



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ \_\_\_\_\_ Информатика и системы управления

КАФЕДРА \_\_\_\_\_ Системы обработки информации и управления

## Отчёт по лабораторной работе №1

По дисциплине:  
«Технологии машинного обучения»

Выполнил:

Студент группы ИУ5Ц-81Б

\_\_\_\_\_

(Подпись, дата)

**Бабакулев Ф.А**

(Фамилия И.О.)

Проверил:

\_\_\_\_\_

(Подпись, дата)

**Гапанюк Ю. Е.**

(Фамилия И.О.)

Москва, 2022

## **Задание**

- Выбрать набор данных
- Создать ноутбук, который содержит следующие разделы:
  1. Текстовое описание набора данных
  2. Основные характеристики набора данных
  3. Визуальное исследование набора данных
  4. Информацию о корреляции признаков

## ЛР №1

### Импорт библиотек

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from pandas.plotting import scatter_matrix
import warnings
warnings.filterwarnings('ignore')
sns.set(style="ticks")
%matplotlib inline
```

### Загрузка данных

```
In [2]: happy_data = pd.read_csv('StudentsPerformance.csv', sep = ',')
```

## 2) Основные характеристики датасета

```
In [3]: # Первые пять строк датасета
happy_data.head()
```

Out[3]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

```
In [4]: # Размер датасета
happy_data.shape
```

Out[4]: (1000, 8)

```
In [5]: # Количество нулевых элементов
happy_data.isnull().sum()
```

```
Out[5]: gender                0
race/ethnicity               0
parental level of education  0
lunch                       0
test preparation course      0
math score                   0
reading score                 0
writing score                 0
dtype: int64
```

```
In [6]: # Колонки и их типы данных
happy_data.dtypes
```

```
Out[6]: gender                object
race/ethnicity               object
parental level of education  object
lunch                       object
test preparation course      object
math score                   int64
reading score                 int64
writing score                 int64
dtype: object
```

```
In [7]: # Описание датасета
happy_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 8 columns):
gender                1000 non-null object
race/ethnicity         1000 non-null object
parental level of education  1000 non-null object
lunch                 1000 non-null object
test preparation course  1000 non-null object
math score             1000 non-null int64
reading score          1000 non-null int64
writing score          1000 non-null int64
dtypes: int64(3), object(5)
memory usage: 62.6+ KB
```

```
In [8]: # Статистические данные
happy_data.describe()
```

```
Out[8]:
```

	math score	reading score	writing score
count	1000.000000	1000.000000	1000.000000
mean	66.089000	69.189000	68.054000
std	15.16308	14.800192	15.195657
min	0.000000	17.000000	10.000000
25%	57.000000	59.000000	57.750000
50%	66.000000	70.000000	69.000000
75%	77.000000	79.000000	79.000000
max	100.000000	100.000000	100.000000

```
In [9]: # Удаляем столбец Lunch
happy_data = happy_data.drop('lunch', axis = 1)
```

```
In [10]: # Первые пять строк датасета
happy_data.head()
```

```
Out[10]:
```

	gender	race/ethnicity	parental level of education	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	none	72	72	74
1	female	group C	some college	completed	69	90	88
2	female	group B	master's degree	none	90	95	93
3	male	group A	associate's degree	none	47	57	44
4	male	group C	some college	none	76	78	75

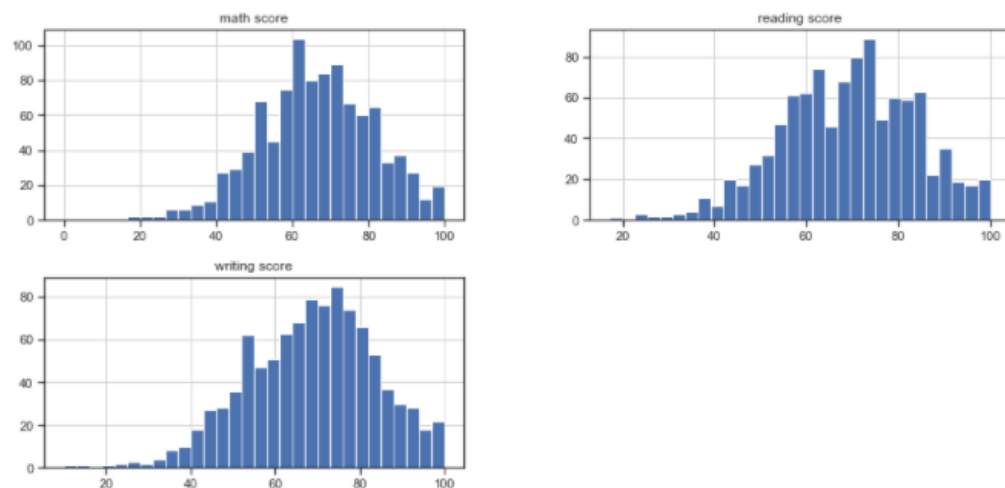
```
In [11]: # Определим уникальные значения для целевого признака
happy_data['race/ethnicity'].unique()
```

```
Out[11]: array(['group B', 'group C', 'group A', 'group D', 'group E'],
      dtype=object)
```

### 3) Визуальное исследование датасета

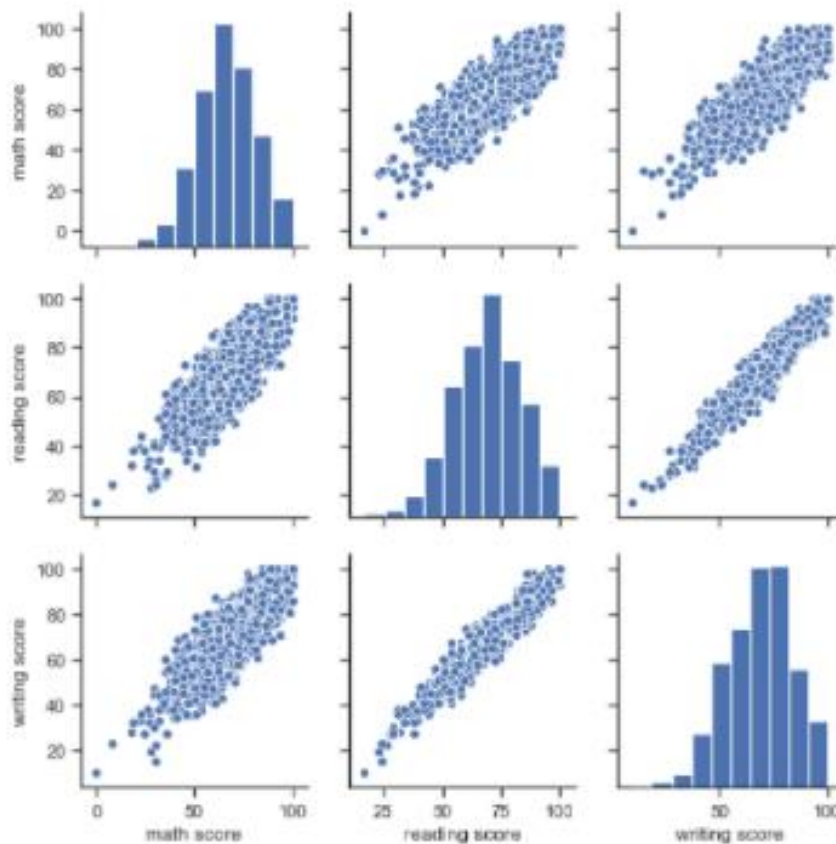
```
In [12]: # Гистограммы для всех признаков
happy_data.hist(bins=30, figsize = (15,7))
```

```
Out[12]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x00000217221F2F60>,
      <matplotlib.axes._subplots.AxesSubplot object at 0x0000021722492B70>],
      <matplotlib.axes._subplots.AxesSubplot object at 0x00000217224C6208>,
      <matplotlib.axes._subplots.AxesSubplot object at 0x00000217224E4860>]],
      dtype=object)
```



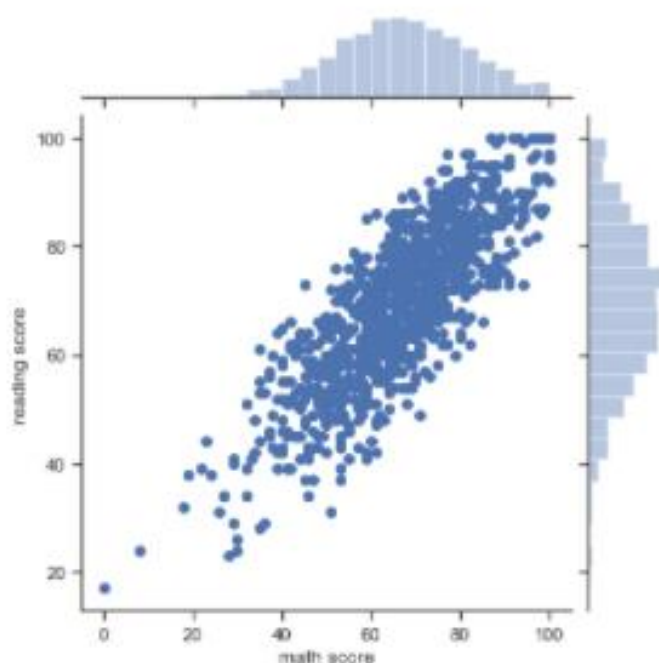
```
In [13]: # Диаграммы рассеяния для всех признаков
plt.figure(figsize=(12,6))
sns.pairplot(happy_data)
```

