Московский государственный технический университет им. Н.Э. Баумана
Кафедра «Системы обработки информации и управления»

Лабораторная работа №2
по дисциплине
«Методы машинного обучения»
на тему
«Изучение библиотек обработки данных»

Выполнил:
студент группы ИУ5-24М
Лещев А. О.

Москва — 2019 г.

1. Цель лабораторной работы

Изучить библиотеки обработки данных Pandas и PandaSQL [1].

2. Задание

Задание состоит из двух частей [1].

2.1. Часть 1

Требуется выполнить первое демонстрационное задание под названием «Exploratory data analysis with Pandas» со страницы курса mlcourse.ai.

2.2. Часть 2

Требуется выполнить следующие запросы с использованием двух различных библиотек — Pandas и PandaSQL:

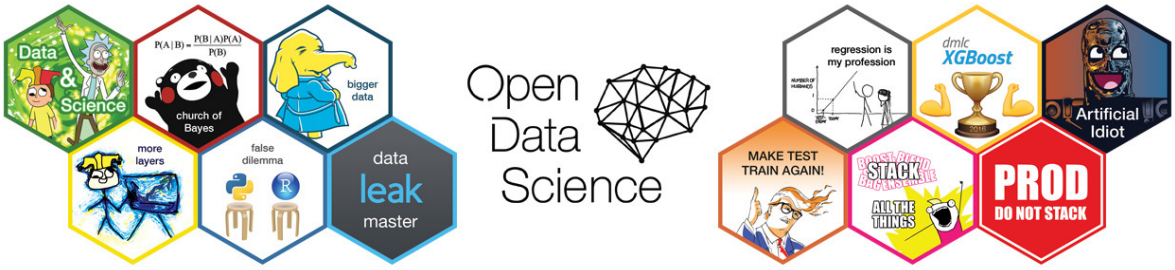
- один произвольный запрос на соединение двух наборов данных,
- один произвольный запрос на группировку набора данных с использованием функций агрегирования.

Также требуется сравнить время выполнения каждого запроса в Pandas и PandaSQL.

3. Ход выполнения работы

3.1. Часть 1

Ниже приведён демонстрационный Jupyter-ноутбук «Exploratory data analysis with Pandas» курса mlcourse.ai (файл `assignment01_pandas_uci_adult.ipynb`). Все пояснения приведены на исходном языке ноутбука — на английском.



mlcourse.ai – Open Machine Learning Course

Author: Yury Kashnitskiy. Translated and edited by Sergey Isaev, Artem Trunov, Anastasia Manokhina, and Yuanyuan Pao This material is subject to the terms and conditions of the Creative Commons CC BY-NC-SA 4.0 license. Free use is permitted for any non-commercial purpose.

Assignment #1 (demo)

Exploratory data analysis with Pandas

In this task you should use Pandas to answer a few questions about the Adult dataset.

Unique values of all features (for more information, please see the links above):

- `age`: continuous.
- `workclass`: `Private`, `Self-emp-not-inc`, `Self-emp-inc`, `Federal-gov`, `Local-gov`, `State-gov`, `Without-pay`, `Never-worked`.
- `fnlwgt`: continuous.
- `education`: `Bachelors`, `Some-college`, `11th`, `HS-grad`, `Prof-school`, `Assoc-acdm`, `Assoc-voc`, `9th`, `7th-8th`, `12th`, `Masters`, `1st-4th`, `10th`, `Doctorate`, `5th-6th`, `Preschool`.
- `education-num`: continuous.
- `marital-status`: `Married-civ-spouse`, `Divorced`, `Never-married`, `Separated`, `Widowed`, `Married-spouse-absent`, `Married-AF-spouse`.
- `occupation`: `Tech-support`, `Craft-repair`, `Other-service`, `Sales`, `Exec-managerial`, `Prof-specialty`, `Handlers-cleaners`, `Machine-op-inspct`, `Adm-clerical`, `Farming-fishing`, `Transport-moving`, `Priv-house-serv`, `Protective-serv`, `Armed-Forces`.
- `relationship`: `Wife`, `Own-child`, `Husband`, `Not-in-family`, `Other-relative`, `Unmarried`.
- `race`: `White`, `Asian-Pac-Islander`, `Amer-Indian-Eskimo`, `Other`, `Black`.
- `sex`: `Female`, `Male`.
- `capital-gain`: continuous.
- `capital-loss`: continuous.
- `hours-per-week`: continuous.
- `native-country`: `United-States`, `Cambodia`, `England`, `Puerto-Rico`, `Canada`, `Germany`, `Outlying-US(Guam-USVI-etc)`, `India`, `Japan`, `Greece`, `South`, `China`, `Cuba`, `Iran`, `Honduras`, `Philippines`, `Italy`, `Poland`, `Jamaica`, `Vietnam`, `Mexico`, `Portugal`, `Ireland`, `France`, `Dominican-Republic`, `Laos`, `Ecuador`, `Taiwan`, `Haiti`, `Columbia`, `Hungary`, `Guatemala`, `Nicaragua`, `Scotland`, `Thailand`, `Yugoslavia`, `El-Salvador`, `Trinadad&Tobago`, `Peru`, `Hong`, `Holand-Netherlands`.

