Лабораторная работа № 2

Изучение библиотек обработки данных ¶

Цель лабораторной работы: изучение библиотек обработки данных Pandas и PandaSQL.

Выполнил: Ханмурзин Тагир ИУ5-64

Часть 1

Выполните первое демонстрационное задание "demo assignment" под названием "Exploratory data analysis with Pandas" со страницы курса https://mlcourse.ai/assignments

In [11]:
```
import pandas as pd
```

In [12]:
```
data = pd.read_csv('adult.data.csv')
data.head()
```

Out[12]:

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

How many men and women (sex feature) are represented in this dataset?
Сколько мужчин и женщин (половая принадлежность) представлено в этом наборе данных?

In [13]:
```
data['sex'].value_counts() # Данная функцция количество элементов по типам
```

Out[13]:
```
Male      21790
Female    10771
Name: sex, dtype: int64
```

What is the average age (age feature) of women?
Каков средний возраст (возрастная характеристика) женщины?

In [14]:
```
data.loc[data['sex'] == 'Female', 'age'].mean()
```

Out[14]:
```
36.85823043357163
```

What is the proportion of German citizens (native-country feature)?
Какова доля граждан Германии (характеристика родной страны)?

In [15]:
```
float((data['native-country'] == 'Germany').sum()) / data.shape[0]
```

Out[15]:
```
0.004207487485028101
```

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?
Каково среднее значение и стандартное отклонение возраста тех, кто получает более 50 тыс. В год (функция заработной платы) и тех, кто получает менее 50 тыс. В год?

In [16]:
```
ages1 = data.loc[data['salary'] == '>50K', 'age']  # Узнаём список людей, которые получают больше, чем 50 тыс руб в год
ages2 = data.loc[data['salary'] == '<=50K', 'age'] # Узнаём список людей, которые получают меньше, чем 50 тыс руб в год

avr1 = ages1.mean() # Среднее значение возраста 1
avr2 = ages2.mean() # Среднее значение возраста 2

print ("Средний возраст > 50: ", round(avr1), "+-", round(ages1.std(), 1))
print ("Средний возраст < 50: ", round(avr2), "+-", round(ages2.std(), 1))
```

```
Средний возраст > 50:  44 +- 10.5
Средний возраст < 50:  37 +- 14.0
```

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)
Правда ли, что люди, которые получают более 50 тысяч, имеют по крайней мере среднее образование? (образование - бакалавриат, проф-школа, доцент, доцент, магистр или докторантура)

In [17]:
```
data.loc[data['salary'] == '>50K', 'education'].unique()
```

Out[17]:
```
array(['HS-grad', 'Masters', 'Bachelors', 'Some-college', 'Assoc-voc',
       'Doctorate', 'Prof-school', 'Assoc-acdm', '7th-8th', '12th',
       '10th', '11th', '9th', '5th-6th', '1st-4th'], dtype=object)
```

Вывели все вариации образовательных учреждений, в которых учились люди, которые зарабатывают больше 50 тыс рублей в год
Ответ: нет, не правда

Display statistics of age for each race (race feature) and each gender. Use groupby() and describe(). Find the maximum age of men of Amer-Indian-Eskimo race.
Показать статистику возраста для каждой расы (особенность расы) и каждого пола. Используйте groupby() и describe(). Найти максимальный возраст мужчин американо-индийско-эскимосской расы.

