Project Title :

Predicting income levels using U.S Census data : A Machine Learning Approach

Aim of the Project :

To develop a machine learning model that predicts whether an individual earns more than $50K per year using demographic and employment-related attributes from the U.S. Census dataset.

Importing required libraries :

In [4]:
```
import warnings
warnings.filterwarnings("ignore")
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Data preprocessing :

In [9]:
```
adult=pd.read_csv(r"D:\adult.data.csv",na_values='?',skipinitialspace=True)
adult
```

Out[9]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	na co
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	Un S
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	Un S
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	Un S
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	Un S
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
48837	39	Private	215419	Bachelors	13	Divorced	Prof-specialty	Not-in-family	White	Female	0	0	36	Un S
48838	64	NaN	321403	HS-grad	9	Widowed	NaN	Other-relative	Black	Male	0	0	40	Un S
48839	38	Private	374983	Bachelors	13	Married-civ-spouse	Prof-specialty	Husband	White	Male	0	0	50	Un S
48840	44	Private	83891	Bachelors	13	Divorced	Adm-clerical	Own-child	Asian-Pac-Islander	Male	5455	0	40	Un S
48841	35	Self-emp-inc	182148	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	60	Un S

48842 rows × 15 columns

In [11]:
```
adult.head()
```

Out[11]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	United-States
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	United-States
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	United-States
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	United-States
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	Cuba

In [13]:
```
adult.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48842 entries, 0 to 48841
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   age             48842 non-null  int64 
 1   workclass       46043 non-null  object
 2   fnlwgt          48842 non-null  int64 
 3   education       48842 non-null  object
 4   education-num   48842 non-null  int64 
 5   marital-status  48842 non-null  object
 6   occupation      46033 non-null  object
 7   relationship    48842 non-null  object
 8   race            48842 non-null  object
 9   sex             48842 non-null  object
 10  capital-gain    48842 non-null  int64 
 11  capital-loss    48842 non-null  int64 
 12  hours-per-week  48842 non-null  int64 
 13  native-country  47985 non-null  object
 14  income          48842 non-null  object
