# AnaliseExploratoriaAdultosPt

November 9, 2024

[41]:
```
import pandas as pd
```

[42]:
```
df = pd.read_csv('adult.data.csv')
```

### 0.1 Observando as características do DataSet Adult

[43]:
```
df.info()
df.isna().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   age             32561 non-null  int64 
 1   workclass       32561 non-null  object
 2   fnlwgt          32561 non-null  int64 
 3   education       32561 non-null  object
 4   education-num   32561 non-null  int64 
 5   marital-status  32561 non-null  object
 6   occupation      32561 non-null  object
 7   relationship    32561 non-null  object
 8   race            32561 non-null  object
 9   sex             32561 non-null  object
 10  capital-gain    32561 non-null  int64 
 11  capital-loss    32561 non-null  int64 
 12  hours-per-week  32561 non-null  int64 
 13  native-country  32561 non-null  object
 14  salary          32561 non-null  object
dtypes: int64(6), object(9)
memory usage: 3.7+ MB
```

[43]:
```
age               0
workclass         0
fnlwgt            0
education         0
education-num     0
marital-status    0
```

```
occupation        0
relationship      0
race              0
sex               0
capital-gain      0
capital-loss      0
hours-per-week    0
native-country    0
salary            0
dtype: int64
```

## 1 Quantas pessoas de cada grupo está representada nesse dataset?

[44]:
```
race_count = df.groupby('race')['race'].count()
print(race_count)
```

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

## 2 Qual a idade media dos homens ?

[45]:
```
average_age_men = df[df['sex'] == 'Male']['age'].mean()
print(round(average_age_men,1))
```

```
39.4
```

## 3 Qual a porcentagem de pessoas que tem um diploma de bacharelado?

[46]:
```
percentage_bachelors = (df[df['education'] == 'Bachelors'].shape[0]) / df.
  ↪shape[0] * 100
print(round(percentage_bachelors,1))
```

```
16.4
```

## 4 Número de pessoas com e sem diplomas Bachelors, Masters, ou Doctorate

[47]:
```
higher_education = df[(df['education'] == 'Bachelors') | (df['education'] ==␣
 ↪'Masters') | (df['education'] == 'Doctorate')]
lower_education = df[(df['education'] != 'Bachelors') & (df['education'] !=␣
 ↪'Masters') & (df['education'] != 'Doctorate')]

print(f'Pessoas com diploma(Bachelors, Masters, ou Doctorate):␣
 ↪{higher_education.shape[0]}')
print(f'Pessoas sem diplomas(Bachelors, Masters, ou Doctorate):␣
 ↪{lower_education.shape[0]}')
```

```
Pessoas com diploma(Bachelors, Masters, ou Doctorate): 7491
Pessoas sem diplomas(Bachelors, Masters, ou Doctorate): 25070
```

## 5 Porcentagem de pessoas com e sem diplomas (Bachelors, Masters, ou Doctorate) que recebem mais de 50K

[48]:
```
higher_education_rich = higher_education[higher_education['salary'] == '>50K'].
 ↪shape[0] / higher_education.shape[0] * 100
print(f'Porcentagem de pessoas com diplomas que recebem mais de 50K:␣
 ↪{round(higher_education_rich,1)}')

lower_education_rich = lower_education[lower_education['salary'] == '>50K'].
 ↪shape[0] / lower_education.shape[0] * 100
print(f'Porcentagem de pessoas sem diploma que recebem mais de 50K:␣
 ↪{round(lower_education_rich,1)}')
```

```
Porcentagem de pessoas com diplomas que recebem mais de 50K: 46.5
Porcentagem de pessoas sem diploma que recebem mais de 50K: 17.4
```

## 6 Qual o menor número de horas que uma pessoa trabalha por semana?

[49]:
```
min_hours_week = df['hours-per-week'].min()
print(min_hours_week)
```

```
1
```

## 7 Qual a porcentagem de pessoas que trabalham o número mínimo de horas e tem um salário maior que 50K?

[50]:
```
num_min_workers = df[df['hours-per-week'] == min_hours_week]
rich_percentage = num_min_workers[num_min_workers['salary'] == '>50K'].shape[0]␣
 ↪/ num_min_workers.shape[0] * 100
print(f'Número de pessoas que trabalham o mínimo de horas e recebem mais de 50K:
 ↪ {round(rich_percentage,1)}%')
```

```
Número de pessoas que trabalham o mínimo de horas e recebem mais de 50K: 10.0%
```

## 8 Qual país tem a maior porcentagem de pessoas que recebem mais de 50K?

[51]:
```
more_then_50 = df[df['salary'] == '>50K']

highest_earning_country = more_then_50.groupby('native-country').
 ↪count()['salary']
highest_earning_country_percentage = (more_then_50.groupby('native-country').
 ↪count()['salary'] / df.groupby('native-country').count()['salary']) * 100
highest_earning_country = highest_earning_country_percentage.idxmax()
highest_earning_country_percentage = highest_earning_country_percentage.max()
print(highest_earning_country)
print(f'{round(highest_earning_country_percentage,1)}%')
```

```
Iran
41.9%
```

## 9 Qual profissão mais popular entre as pessoas que recebem mais de 50K na India?

[52]:
```
# Identify the most popular occupation for those who earn >50K in India.
top_IN_occupation = df.loc[(df['native-country'] == "India")]
top_IN_occupation = top_IN_occupation.loc[(df['salary'] == ">50K")]
top_IN_occupation = top_IN_occupation['occupation'].value_counts().idxmax()
print(top_IN_occupation)
```

```
Prof-specialty
```

### 9.0.1 O que é Prof-specialty?

É um trabalho que requer o uso de conhecimentos e habilidades especializadas em uma área específica, como medicina, engenharia ou biotecnologia.