Topic Line','City'] , axis **=** 1) *# ye wo column hataye hai jo ir* y **=** df**.**iloc[:,9] *# Target data*

In [15]: **from** sklearn.model\_selection **import** train\_test\_split

In [16]: x\_train , x\_test , y\_train , y\_test **=** train\_test\_split(x,y,test\_size **=** 0.2 ,random\_state

In [17]: **from** sklearn.ensemble **import** RandomForestClassifier *# Supervise Ml Algoritham*

|  |  |
| --- | --- |
| rf **=** RandomForestClassifier() | *# here we will make object of data* |
|  |  |
| rf**.**fit(x,y) |  |

#rf = RandomForestClassifier() # here In [18]:

In [19]:

Out[19]: ▾RandomForestClassifier

RandomForestClassifier()

Now lets test the model by making prediction

In [20]:

features **=** np**.**array([[a,b,c,d,e]]) print("Will the User Click on Ad =" ,rf**.**predict(features))

Ads Click Through Rate Prediction

Daily Time Spent on Site:74.15

Age:31

Area of Income:59677.64

Daily Internet Usage:207.17

Gender (Male = 1, Female = 0):1

Will the User Click on Ad = ['No']

C:\Users\Hp\anaconda3\Lib\site-packages\sklearn\base.py:464: UserWarning:

X does not have valid feature names, but RandomForestClassifier was fitted with feature names

In [21]: x\_train

Out[21]: **Daily Time Spent on Site Age Area Income Daily Internet Usage Gender**

**9254** 74.15 31.0 59677.64 207.17 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1561** |  | 72.80 | 29.0 | 73910.90 | 195.69 | 1 |
| **1670** |  | 42.04 | 40.0 | 39723.97 | 165.27 | 2 |
| **6087** |  | 66.18 | 49.0 | 45632.51 | 124.32 | 2 |
| **6669** |  | 44.57 | 36.0 | 50628.31 | 168.92 | 1 |
| **...** |  | ... | ... | ... | ... | ... |
| **5734** |  | 82.07 | 41.0 | 57846.68 | 126.39 | 2 |
| **5191** |  | 89.00 | 36.0 | 50628.31 | 126.39 | 1 |
| **5390** |  | 78.84 | 35.0 | 25603.93 | 236.87 | 2 |
| **860** |  | 59.51 | 30.0 | 57877.15 | 138.71 | 1 |
| **7270** |  | 59.05 | 33.0 | 52736.33 | 113.12 | 1 |

8000 rows × 5 columns

|  |
| --- |
|  |

In [ ]: