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 ИУ5 «Системы обработки информации и управления»

Лабораторная работа №1 по дисциплине

Технологии машинного обучения

“Разведочный анализ данных.

Исследование и визуализация данных.”

Выполнил:
студент группы ИУ5-64Б
Такташова Д.Ю.

Проверил:
Гапанюк Ю.Е.

2024 г.

Задание

1. Выбрать набор данных (датасет).

2. Создать ноутбук, который содержит следующие разделы:

- Текстовое описание выбранного Вами набора данных.

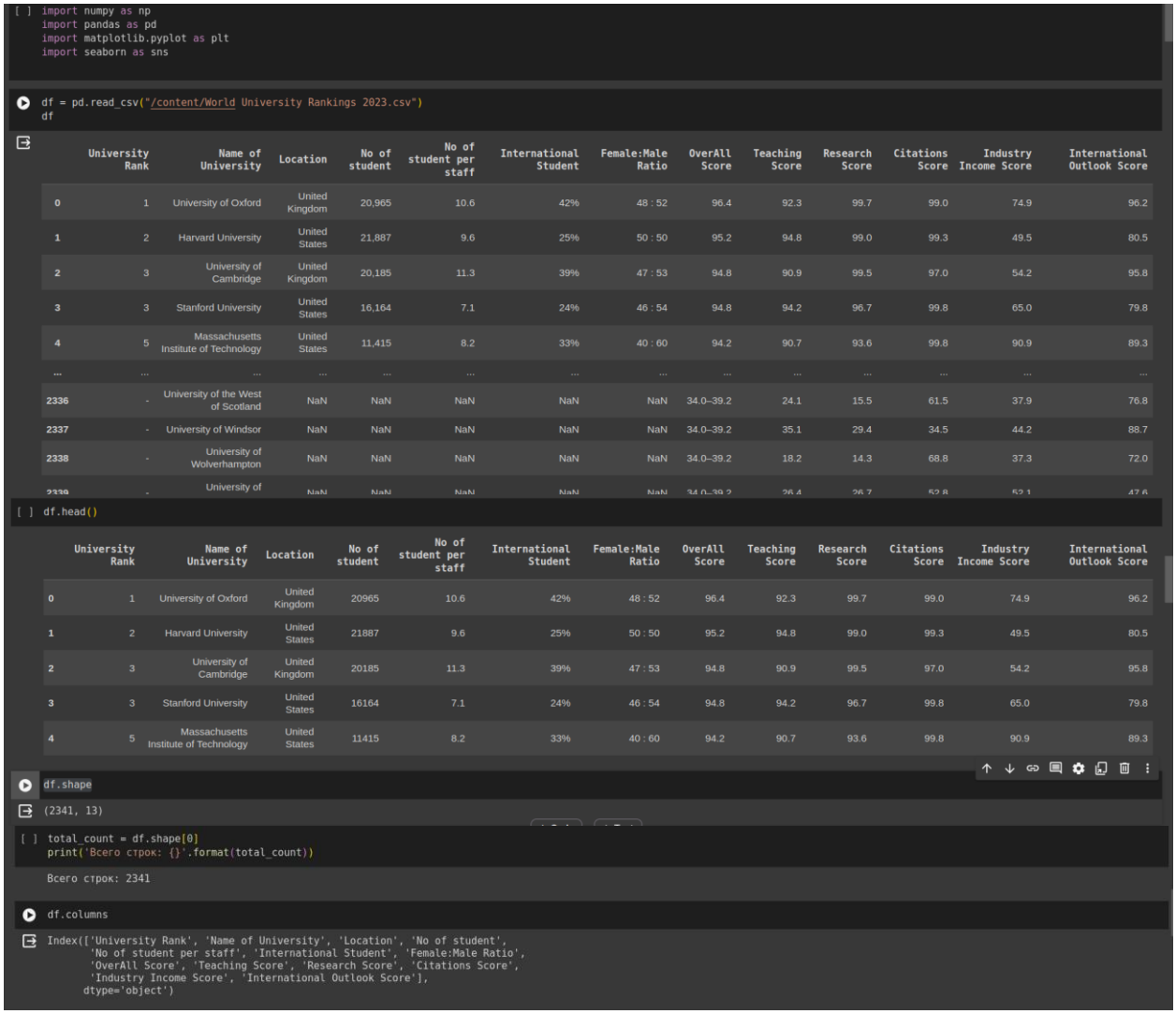
- Основные характеристики датасета.