dtypes: int64(6), object(9)
memory usage: 5.6+ MB
```

In [17]:
```
adult.shape
```

Out[17]: (48842, 15)

In [19]:
```
adult.dtypes
```

Out[19]:
```
age               int64
workclass        object
fnlwgt            int64
education        object
education-num     int64
marital-status   object
occupation       object
relationship     object
race             object
sex              object
capital-gain      int64
capital-loss      int64
hours-per-week    int64
native-country   object
income           object
dtype: object
```

In [21]:
```
adult.isna().sum()
```

Out[21]:
```
age                0
workclass       2799
fnlwgt             0
education          0
education-num      0
marital-status     0
occupation      2809
relationship       0
race               0
sex                0
capital-gain       0
capital-loss       0
hours-per-week     0
native-country   857
income             0
dtype: int64
```

In [23]:
```
adult_new=adult.fillna(method='ffill')
adult_new
```

Out[23]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	na co
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	Ur
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	Ur
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	Ur
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	Ur
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
48837	39	Private	215419	Bachelors	13	Divorced	Prof-specialty	Not-in-family	White	Female	0	0	36	Ur
48838	64	Private	321403	HS-grad	9	Widowed	Prof-specialty	Other-relative	Black	Male	0	0	40	Ur
48839	38	Private	374983	Bachelors	13	Married-civ-spouse	Prof-specialty	Husband	White	Male	0	0	50	Ur
48840	44	Private	83891	Bachelors	13	Divorced	Adm-clerical	Own-child	Asian-Pac-Islander	Male	5455	0	40	Ur
48841	35	Self-emp-inc	182148	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	60	Ur

48842 rows × 15 columns

In [25]:
```
adult_new=adult_new.drop(columns=['fnlwgt','relationship','native-country'],axis=1)
adult_new
```

Out[25]:

	age	workclass	education	education-num	marital-status	occupation	race	sex	capital-gain	capital-loss	hours-per-week	income
0	39	State-gov	Bachelors	13	Never-married	Adm-clerical	White	Male	2174	0	40	<=50K
1	50	Self-emp-not-inc	Bachelors	13	Married-civ-spouse	Exec-managerial	White	Male	0	0	13	<=50K
2	38	Private	HS-grad	9	Divorced	Handlers-cleaners	White	Male	0	0	40	<=50K
3	53	Private	11th	7	Married-civ-spouse	Handlers-cleaners	Black	Male	0	0	40	<=50K
4	28	Private	Bachelors	13	Married-civ-spouse	Prof-specialty	Black	Female	0	0	40	<=50K
...	...	...	...	...	...	...	...	...	...	...	...	...
48837	39	Private	Bachelors	13	Divorced	Prof-specialty	White	Female	0	0	36	<=50K.