In [18]:
```
# Максимальный возраст мужчин американо-индийско-эскимосской расы
print("Максимальный возраст мужчин американо-индийско-эскимосской расы: ", data.loc[(data['race'] == 'Amer-Indian-Eskimo') & (da
ta['sex'] == 'Male')]['age'].max())
print("\n")
# Статистика возраста для каждой расы
for (race, sex), sub_df in data.groupby(['race', 'sex']):
    print("Race: {0}, sex: {1}".format(race, sex))
    print(sub_df['age'].describe())
```

```
Максимальный возраст мужчин американо-индийско-эскимосской расы:  82


Race: Amer-Indian-Eskimo, sex: Female
count    119.000000
mean      37.117647
std       13.114991
min       17.000000
25%       27.000000
50%       36.000000
75%       46.000000
max       80.000000
Name: age, dtype: float64
Race: Amer-Indian-Eskimo, sex: Male
count    192.000000
mean      37.208333
std       12.049563
min       17.000000
25%       28.000000
50%       35.000000
75%       45.000000
max       82.000000
Name: age, dtype: float64
Race: Asian-Pac-Islander, sex: Female
count    346.000000
mean      35.089595
std       12.300845
min       17.000000
25%       25.000000
50%       33.000000
75%       43.750000
max       75.000000
Name: age, dtype: float64
Race: Asian-Pac-Islander, sex: Male
count    693.000000
mean      39.073593
std       12.883944
min       18.000000
25%       29.000000
50%       37.000000
75%       46.000000
max       90.000000
Name: age, dtype: float64
Race: Black, sex: Female
count    1555.000000
mean       37.854019
std        12.637197
min        17.000000
25%        28.000000
50%        37.000000
75%        46.000000
max        90.000000
Name: age, dtype: float64
Race: Black, sex: Male
count    1569.000000
mean       37.682600
std        12.882612
min        17.000000
25%        27.000000
50%        36.000000
75%        46.000000
max        90.000000
Name: age, dtype: float64
Race: Other, sex: Female
count    109.000000
mean      31.678899
std       11.631599
min       17.000000
25%       23.000000
50%       29.000000
75%       39.000000
max       74.000000
Name: age, dtype: float64
Race: Other, sex: Male
count    162.000000
mean      34.654321
std       11.355531
min       17.000000
25%       26.000000
50%       32.000000
75%       42.000000
max       77.000000
Name: age, dtype: float64
Race: White, sex: Female
count    8642.000000
mean       36.811618
std        14.329093
min        17.000000
25%        25.000000
50%        35.000000
75%        46.000000
max        90.000000
Name: age, dtype: float64
Race: White, sex: Male
count    19174.000000
mean        39.652498
std         13.436029
min         17.000000
25%         29.000000
50%         38.000000
75%         49.000000
max         90.000000
Name: age, dtype: float64
```

Among whom the proportion of those who earn a lot(>50K) is more: among married or single men (marital-status feature)? Consider married those who have a marital-status starting with Married (Married-civ-spouse, Married-spouse-absent or Married-AF-spouse), the rest are considered bachelors.
Среди кого больше доля тех, кто много зарабатывает (> 50 тыс.): Среди женатых или одиноких мужчин (особенность семейного положения)? Считается, что в браке находятся те, кто имеет семейное положение, начиная с женатых (женатых гражданских супругов, женатых супругов нет или женатых супругов), остальные считаются холостяками.

In [19]:
```
data.loc[(data['sex'] == 'Male') & (data['marital-status'].isin([
    'Never-married',
    'Separated',
    'Divorced',
    'Widowed'
])), 'salary'].value_counts()
```

Out[19]:
```
<=50K    7552
>50K      697
Name: salary, dtype: int64
```

In [20]:
```
data.loc[(data['sex'] == 'Male') & (data['marital-status'].str.startswith('Married')), 'salary'].value_counts()
```

Out[20]:
```
<=50K    7576
>50K     5965
Name: salary, dtype: int64
```

In [21]:
```
data['marital-status'].value_counts()
```

Out[21]:
```
Married-civ-spouse       14976
Never-married            10683
Divorced                  4443
Separated                 1025
Widowed                    993
Married-spouse-absent      418
Married-AF-spouse           23
Name: marital-status, dtype: int64
```

What is the maximum number of hours a person works per week (hours-per-week feature)? How many people work such a number of hours and what is the percentage of those who earn a lot among them?
Какое максимальное количество часов работает человек в неделю (функция часов в неделю)? Сколько людей работает такое количество часов и каков процент тех, кто много зарабатывает среди них?