- Визуальное исследование датасета.

- Информация о корреляции признаков.

3. Сформировать отчет и разместить его в своем репозитории на github.

Текст программы



```
[ ] import numpy as np
    import pandas as pd
    import matplotlib.pyplot as plt
    import seaborn as sns
```

```
df = pd.read_csv("/content/World University Rankings 2023.csv")
df
```

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	Female:Male Ratio	OverAll Score	Teaching Score	Research Score	Citations Score	Industry Income Score	International Outlook Score
0	1	University of Oxford	United Kingdom	20,965	10.6	42%	48 : 52	96.4	92.3	99.7	99.0	74.9	96.2
1	2	Harvard University	United States	21,887	9.6	25%	50 : 50	95.2	94.8	99.0	99.3	49.5	80.5
2	3	University of Cambridge	United Kingdom	20,185	11.3	39%	47 : 53	94.8	90.9	99.5	97.0	54.2	95.8
3	3	Stanford University	United States	16,164	7.1	24%	46 : 54	94.8	94.2	96.7	99.8	65.0	79.8
4	5	Massachusetts Institute of Technology	United States	11,415	8.2	33%	40 : 60	94.2	90.7	93.6	99.8	90.9	89.3
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2336	-	University of the West of Scotland	NaN	NaN	NaN	NaN	NaN	34.0–39.2	24.1	15.5	61.5	37.9	76.8
2337	-	University of Windsor	NaN	NaN	NaN	NaN	NaN	34.0–39.2	35.1	29.4	34.5	44.2	88.7
2338	-	University of Wolverhampton	NaN	NaN	NaN	NaN	NaN	34.0–39.2	18.2	14.3	68.8	37.3	72.0
2339	-	University of	NaN	NaN	NaN	NaN	NaN	34.0–39.2	26.4	26.7	52.8	52.1	47.6

```
[ ] df.head()
```

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	Female:Male Ratio	OverAll Score	Teaching Score	Research Score	Citations Score	Industry Income Score	International Outlook Score
0	1	University of Oxford	United Kingdom	20965	10.6	42%	48 : 52	96.4	92.3	99.7	99.0	74.9	96.2
1	2	Harvard University	United States	21887	9.6	25%	50 : 50	95.2	94.8	99.0	99.3	49.5	80.5
2	3	University of Cambridge	United Kingdom	20185	11.3	39%	47 : 53	94.8	90.9	99.5	97.0	54.2	95.8
3	3	Stanford University	United States	16164	7.1	24%	46 : 54	94.8	94.2	96.7	99.8	65.0	79.8
4	5	Massachusetts Institute of Technology	United States	11415	8.2	33%	40 : 60	94.2	90.7	93.6	99.8	90.9	89.3

```
df.shape
```

```
(2341, 13)
```

```
[ ] total_count = df.shape[0]
    print('Всего строк: {}'.format(total_count))
```

```
Всего строк: 2341
```

```
df.columns
```

```
Index(['University Rank', 'Name of University', 'Location', 'No of student',
       'No of student per staff', 'International Student', 'Female:Male Ratio',
       'OverAll Score', 'Teaching Score', 'Research Score', 'Citations Score',
       'Industry Income Score', 'International Outlook Score'],
      dtype='object')
```

```
[ ] print(df.isnull().sum())
```

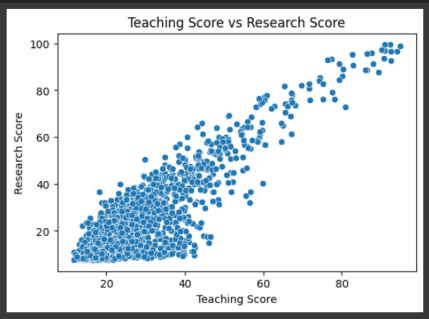
```
University Rank                  0
Name of University             108
Location                       294
No of student                  132
No of student per staff        133
International Student          132
Female:Male Ratio              213
OverAll Score                  542
Teaching Score                 542
Research Score                 542
Citations Score                542
Industry Income Score          542
International Outlook Score    542
dtype: int64
```

```
[ ] df.fillna(df.mode().iloc[0], inplace=True)
```

```
[ ] df.isna().sum()
```