```
Out[13]: <seaborn.axisgrid.PairGrid at 0x21722575da0>
<Figure size 864x432 with 9 Axes>
```



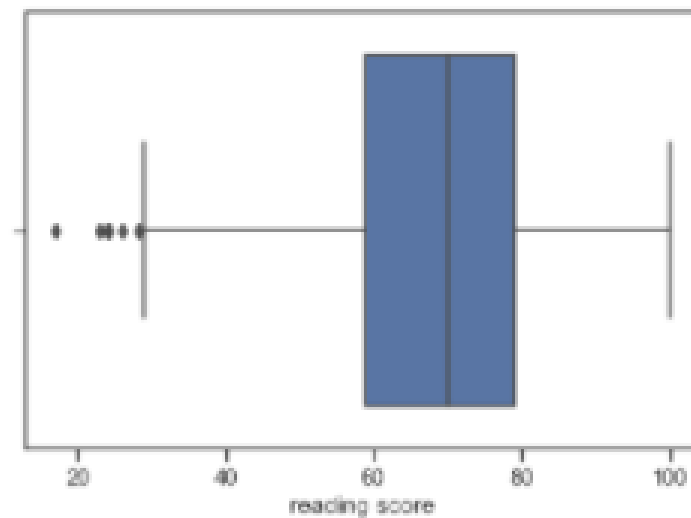
```
In [14]: # Увеличенные диаграммы рассеяния для признаков
sns.jointplot(x = "math score", y = "reading score", kind="scatter", data = happy_data)
```

```
Out[14]: <seaborn.axisgrid.JointGrid at 0x217228d1860>
```



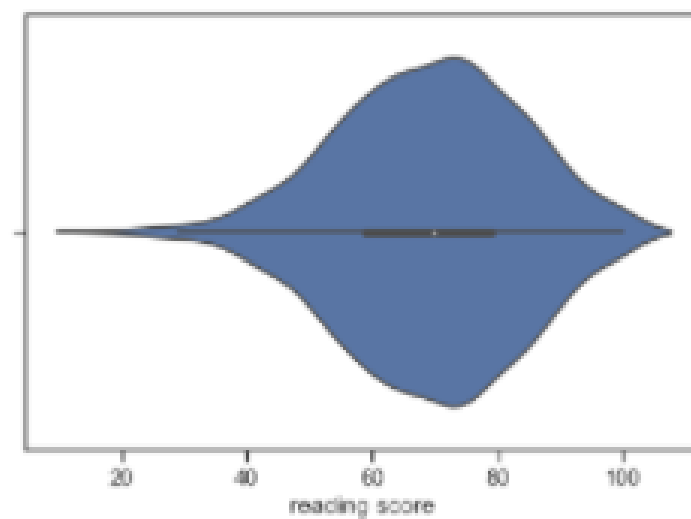
```
In [15]: # Одномерное распределение вероятности  
sns.boxplot(x=happy_data['reading score'])
```

Out[15]: <matplotlib.axes.\_subplots.AxesSubplot at 0x21723136470>



```
In [16]: #Скрипичная диаграмма  
sns.violinplot(x=happy_data['reading score'])
```

Out[16]: <matplotlib.axes.\_subplots.AxesSubplot at 0x217231a2a20>



## 4) Корреляции признаков

```
In [17]: corr_matrix = happy_data.corr()
```

```
In [18]: corr_matrix['writing score']
```

```
Out[18]: math score      0.802642  
reading score  0.954598  
writing score   1.000000  
Name: writing score, dtype: float64
```

```
In [19]: sns.heatmap(happy_data.corr(), annot=True, fmt='.3f')
```

```
Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x21722dd54e0>
```