48838	64	Private	HS-grad	9	Widowed	Prof-specialty	Black	Male	0	0	40	<=50K.
48839	38	Private	Bachelors	13	Married-civ-spouse	Prof-specialty	White	Male	0	0	50	<=50K.
48840	44	Private	Bachelors	13	Divorced	Adm-clerical	Asian-Pac-Islander	Male	5455	0	40	<=50K.
48841	35	Self-emp-inc	Bachelors	13	Married-civ-spouse	Exec-managerial	White	Male	0	0	60	>50K.

48842 rows × 12 columns

In [27]:
```
adult_new.describe()
```

Out[27]:

	age	education-num	capital-gain	capital-loss	hours-per-week
count	48842.000000	48842.000000	48842.000000	48842.000000	48842.000000
mean	38.643585	10.078089	1079.067626	87.502314	40.422382
std	13.710510	2.570973	7452.019058	403.004552	12.391444
min	17.000000	1.000000	0.000000	0.000000	1.000000
25%	28.000000	9.000000	0.000000	0.000000	40.000000
50%	37.000000	10.000000	0.000000	0.000000	40.000000
75%	48.000000	12.000000	0.000000	0.000000	45.000000
max	90.000000	16.000000	99999.000000	4356.000000	99.000000

In [29]:
```
adult_new.isna().sum()
```

Out[29]:
```
age               0
workclass         0
education         0
education-num     0
marital-status    0
occupation        0
race              0
sex               0
capital-gain      0
capital-loss      0
hours-per-week    0
income            0
dtype: int64
```

In [31]:
```
adult_new['income']=adult_new['income'].str.strip().str.replace('.','',regex=False)
adult_new
```

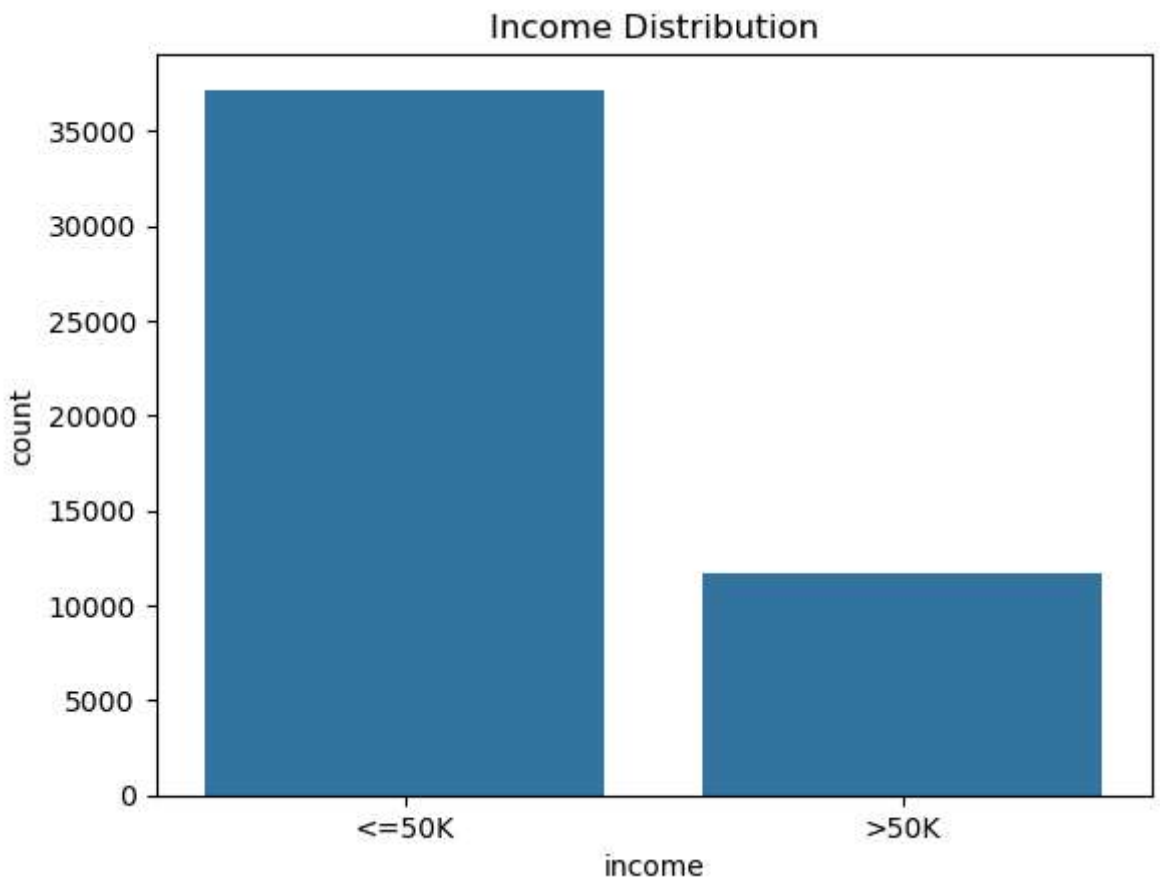
Out[31]:

	age	workclass	education	education-num	marital-status	occupation	race	sex	capital-gain	capital-loss	hours-per-week	income
0	39	State-gov	Bachelors	13	Never-married	Adm-clerical	White	Male	2174	0	40	<=50K
1	50	Self-emp-not-inc	Bachelors	13	Married-civ-spouse	Exec-managerial	White	Male	0	0	13	<=50K
2	38	Private	HS-grad	9	Divorced	Handlers-cleaners	White	Male	0	0	40	<=50K
3	53	Private	11th	7	Married-civ-spouse	Handlers-cleaners	Black	Male	0	0	40	<=50K
4	28	Private	Bachelors	13	Married-civ-spouse	Prof-specialty	Black	Female	0	0	40	<=50K
...	...	...	...	...	...	...	...	...	...	...	...	...
48837	39	Private	Bachelors	13	Divorced	Prof-specialty	White	Female	0	0	36	<=50K
48838	64	Private	HS-grad	9	Widowed	Prof-specialty	Black	Male	0	0	40	<=50K
48839	38	Private	Bachelors	13	Married-civ-spouse	Prof-specialty	White	Male	0	0	50	<=50K
48840	44	Private	Bachelors	13	Divorced	Adm-clerical	Asian-Pac-Islander	Male	5455	0	40	<=50K
48841	35	Self-emp-inc	Bachelors	13	Married-civ-spouse	Exec-managerial	White	Male	0	0	60	>50K

48842 rows × 12 columns

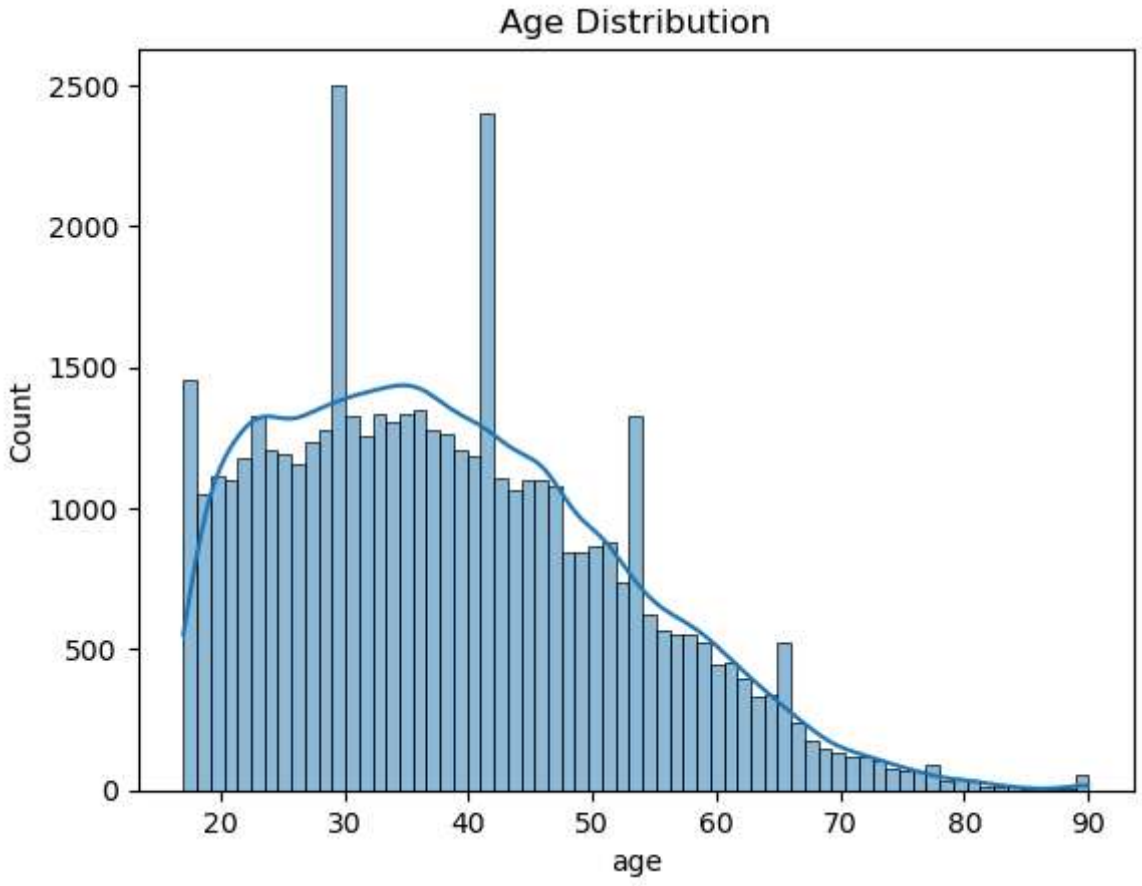
In [33]:
```
sns.countplot(x='income',data=adult_new)
plt.title('Income Distribution')
```