In [22]:
```
max_load = data['hours-per-week'].max()
print("Max time - {0} hours./week.".format(max_load))

num_workaholics = data[data['hours-per-week'] == max_load].shape[0]
print("Total number of such hard workers {0}".format(num_workaholics))

rich_share = float(data[(data['hours-per-week'] == max_load)
                 & (data['salary'] == '>50K')].shape[0]) / num_workaholics
print("Percentage of rich among them {0}%".format(int(100 * rich_share)))
```

```
Max time - 99 hours./week.
Total number of such hard workers 85
Percentage of rich among them 29%
```

Count the average time of work (hours-per-week) those who earning a little and a lot (salary) for each country (native-country).
Посчитайте среднее время работы (в часах в неделю) тех, кто зарабатывает мало и много (зарплата) для каждой страны (родной страны).

In [23]:
```
for (country, salary), sub_df in data.groupby(['native-country', 'salary']):
    print(country, salary, round(sub_df['hours-per-week'].mean(), 2))
```

```
? <=50K 40.16
? >50K 45.55
Cambodia <=50K 41.42
Cambodia >50K 40.0
Canada <=50K 37.91
Canada >50K 45.64
China <=50K 37.38
China >50K 38.9
Columbia <=50K 38.68
Columbia >50K 50.0
Cuba <=50K 37.99
Cuba >50K 42.44
Dominican-Republic <=50K 42.34
Dominican-Republic >50K 47.0
Ecuador <=50K 38.04
Ecuador >50K 48.75
El-Salvador <=50K 36.03
El-Salvador >50K 45.0
England <=50K 40.48
England >50K 44.53
France <=50K 41.06
France >50K 50.75
Germany <=50K 39.14
Germany >50K 44.98
Greece <=50K 41.81
Greece >50K 50.62
Guatemala <=50K 39.36
Guatemala >50K 36.67
Haiti <=50K 36.33
Haiti >50K 42.75
Holand-Netherlands <=50K 40.0
Honduras <=50K 34.33
Honduras >50K 60.0
Hong <=50K 39.14
Hong >50K 45.0
Hungary <=50K 31.3
Hungary >50K 50.0
India <=50K 38.23
India >50K 46.48
Iran <=50K 41.44
Iran >50K 47.5
Ireland <=50K 40.95
Ireland >50K 48.0
Italy <=50K 39.62
Italy >50K 45.4
Jamaica <=50K 38.24
Jamaica >50K 41.1
Japan <=50K 41.0
Japan >50K 47.96
Laos <=50K 40.38
Laos >50K 40.0
Mexico <=50K 40.0
Mexico >50K 46.58
Nicaragua <=50K 36.09
Nicaragua >50K 37.5
Outlying-US(Guam-USVI-etc) <=50K 41.86
Peru <=50K 35.07
Peru >50K 40.0
Philippines <=50K 38.07
Philippines >50K 43.03
Poland <=50K 38.17
Poland >50K 39.0
Portugal <=50K 41.94
Portugal >50K 41.5
Puerto-Rico <=50K 38.47
Puerto-Rico >50K 39.42
Scotland <=50K 39.44
Scotland >50K 46.67
South <=50K 40.16
South >50K 51.44
Taiwan <=50K 33.77
Taiwan >50K 46.8
Thailand <=50K 42.87
Thailand >50K 58.33
Trinadad&Tobago <=50K 37.06
Trinadad&Tobago >50K 40.0
United-States <=50K 38.8
United-States >50K 45.51
Vietnam <=50K 37.19
Vietnam >50K 39.2
Yugoslavia <=50K 41.6
Yugoslavia >50K 49.5
```

Часть 2

Выполните следующие запросы с использованием двух различных библиотек - Pandas и PandaSQL

- один произвольный запрос на соединение двух наборов данных
- один произвольный запрос на группировку набора данных с использованием функций агрегирования Сравните время выполнения каждого запроса в Pandas и PandaSQL.

