2306aml109-byogeshwar-assignment-4

July 1, 2023

[ ]:
```
##Assignment 4
```

[1]:
```
import pandas as pd
import numpy as np
```

[20]:
```
features = ["Age", "Workclass", "fnlwgt", "Education", "Education-Num",␣
 ↪"Martial Status", "Occupation", "Relationship",
            "Race", "Sex", "Capital Gain", "Capital Loss", "Hours per week",␣
 ↪"Country", "Target"]

df = pd.read_csv('adult.data', names=features)
df
```

[20]:

	Age	Workclass	fnlwgt	Education	Education-Num	\
0	39	State-gov	77516	Bachelors	13	
1	50	Self-emp-not-inc	83311	Bachelors	13	
2	38	Private	215646	HS-grad	9	
3	53	Private	234721	11th	7	
4	28	Private	338409	Bachelors	13	
…	…	…	…	…	…	
32556	27	Private	257302	Assoc-acdm	12	
32557	40	Private	154374	HS-grad	9	
32558	58	Private	151910	HS-grad	9	
32559	22	Private	201490	HS-grad	9	
32560	52	Self-emp-inc	287927	HS-grad	9	

	Martial Status	Occupation	Relationship	Race	\
0	Never-married	Adm-clerical	Not-in-family	White	
1	Married-civ-spouse	Exec-managerial	Husband	White	
2	Divorced	Handlers-cleaners	Not-in-family	White	
3	Married-civ-spouse	Handlers-cleaners	Husband	Black	
4	Married-civ-spouse	Prof-specialty	Wife	Black	
…	…	…	…	…	
32556	Married-civ-spouse	Tech-support	Wife	White	
32557	Married-civ-spouse	Machine-op-inspct	Husband	White	
32558	Widowed	Adm-clerical	Unmarried	White	
32559	Never-married	Adm-clerical	Own-child	White	

```
32560   Married-civ-spouse     Exec-managerial            Wife   White

            Sex  Capital Gain  Capital Loss  Hours per week         Country  \
0          Male          2174             0              40   United-States
1          Male             0             0              13   United-States
2          Male             0             0              40   United-States
3          Male             0             0              40   United-States
4        Female             0             0              40            Cuba
…         …           …           …             …             …
32556    Female             0             0              38   United-States
32557      Male             0             0              40   United-States
32558    Female             0             0              40   United-States
32559      Male             0             0              20   United-States
32560    Female         15024             0              40   United-States

        Target
0        <=50K
1        <=50K
2        <=50K
3        <=50K
4        <=50K
…        …
32556    <=50K
32557     >50K
32558    <=50K
32559    <=50K
32560     >50K

[32561 rows x 15 columns]
```

[ ]:
```
## 1. How many men and women (sex feature) are represented in this dataset?
```

[8]:
```
# Count the number of men and women
gender_counts = df['Sex'].value_counts()

# Display the counts
print(gender_counts)
```

```
 Male      21790
 Female    10771
Name: Sex, dtype: int64
```

[ ]:
```
##2. What is the average age (age feature) of women?
```

[33]:
```
# Filter the dataset for women
women_data = df[df['Sex'] == ' Female']
##print(len(women_data))
```

```
# Calculate the average age
average_age_women = women_data['Age'].mean()

# Display the average age of women
print('Average Age of Women ',average_age_women)
```

```
10771
Average Age of Women  36.85823043357163
```

[ ]:
```
##3. What is the proportion of German citizens (native-country feature)?
```

[39]:
```
# Calculate the proportion of German citizens
german_citizens = (df['Country'] == ' Germany').sum()
total_citizens = len(df)
print('Total Number of German Citizens',german_citizens)
print('Total Number of Citizens',total_citizens)
proportion_german_citizens = german_citizens / total_citizens

# Display the proportion of German citizens
print('proportion of German citizens is ',proportion_german_citizens)
```

```
Total Number of German Citizens 137
Total Number of Citizens 32561
proportion of German citizens is  0.004207487485028101
```

[ ]:
```
##4-5. What are mean value and standard deviation of the age of those who␣
↪recieve more than 50K per year (salary feature) and those who receive less␣
↪than 50K per year?
```

[40]:
```
# Subset for individuals earning more than 50K
high_income_ages = df[df['Target'] == ' >50K']['Age']
mean_high_income_age = high_income_ages.mean()
std_high_income_age = high_income_ages.std()

# Subset for individuals earning less than or equal to 50K
low_income_ages = df[df['Target'] == ' <=50K']['Age']
mean_low_income_age = low_income_ages.mean()
std_low_income_age = low_income_ages.std()

# Display the results
print("Mean age of high-income individuals:", mean_high_income_age)
print("Standard deviation of age for high-income individuals:",␣
↪std_high_income_age)
print("Mean age of low-income individuals:", mean_low_income_age)
print("Standard deviation of age for low-income individuals:",␣
↪std_low_income_age)
```

```
Mean age of high-income individuals: 44.24984058155847
```

```
Standard deviation of age for high-income individuals: 10.51902771985177
Mean age of low-income individuals: 36.78373786407767
Standard deviation of age for low-income individuals: 14.020088490824813
```

[ ]:
```
##6. Is it true that people who receive more than 50k have at least high school␣
↪education? (education - Bachelors, Prof-school, Assoc-acdm, Assoc-voc,␣
↪Masters or Doctorate feature)
```

[46]:
```
# Subset for individuals earning more than 50K
high_income_data = df[df['Target'] == ' >50K']

##print(len(high_income_data))

# Check if there are any individuals with education level lower than high school
lower_education = high_income_data[~high_income_data['Education'].isin(['␣
↪Bachelors', ' Prof-school', ' Assoc-acdm', ' Assoc-voc', ' Masters', '␣
↪Doctorate'])]

##print(len(lower_education))

# Determine if the statement is true or false
statement_true = len(lower_education) == 0

# Display the result
print("Is it true that people who receive more than 50K have at least high␣
↪school education?", statement_true)
```

```
Is it true that people who receive more than 50K have at least high school
education? False
```

[ ]:
```

```