**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Технологии машинного обучения»

Отчет по лабораторной работе №2

Выполнил:  
Студент группы ИУ5-62Б   
Андреев Илья

Подпись и дата:

Проверил:  
преподаватель каф. ИУ5   
Гапанюк Ю.Е.

Подпись и дата:

г. Москва, 2020 г.

# Лабораторная работа 2

## Изучение библиотек обработки данных.

**Цель лабораторной работы:** изучение библиотеки обработки данных Pandas.

### Задание:

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

Условие задания - <https://nbviewer.jupyter.org/github/Yorko/mlcourse_open/blob/master/jupyter_english/assignments_demo/assignment01_pandas_uci_adult.ipynb?flush_cache=true>

Официальный датасет находится здесь, но данные и заголовки хранятся отдельно, что неудобно для анализа - <https://archive.ics.uci.edu/ml/datasets/Adult>

Поэтому готовый набор данных для лабораторной работы удобнее скачать здесь - <https://raw.githubusercontent.com/Yorko/mlcourse.ai/master/data/adult.data.csv> (удобнее всего нажать на данной ссылке правую кнопку мыши и выбрать в контекстном меню пункт "сохранить ссылку", будет предложено сохранить файл в формате CSV)

Пример решения задания - <https://www.kaggle.com/kashnitsky/a1-demo-pandas-and-uci-adult-dataset-solution>

Набор упражнений по Pandas с решениями - <https://github.com/guipsamora/pandas_exercises>

**import** **numpy** **as** **np**

**import** **pandas** **as** **pd**

pd.set\_option('display.max.columns', 100)

%**matplotlib** inline

**import** **matplotlib.pyplot** **as** **plt**

**import** **seaborn** **as** **sns**

**import** **warnings**

warnings.filterwarnings('ignore')

In [2]:

data = pd.read\_csv('adult\_data.csv')

In [3]:

data.head()

Out[3]:

|  | **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 |

### 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.loc[data.sex == 'Female'].age.mean()

Out[5]:

36.85823043357163

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

In [6]:

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

Out[6]:

0.004207487485028101

### 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]:

age1 = data.loc[data.salary == '<=50K'].age

age2 = data.loc[data.salary == '>50K'].age

print ('<=50K ', age1.mean(),'+-', age1.std())

print ('>50K ',age2.mean(),'+-', age2.std())

<=50K 36.78373786407767 +- 14.020088490824813

>50K 44.24984058155847 +- 10.51902771985177

### 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]:

data.loc[data.salary == '>50K'].education.value\_counts()

Out[8]:

Bachelors 2221

HS-grad 1675

Some-college 1387

Masters 959

Prof-school 423

Assoc-voc 361

Doctorate 306

Assoc-acdm 265

10th 62

11th 60

7th-8th 40

12th 33

9th 27

5th-6th 16

1st-4th 6

Name: education, dtype: int64

### 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** | **25%** | **50%** | **75%** | **max** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **race** | **sex** |  |  |  |  |  |  |  |  |
| **Amer-Indian-Eskimo** | **Female** | 119.0 | 37.117647 | 13.114991 | 17.0 | 27.0 | 36.0 | 46.00 | 80.0 |
| **Male** | 192.0 | 37.208333 | 12.049563 | 17.0 | 28.0 | 35.0 | 45.00 | 82.0 |
| **Asian-Pac-Islander** | **Female** | 346.0 | 35.089595 | 12.300845 | 17.0 | 25.0 | 33.0 | 43.75 | 75.0 |
| **Male** | 693.0 | 39.073593 | 12.883944 | 18.0 | 29.0 | 37.0 | 46.00 | 90.0 |
| **Black** | **Female** | 1555.0 | 37.854019 | 12.637197 | 17.0 | 28.0 | 37.0 | 46.00 | 90.0 |
| **Male** | 1569.0 | 37.682600 | 12.882612 | 17.0 | 27.0 | 36.0 | 46.00 | 90.0 |
| **Other** | **Female** | 109.0 | 31.678899 | 11.631599 | 17.0 | 23.0 | 29.0 | 39.00 | 74.0 |
| **Male** | 162.0 | 34.654321 | 11.355531 | 17.0 | 26.0 | 32.0 | 42.00 | 77.0 |
| **White** | **Female** | 8642.0 | 36.811618 | 14.329093 | 17.0 | 25.0 | 35.0 | 46.00 | 90.0 |
| **Male** | 19174.0 | 39.652498 | 13.436029 | 17.0 | 29.0 | 38.0 | 49.00 | 90.0 |

### 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 [10]:

print(data.loc[(data['sex'] == 'Male') & (data['marital-status'].isin(['Never-married', 'Separated',

'Divorced',

'Widowed']))].salary.value\_counts())

print(data.loc[(data['sex'] == 'Male') &

(data['marital-status'].str.startswith('Married'))].salary.value\_counts())

<=50K 7552

>50K 697

Name: salary, dtype: int64

<=50K 7576

>50K 5965

Name: salary, dtype: int64

### 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 [11]:

max\_h = data['hours-per-week'].max()

people\_count = (data['hours-per-week'] == max\_h).sum()

percent = ((((data[(data['hours-per-week'] == max\_h) & (data.salary == '>50K')]).shape[0])/people\_count)\*100).round()

print('Максимальное количество часов = **{0}**. Людей, которые работают столько же = **{1}**. Процент людей, зарабатывающих >50К = **{2}**%'.format(max\_h, people\_count, percent))

Максимальное количество часов = 99. Людей, которые работают столько же = 85. Процент людей, зарабатывающих >50К = 29.0%

### 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 [12]:

data.groupby(['native-country','salary'])['hours-per-week'].mean().head(20)

Out[12]:

native-country salary

? <=50K 40.164760

>50K 45.547945

Cambodia <=50K 41.416667

>50K 40.000000

Canada <=50K 37.914634

>50K 45.641026

China <=50K 37.381818

>50K 38.900000

Columbia <=50K 38.684211

>50K 50.000000

Cuba <=50K 37.985714

>50K 42.440000

Dominican-Republic <=50K 42.338235

>50K 47.000000

Ecuador <=50K 38.041667

>50K 48.750000

El-Salvador <=50K 36.030928

>50K 45.000000

England <=50K 40.483333

>50K 44.533333

Name: hours-per-week, dtype: float64