ПРИМЕР: https://github.com/miptgirl/udacity_engagement_analysis/blob/master/pandasql_example.ipynb

In [101]:
```
googleplaystore = pd.read_csv('googleplaystore.csv')
googleplaystore.head(5)
```

Out[101]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [102]:
```
userreviews = pd.read_csv('googleplaystore_user_reviews.csv')
userreviews.head(5)
```

Out[102]:

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00	0.533333
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462
2	10 Best Foods for You	NaN	NaN	NaN	NaN
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40	0.875000
4	10 Best Foods for You	Best idea us	Positive	1.00	0.300000

В первом датасете слишком много лишних колонок, убираем их

In [103]:
```
googleplaystore = googleplaystore.dropna() #Убираем строки с пропущенными значениями
```

Убираем столбцы

In [104]:
```
googleplaystore = googleplaystore.drop(['Reviews', 'Size', 'Installs', 'Price', 'Content Rating', 'Genres', 'Last Updated', 'And
roid Ver'], 1)
```

In [105]:
```
googleplaystore.head()
```

Out[105]:

	App	Category	Rating	Type	Current Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	Free	1.0.0
1	Coloring book moana	ART_AND_DESIGN	3.9	Free	2.0.0
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	Free	1.2.4
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	Free	Varies with device
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	Free	1.1

In [106]:
```
userreviews = userreviews.dropna() #Убираем строки с пропущенными значениями
```

In [107]:
```
userreviews = userreviews.drop(['Sentiment_Subjectivity'], 1)
```

In [109]:
```
userreviews.head()
```

Out[109]:

	App	Translated_Review	Sentiment	Sentiment_Polarity
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40
4	10 Best Foods for You	Best idea us	Positive	1.00
5	10 Best Foods for You	Best way	Positive	1.00

Прикрепляем к датасету с играми отзывы от этих игр по ключу App

In [110]:
```
# %%timeit
# Добавили "App", "Translated_Review", "Sentiment" к основной таблице
mergeOne = googleplaystore.merge(userreviews[["App", "Translated_Review", "Sentiment"]], on="App")
mergeOne.head(5)
```

Out[110]:

	App	Category	Rating	Type	Current Ver	Translated_Review	Sentiment
0	Coloring book moana	ART_AND_DESIGN	3.9	Free	2.0.0	A kid's excessive ads. The types ads allowed a...	Negative
1	Coloring book moana	ART_AND_DESIGN	3.9	Free	2.0.0	It bad >:(	Negative
2	Coloring book moana	ART_AND_DESIGN	3.9	Free	2.0.0	like	Neutral
3	Coloring book moana	ART_AND_DESIGN	3.9	Free	2.0.0	I love colors inspyering	Positive
4	Coloring book moana	ART_AND_DESIGN	3.9	Free	2.0.0	I hate	Negative

In [111]:
```
%%timeit # Сколько примерно времени мы на это тратим
# Добавили "App", "Translated_Review", "Sentiment" к основной таблице
googleplaystore.merge(userreviews[["App", "Translated_Review", "Sentiment"]], on="App")
```

```
22.4 ms ± 542 µs per loop (mean ± std. dev. of 7 runs, 10 loops each)
```

In [113]:
```
userreviews.groupby("App")["Sentiment_Polarity"].mean().head(5) # Группировать строчки по названию и при это считать среднее зна
чение популярности
```

Out[113]:
```
App
10 Best Foods for You               0.470733
104 找工作 - 找工作 找打工 找兼職 履歷健檢 履歷診療室    0.392405
11st                                0.185943
1800 Contacts - Lens Store          0.318145
1LINE – One Line with One Touch     0.196290
Name: Sentiment_Polarity, dtype: float64
```

In [115]:
```
%%timeit # Проверяем примерное время выполнение данной операции
userreviews.groupby("App")["Sentiment_Polarity"].mean()
```

```
3.47 ms ± 46 µs per loop (mean ± std. dev. of 7 runs, 100 loops each)
```