- `salary`: `>50K`, `<=50K`.

Importing all required packages:

```
In [1]: import pandas as pd
```

Setting maximum display width for text report [2]:

```
In [2]: pd.set_option("display.width", 70)
```

Loading data:

```
In [3]: data = pd.read_csv('adult.data.csv')
        data.head()
```

```
Out[3]:    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

              marital-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

             sex  capital-gain  capital-loss  hours-per-week  \
        0   Male          2174             0              40
        1   Male             0             0              13
        2   Male             0             0              40
        3   Male             0             0              40
        4 Female             0             0              40

          native-country salary
        0  United-States  <=50K
        1  United-States  <=50K
        2  United-States  <=50K
        3  United-States  <=50K
        4           Cuba  <=50K
```

1. How many men and women (sex feature) are represented in this dataset?

```
In [4]: data["sex"].value_counts()

Out[4]: Male      21790
        Female    10771
        Name: sex, dtype: int64
```

2. What is the average age (age feature) of women?

```
In [5]: data[data["sex"] == "Female"]["age"].mean()

Out[5]: 36.85823043357163
```

3. What is the percentage of German citizens (native-country feature)?

```
In [6]: print("{0:%}".format(data[data["native-country"] == "Germany"].size
                            / data.size))

0.420749%
```

4-5. What are the mean and standard deviation of age for those who earn more than 50K per year (salary feature) and those who earn less than 50K per year?

```
In [7]: ages1 = data[data["salary"] == "<=50K"]["age"]
        ages2 = data[data["salary"] ==  ">50K"]["age"]
        print("<=50K: = {0} ± {1} years".format(ages1.mean(), ages1.std()))
        print(" >50K: = {0} ± {1} years".format(ages2.mean(), ages2.std()))
```

```
<=50K: = 36.78373786407767 ± 14.02008849082488 years
 >50K: = 44.24984058155847 ± 10.519027719851826 years
```

6. Is it true that people who earn more than 50K have at least high school education? (`education` – `Bachelors`, `Prof-school`, `Assoc-acdm`, `Assoc-voc`, `Masters` or `Doctorate` feature)

```
In [8]: high_educations = set(["Bachelors", "Prof-school", "Assoc-acdm",
                               "Assoc-voc", "Masters", "Doctorate"])
        def high_educated(e):
            return e in high_educations
        data[data["salary"] == ">50K"]["education"].map(high_educated).all()
```

```
Out[8]: False
```

7. Display age statistics for each race (`race` feature) and each gender (`sex` feature). Use `groupby()` and `describe()`. Find the maximum age of men of `Amer-Indian-Eskimo` race.

```
In [9]: data.groupby(["race", "sex"])["age"].describe()
```

```
Out[9]:                                count       mean        std   min  \
        race                 sex
        Amer-Indian-Eskimo   Female    119.0  37.117647  13.114991  17.0
                             Male      192.0  37.208333  12.049563  17.0
        Asian-Pac-Islander   Female    346.0  35.089595  12.300845  17.0
                             Male      693.0  39.073593  12.883944  18.0
        Black                Female   1555.0  37.854019  12.637197  17.0
                             Male     1569.0  37.682600  12.882612  17.0
        Other                Female    109.0  31.678899  11.631599  17.0
                             Male      162.0  34.654321  11.355531  17.0
        White                Female   8642.0  36.811618  14.329093  17.0
                             Male    19174.0  39.652498  13.436029  17.0

                                       25%   50%    75%   max
        race                 sex
        Amer-Indian-Eskimo   Female   27.0  36.0  46.00  80.0
                             Male     28.0  35.0  45.00  82.0
        Asian-Pac-Islander   Female   25.0  33.0  43.75  75.0
                             Male     29.0  37.0  46.00  90.0
        Black                Female   28.0  37.0  46.00  90.0
                             Male     27.0  36.0  46.00  90.0
        Other                Female   23.0  29.0  39.00  74.0
                             Male     26.0  32.0  42.00  77.0
        White                Female   25.0  35.0  46.00  90.0
                             Male     29.0  38.0  49.00  90.0
```

```
In [10]: data[(data["race"] == "Amer-Indian-Eskimo")
              & (data["sex"] == "Male")]["age"].max()
```

```
Out[10]: 82
```

8. Among whom is the proportion of those who earn a lot (>50K) greater: married or single men (`marital-status` feature)? Consider as 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.

```
In [11]: # You code here
```

9. 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 (>50K) among them?

```
In [12]: # You code here
```

10. Count the average time of work (`hours-per-week`) for those who earn a little and a lot (`salary`) for each country (`native-country`). What will these be for Japan?

```
In [13]: # You code here
```

3.2. Часть 2

```
In [ ]:
```

Список литературы

[1] Гапанюк Ю. Е. Лабораторная работа «Изучение библиотек обработки данных» [Электронный ресурс] // GitHub. — 2019. — Режим доступа: `https://github.com/ugapanyuk/ml_course/wiki/LAB_PANDAS` (дата обращения: 20.02.2019).

[2] pandas 0.24.1 documentation [Electronic resource] // PyData. — 2019. — Access mode: `http://pandas.pydata.org/pandas-docs/stable/` (online; accessed: 20.02.2019).