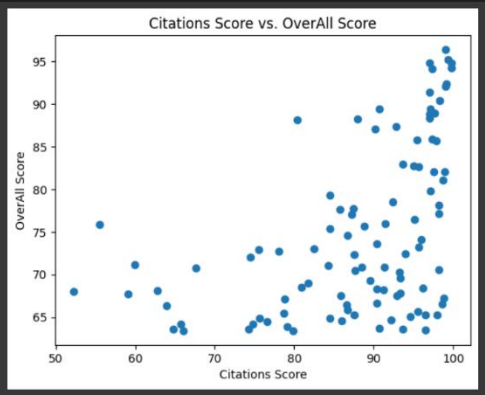
```
University Rank                0
Name of University             0
Location                       0
No of student                  0
No of student per staff        0
International Student          0
Female:Male Ratio              0
OverAll Score                  0
Teaching Score                 0
Research Score                 0
Citations Score                0
Industry Income Score          0
International Outlook Score    0
dtype: int64
```

```
[ ] # Plot a scatterplot of 'Teaching Score' vs 'Research Score'
    plt.figure(figsize=(6, 4))
    sns.scatterplot(data=df, x='Teaching Score', y='Research Score')
    plt.title('Teaching Score vs Research Score')
    plt.xlabel('Teaching Score')
    plt.ylabel('Research Score')
    plt.show()
```



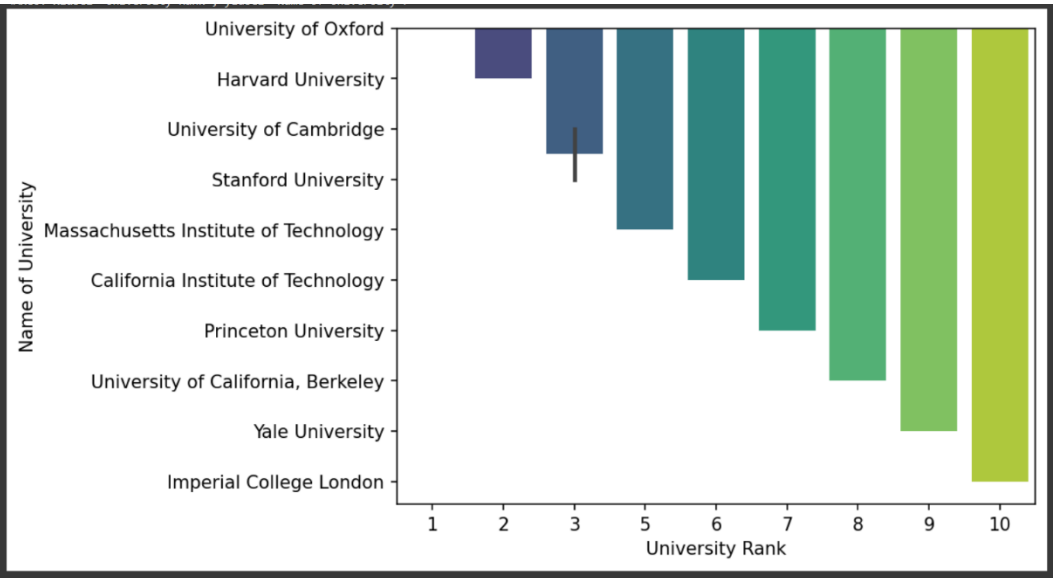

```
top_100=df.head(100)
top_100.isetitem(top_100.columns.get_loc('OverAll Score'), top_100['OverAll Score'].astype(float))

# Scatterplot for Citations Score vs. OverAll Score
plt.scatter(top_100['Citations Score'], top_100['OverAll Score'])
plt.title('Citations Score vs. OverAll Score')
plt.xlabel('Citations Score')
plt.ylabel('OverAll Score')
plt.show()
```




```
[ ] top_10_university = df.head(10)

    plt.figure(dpi=150)
    sns.barplot(data=top_10_university, x="University Rank",y="Name of University",  palette="viridis")
```