Out[33]:
```
Text(0.5, 1.0, 'Income Distribution')
```



In [35]:
```
sns.histplot(adult_new['age'],kde=True)
plt.title('Age Distribution')
```

Out[35]:
```
Text(0.5, 1.0, 'Age Distribution')
```



Encode categorical variables

In [38]:
```
from sklearn.preprocessing import LabelEncoder
```

In [40]:
```
encode=LabelEncoder()
adult_new['income']=encode.fit_transform(adult_new['income'])
adult_new['income']
```

Out[40]:
```
0        0
1        0
2        0
3        0
4        0
        ..
48837    0
48838    0
48839    0
48840    0
48841    1
Name: income, Length: 48842, dtype: int32
```

In [44]:
```
adult_new=pd.get_dummies(adult_new,drop_first=True)
adult_new
```

Out[44]:

	age	education-num	capital-gain	capital-loss	hours-per-week	income	workclass_Local-gov	workclass_Never-worked	workclass_Private	workclass_Self-emp-inc	...	occu
0	39	13	2174	0	40	0	False	False	False	False	...	
1	50	13	0	0	13	0	False	False	False	False	...	
2	38	9	0	0	40	0	False	False	True	False	...	
3	53	7	0	0	40	0	False	False	True	False	...	
4	28	13	0	0	40	0	False	False	True	False	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
48837	39	13	0	0	36	0	False	False	True	False	...	
48838	64	9	0	0	40	0	False	False	True	False	...	
48839	38	13	0	0	50	0	False	False	True	False	...	
48840	44	13	5455	0	40	0	False	False	True	False	...	
48841	35	13	0	0	60	1	False	False	False	True	...	

48842 rows × 52 columns

Split into features and target sets

In [47]:
```
x=adult_new.drop(columns=['income'])
x
```

Out[47]:

	age	education-num	capital-gain	capital-loss	hours-per-week	workclass_Local-gov	workclass_Never-worked	workclass_Private	workclass_Self-emp-inc	workclass_Self-emp-not-inc
0	39	13	2174	0	40	False	False	False	False	False
1	50	13	0	0	13	False	False	False	False	True
2	38	9	0	0	40	False	False	True	False	False
3	53	7	0	0	40	False	False	True	False	False
4	28	13	0	0	40	False	False	True	False	False
...	...	...	...	...	...	...	...	...	...	...
48837	39	13	0	0	36	False	False	True	False	False
48838	64	9	0	0	40	False	False	True	False	False
48839	38	13	0	0	50	False	False	True	False	False
48840	44	13	5455	0	40	False	False	True	False	False
48841	35	13	0	0	60	False	False	False	True	False

48842 rows × 51 columns

In [49]:
```
y=adult_new['income']
y
```

Out[49]:
```
0        0
1        0
2        0
3        0
4        0
        ..
48837    0
48838    0
48839    0
48840    0
48841    1
Name: income, Length: 48842, dtype: int32
```

Train-test split

In [52]:
```
from sklearn.model_selection import train_test_split
```

In [54]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.8,random_state=42,stratify=y)
```

