demographicDataAnalyzer

July 11, 2023

1 DEMOGRAPHIC DATA ANALYZER

```
[3]: import pandas as pd
```

1.1 Read Data From File

```
[4]: df = pd.read_csv("adult.data.csv")
```

1.2 How many of each race are represented?

```
[5]: race_count = df["race"].value_counts()
     race_count
```

```
[5]: race
     White                 27816
     Black                  3124
     Asian-Pac-Islander     1039
     Amer-Indian-Eskimo      311
     Other                   271
     Name: count, dtype: int64
```

1.3 Average age of men

```
[6]: average_age_men = df.loc[df["sex"] == "Male", "age"].mean()
     average_age_men
```

```
[6]: 39.43354749885268
```

1.4 Percentage of people who have a Bachelor's degree

```
[41]: percentage_bachelors = round((df.loc[df["education"] == "Bachelors"].shape[0]/
        ↪df.shape[0])*100,1)
      percentage_bachelors
```

```
[41]: 16.4
```

1.5 Percentage of people with advanced and without advanced education (Bachelors, Masters, or Doctorate) make more than 50K

```
[8]: higher_education = df[df["education"].str.contains("|".join(["Bachelors",␣
      ↪"Masters", "Doctorate"]))]
     higher_education
```

```
[8]:        age         workclass  fnlwgt  education  education-num  \
     0       39         State-gov   77516  Bachelors             13
     1       50  Self-emp-not-inc   83311  Bachelors             13
     4       28           Private  338409  Bachelors             13
     5       37           Private  284582    Masters             14
     8       31           Private   45781    Masters             14
     …     …                 …      …      …              …
     32538   38           Private  139180  Bachelors             13
     32539   71                 ?  287372  Doctorate             16
     32544   31           Private  199655    Masters             14
     32553   32           Private  116138    Masters             14
     32554   53           Private  321865    Masters             14

                marital-status       occupation   relationship                race  \
     0           Never-married     Adm-clerical  Not-in-family               White
     1      Married-civ-spouse  Exec-managerial        Husband               White
     4      Married-civ-spouse   Prof-specialty           Wife               Black
     5      Married-civ-spouse  Exec-managerial           Wife               White
     8           Never-married   Prof-specialty  Not-in-family               White
     …                     …              …            …                 …
     32538            Divorced   Prof-specialty      Unmarried               Black
     32539  Married-civ-spouse                ?        Husband               White
     32544            Divorced    Other-service  Not-in-family               Other
     32553       Never-married     Tech-support  Not-in-family  Asian-Pac-Islander
     32554  Married-civ-spouse  Exec-managerial        Husband               White

               sex  capital-gain  capital-loss  hours-per-week native-country  \
     0       Male          2174             0              40  United-States
     1       Male             0             0              13  United-States
     4     Female             0             0              40           Cuba
     5     Female             0             0              40  United-States
     8     Female         14084             0              50  United-States
     …       …           …            …             …              …
     32538 Female         15020             0              45  United-States
     32539   Male             0             0              10  United-States
     32544 Female             0             0              30  United-States
     32553   Male             0             0              11         Taiwan
     32554   Male             0             0              40  United-States

           salary
```

```
0      <=50K
1      <=50K
4      <=50K
5      <=50K
8       >50K
…      …
32538   >50K
32539   >50K
32544  <=50K
32553  <=50K
32554   >50K

[7491 rows x 15 columns]
```

[9]:
```
lower_education = df[~df["education"].str.contains("|".join(["Bachelors",␣
↪"Masters", "Doctorate"]))]
lower_education
```

[9]:
```
       age         workclass  fnlwgt     education  education-num  \
2       38           Private  215646        HS-grad              9
3       53           Private  234721          11th              7
6       49           Private  160187           9th              5
7       52  Self-emp-not-inc  209642        HS-grad              9
10      37           Private  280464  Some-college             10
…    …               …     …           …            …
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

              marital-status         occupation   relationship   race     sex  \
2                   Divorced  Handlers-cleaners  Not-in-family  White    Male
3         Married-civ-spouse  Handlers-cleaners        Husband  Black    Male
6      Married-spouse-absent      Other-service  Not-in-family  Black  Female
7         Married-civ-spouse    Exec-managerial        Husband  White    Male
10        Married-civ-spouse    Exec-managerial        Husband  Black    Male
…                      …                …            …    …     …
32556     Married-civ-spouse       Tech-support           Wife  White  Female
32557     Married-civ-spouse  Machine-op-inspct        Husband  White    Male
32558                Widowed       Adm-clerical      Unmarried  White  Female
32559          Never-married       Adm-clerical      Own-child  White    Male
32560     Married-civ-spouse    Exec-managerial           Wife  White  Female

       capital-gain  capital-loss  hours-per-week native-country salary
2                 0             0              40  United-States  <=50K
3                 0             0              40  United-States  <=50K
```

```
6                    0             0               16        Jamaica  <=50K
7                    0             0               45  United-States   >50K
10                   0             0               80  United-States   >50K
…              …            …             …             …      …
32556                0             0               38  United-States  <=50K
32557                0             0               40  United-States   >50K
32558                0             0               40  United-States  <=50K
32559                0             0               20  United-States  <=50K
32560            15024             0               40  United-States   >50K

[25070 rows x 15 columns]
```

[10]:
```
higher_education_rich = round((higher_education[higher_education["salary"].str.
  ↪contains(">")].shape[0]/higher_education.shape[0])*100,1)
higher_education_rich
```

[10]:
```
46.5
```

[11]:
```
lower_education_rich = round((lower_education[lower_education["salary"].str.
  ↪contains(">")].shape[0]/lower_education.shape[0])*100,1)
lower_education_rich
```

[11]:
```
17.4
```

1.6 Minimum number of hours a person works per week

[12]:
```
min_work_hours = df["hours-per-week"].min()
min_work_hours
```

[12]:
```
1
```

[45]:
```
num_min_workers = df.loc[df["hours-per-week"] == 1].shape[0]
num_min_workers
```

[45]:
```
20
```

1.7 Percentage of the people who work the minimum number of hours per week have a salary of >50K

[46]:
```
rich_percentage = (df.loc[df["hours-per-week"] == 1 & df["salary"].str.
  ↪contains(">")].shape[0]/num_min_workers)*100
rich_percentage
```

[46]:
```
10.0
```

1.8 What country has the highest percentage of people that earn >50K?

```
[29]: highest_earning_country = df.groupby(["native-country"])["salary"].apply(lambda␣
       ↪x: (x == ">50K").mean()*100).idxmax() #[df["salary"].str.contains(">"),␣
       ↪"native-country"]
      highest_earning_country
```

```
[29]: 'Iran'
```

```
[32]: highest_earning_country_percentage = round(df.
       ↪groupby(["native-country"])["salary"].apply(lambda x: (x == ">50K").
       ↪mean()*100).max(),1)
      highest_earning_country_percentage
```

```
[32]: 41.9
```

1.9 The most popular occupation for those who earn >50K in India

```
[40]: top_IN_occupation = df.loc[(df["native-country"] == "India") & (df["salary"] ==␣
       ↪">50K"), "occupation"].value_counts().idxmax()
      top_IN_occupation
```

```
[40]: 'Prof-specialty'
```