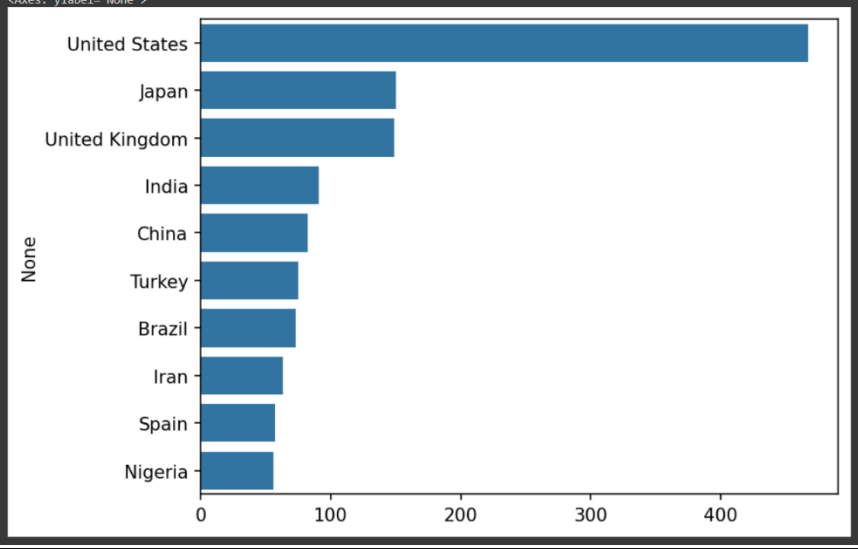



```
[ ] top_10_university_count = df["Location"].value_counts()[:10]
    top_10_university_count
```

```
United States     467
Japan             150
United Kingdom    149
India              91
China              82
Turkey             75
Brazil             73
Iran               63
Spain              57
Nigeria            56
Name: Location, dtype: int64
```

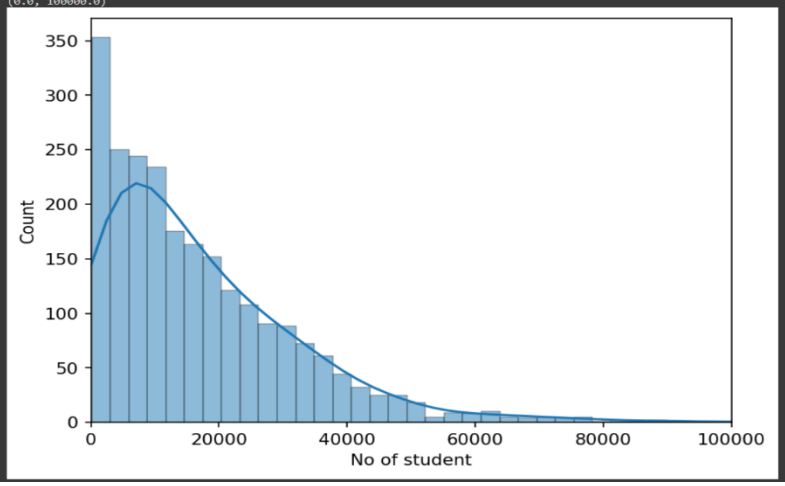
```
[ ] plt.figure(dpi=150)
    sns.barplot(y=top_10_university_count.index, x=top_10_university_count.values)
```

```
<Axes: ylabel='None'>
```




```
plt.figure(dpi=150)
sns.histplot(data=df,x="No of student", kde=True)
plt.xlim([0, 100000])
```

```
(0.0, 100000.0)
```



```
[ ]  score_cols = ['OverAll Score',
           'Teaching Score', 'Research Score', 'Citations Score',
           'Industry Income Score', 'International Outlook Score']

     plt.figure(dpi=150)
     sns.heatmap(df[score_cols].corr())
```