Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления»
Кафедра «Автоматизированные системы обработки информации и управления»

Отчет
Лабораторная работа № 2

По курсу «Технологии машинного обучения»

«Изучение библиотек обработки данных»

ИСПОЛНИТЕЛЬ:

Тарасов Владислав
Группа ИУ5-64

____

"__"__________2020 г.

ПРЕПОДАВАТЕЛЬ:

Гапанюк Ю.Е.

____

"__"__________2020 г.

Москва 2020

1. Цель работы

Изучение библиотеки обработки данных Pandas.

2. Описание задания

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

3. Текст программы и экранные формы с примерами выполнения

```
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
```

In [2]:

```
import numpy as np
import pandas as pd
pd.set_option('display.max.columns', 100)
# to draw pictures in jupyter notebook
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
# we don't like warnings
# you can comment the following 2 lines if you'd like to
import warnings
warnings.filterwarnings('ignore')
```

In [3]:

```
data = pd.read_csv('../data/adult.csv')
data.head()
```

Out[3]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	I
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	I
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	I
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	I
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Fe

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 [6]:

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

Out[6]:

```
36.85823043357163
```

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

In [29]:

```
(data[data['native-country'] == 'Germany'].shape[0] / data.shape[0]) * 100
```

Out[29]:

```
0.42074874850281013
```

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 [27]:

```
more50 = data[data['salary'] == '>50K']['age']
less50 = data[data['salary'] == '<=50K']['age']

print("More than 50K mean: {0} std: {1}".format(more50.mean().round(3), more50.std()
print("Less than 50K mean: {0} std: {1}".format(less50.mean().round(3), less50.std()
```

```
More than 50K mean: 44.25 std: 10.519
Less than 50K mean: 36.784 std: 14.02
```

```
**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 [36]:

```
def isInHigh(e):
    return e in ['Bachelors', 'Prof-school', 'Assoc-acdm', 'Assoc-voc', 'Masters', '

more50 = data[data['salary'] == '>50K']['education']
more50.map(isInHigh).all()
```

Out[36]:

```
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 [48]:

```
data.groupby(['race', 'sex'])['age'].describe()
```

Out[48]:

race	sex	count	mean	std	min	25%	50%	75%	max
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 [63]:

```
married = ['Married-civ-spouse', 'Married-spouse-absent', 'Married-AF-spouse']

def isMarried(m):
    return m in married

stat = data[(data['sex'] == 'Male') & (data['salary'] == '>50K')]['marital-status'].

print('Married count:', stat[1], 'single count', stat[0])
```

```
Married count: 5965 single count 697
```

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 [83]:

```
max = data['hours-per-week'].max()
count = data[data['hours-per-week'] == max].shape[0]
percentage = (data[(data['hours-per-week'] == max) & (data['salary'] == '>50K')].sha

print('max:,', max, 'count:', count, 'percentage:', percentage)
```

```
max:, 99 count: 85 percentage: 29.411764705882355
```

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 [98]:

```
data.groupby(['native-country', 'salary'])['hours-per-week'].describe().unstack()[['
```

Out[98]:

	mean	
salary	**<=50K**	**>50K**
native-country		
?	40.164760	45.547945
Cambodia	41.416667	40.000000
Canada	37.914634	45.641026
China	37.381818	38.900000
Columbia	38.684211	50.000000
Cuba	37.985714	42.440000
Dominican-Republic	42.338235	47.000000
Ecuador	38.041667	48.750000
El-Salvador	36.030928	45.000000
England	40.483333	44.533333
France	41.058824	50.750000
Germany	39.139785	44.977273
Greece	41.809524	50.625000
Guatemala	39.360656	36.666667
Haiti	36.325000	42.750000
Holand-Netherlands	40.000000	0.000000
Honduras	34.333333	60.000000
Hong	39.142857	45.000000
Hungary	31.300000	50.000000
India	38.233333	46.475000
Iran	41.440000	47.500000
Ireland	40.947368	48.000000
Italy	39.625000	45.400000
Jamaica	38.239437	41.100000
Japan	41.000000	47.958333
Laos	40.375000	40.000000
Mexico	40.003279	46.575758
Nicaragua	36.093750	37.500000
Outlying-US(Guam-USVI-etc)	41.857143	0.000000
Peru	35.068966	40.000000
Philippines	38.065693	43.032787

	mean	
salary	<=50K	>50K
native-country		
Poland	38.166667	39.000000
Portugal	41.939394	41.500000
Puerto-Rico	38.470588	39.416667
Scotland	39.444444	46.666667
South	40.156250	51.437500
Taiwan	33.774194	46.800000
Thailand	42.866667	58.333333
Trinadad&Tobago	37.058824	40.000000
United-States	38.799127	45.505369
Vietnam	37.193548	39.200000
Yugoslavia	41.600000	49.500000