

**Московский государственный технический  
университет им. Н. Э. Баумана**  
**Факультет «Информатика и системы управления»**

Кафедра «Системы обработки информации и управления»  
Курс «Технологии машинного обучения»

Отчет по лабораторной работе №2  
Изучение библиотек обработки данных.

Группа: РТ5-61  
Студент: Савушкин Д.А.  
Преподаватель: Гапанюк Ю.Е.

Москва, 2020 г.

**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>

**Текст программы и экранные формы с примерами выполнения программы:**

```
[ ] import numpy as np
import pandas as pd

[ ] data = pd.read_csv('adult.data.txt')
data.head()

[ ] data['sex'].value_counts()

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

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

[ ] ages1 = data.loc[data['salary'] == '>50K', 'age']
ages2 = data.loc[data['salary'] == '<=50K', 'age']
print("Средний возраст богатых: {} +- {} лет, бедных - {} +- {} лет.".format(
    round(ages1.mean()), round(ages1.std(), 1),
    round(ages2.mean()), round(ages2.std(), 1)))

[ ] Средний возраст богатых: 44 +- 10.5 лет, бедных - 37 +- 14.0 лет.
```

```
[ ] data.loc[data['salary'] == '>50K', 'education'].unique() # No

[ ] 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)

[ ] for (race, sex), sub_df in data.groupby(['race', 'sex']):
    print("Paca: {0}, пол: {1}".format(race, sex))
    print(sub_df['age'].describe())

[ ] 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
[ ] Paca: Asian-Pac-Islander, пол: Female
[ ] count      346.000000
[ ] mean       35.089595
[ ] std        12.300845
[ ] min        17.000000
[ ] 25%        25.000000
[ ] 50%        33.000000
[ ] 75%        43.750000

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

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

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

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

```
▶ data['marital-status'].value_counts()
```

```
↳ 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
```

```
[ ] max_load = data['hours-per-week'].max()
print("Максимальное время - {0} часов/неделя".format(max_load))

num_workaholics = data[data['hours-per-week'] == max_load].shape[0]
print("Общее количество трудоголиков {0}".format(num_workaholics))

rich_share = float(data[(data['hours-per-week'] == max_load) & (data['salary'] == '>50K')].shape[0]) / num_workaholics
print("Процент богатых из них {0}%".format(int(100 * rich_share)))
```

```
↳ Максимальное время - 99 часов/неделя
    Общее количество трудоголиков 85
    Процент богатых из них 29%
```

```
[ ] 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
```

```
[ ] pd.crosstab(data['native-country'], data['salary'],
                 values=data['hours-per-week'], aggfunc=np.mean).T
```

native-country	?	Cambodia	Canada	China	Columbia	Cuba	Dominican-Republic	Ecuador	El-Salvador	England	France	Germany	Greece	
salary	<=50K	40.164760	41.416667	37.914634	37.381818	38.684211	37.985714	42.338235	38.041667	36.030928	40.483333	41.058824	39.139785	41.809524
	>50K	45.547945	40.000000	45.641026	38.900000	50.000000	42.440000	47.000000	48.750000	45.000000	44.533333	50.750000	44.977273	50.625000