In [56]:
```
x_test.head()
```

Out[56]:

	age	education-num	capital-gain	capital-loss	hours-per-week	workclass_Local-gov	workclass_Never-worked	workclass_Private	workclass_Self-emp-inc	workclass_Self-emp-not-inc
40421	30	9	0	0	40	False	False	True	False	False
47738	54	12	0	0	39	False	False	False	False	False
518	21	10	0	0	35	False	False	True	False	False
8564	35	9	2885	0	40	False	False	True	False	False
31355	42	10	0	0	45	True	False	False	False	False

5 rows × 51 columns

In [58]:
```
x_train.shape
```

Out[58]:
```
(39073, 51)
```

In [60]:
```
x_test.shape
```

Out[60]:
```
(9769, 51)
```

Feature scaling

In [63]:
```
from sklearn.preprocessing import StandardScaler
```

In [65]:
```
scaler=StandardScaler()
scaler
```

Out[65]:

StandardScaler

StandardScaler()

In [67]:
```
scaled_x_train=scaler.fit_transform(x_train)
scaled_x_train
```

Out[67]:
```
array([[-0.12009008,  1.13821044, -0.14407503, ..., -0.09327335,
         0.4115339 , -1.42415255],
       [ 1.26824482, -0.80402454, -0.14407503, ..., -0.09327335,
         0.4115339 ,  0.70217197],
       [ 1.04903405, -0.41557754, -0.14407503, ..., -0.09327335,
         0.4115339 ,  0.70217197],
       ...,
       [ 1.63359611, -0.02713055, -0.14407503, ..., -0.09327335,
         0.4115339 ,  0.70217197],
       [-0.04701982,  0.36131645, -0.14407503, ..., -0.09327335,
         0.4115339 , -1.42415255],
       [ 0.31833147, -0.02713055, -0.14407503, ..., -0.09327335,
        -2.42993349,  0.70217197]])
```

In [69]:
```
scaled_x_test=scaler.fit_transform(x_test)
scaled_x_test
```

Out[69]:
```
array([[-0.62594162, -0.43451805, -0.14860513, ..., -0.08434099,
         0.41257776,  0.71243367],
       [ 1.11198537,  0.73872873, -0.14860513, ..., -0.08434099,
         0.41257776,  0.71243367],
       [-1.27766425, -0.04343579, -0.14860513, ..., -0.08434099,
         0.41257776, -1.40363945],
       ...,
       [ 1.4740535 , -0.43451805, -0.14860513, ..., -0.08434099,
         0.41257776,  0.71243367],
       [-0.69835525, -0.04343579, -0.14860513, ..., -0.08434099,
         0.41257776, -1.40363945],
       [ 1.4740535 , -0.04343579, -0.14860513, ..., -0.08434099,
        -2.42378549,  0.71243367]])
```

In [ ]: