

Московский государственный технический университет им. Н.Э. Баумана  
Кафедра «Системы обработки информации и управления»



Лабораторная работа №1  
по дисциплине  
«Методы машинного обучения»  
на тему

**«Разведочный анализ данных. Исследование и визуализация данных»**

Выполнила:  
студентка Хэ Синьчэнь  
группы ИУ5И-24М

Москва — 2023 г.

## 1. Цель лабораторной работы

изучение различных методов визуализация данных и создание истории на основе данных.

## 2. Задание

- Выбрать набор данных (датасет). Вы можете найти список свободно распространяемых датасетов [здесь](#).

Для лабораторных работ не рекомендуется выбирать датасеты очень большого размера.

- Создать "историю о данных" в виде юпитер-ноутбука, с учетом следующих требований:
  1. История должна содержать не менее 5 шагов (где 5 - рекомендуемое количество шагов). Каждый шаг содержит график и его текстовую интерпретацию.
  2. На каждом шаге наряду с удачным итоговым графиком рекомендуется в юпитер-ноутбуке оставлять результаты предварительных "неудачных" графиков.
  3. Не рекомендуется повторять виды графиков, желательно создать 5 графиков различных видов.
  4. Выбор графиков должен быть обоснован использованием методологии data-to-viz. Рекомендуется учитывать типичные ошибки построения выбранного вида графика по методологии data-to-viz. Если методология Вами отвергается, то просьба обосновать Ваше решение по выбору графика.
  5. История должна содержать итоговые выводы. В реальных "историях о данных" именно эти выводы представляют собой основную ценность для предприятия.
- Сформировать отчет и разместить его в своем репозитории на github.

## 3. Ход выполнения работы

### 3.1. Текстовое описание набора данных

В этой тетради я буду использовать графики для визуализации взаимосвязи между переменными в наборе данных "Успеваемость студентов на экзаменах".

- **The dataset includes the following columns:**
  - Gender** (object) - The gender of the participant
    - female
    - male
- **Parental level of education** (object) - The parental level of education of the participant
  - bachelor's degree
  - some college
  - master's degree
  - associate's degree

- high school
  - some high school
- Lunch** (object) - Whether the participant is:
  - standard
  - free/ reduced
- Test preparation course** (object) - Whether the participant took the test preparation course or not
  - none
  - completed

## 3.2. Основные характеристики набора данных

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.impute import SimpleImputer
from sklearn.impute import MissingIndicator
from sklearn.impute import KNNImputer
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.model_selection import GridSearchCV
from sklearn.experimental import enable_iterative_imputer
from sklearn.impute import IterativeImputer
from IPython.display import Image
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

```
[ ] data = pd.read_csv(r'/content/data/india_population.csv')
data.head()
```

	Year	Population	Yearly % Change	Yearly Change	Migrants (net)	Median Age	Fertility Rate	Density (P/Km <sup>2</sup> )	Urban Pop %	Urban Population	Country's Share of World Pop	World Population	India Global Rank
0	2020	1380004385	0.99	13586631	-532687	28.4	2.24	464	35.0	483098640	17.70	7794798739	2
1	2019	1366417754	1.02	13775474	-532687	27.1	2.36	460	34.5	471828295	17.71	7713468100	2
2	2018	1352642280	1.04	13965495	-532687	27.1	2.36	455	34.1	460779764	17.73	7631091040	2
3	2017	1338676785	1.07	14159536	-532687	27.1	2.36	450	33.6	449963381	17.74	7547858925	2
4	2016	1324517249	1.10	14364846	-532687	27.1	2.36	445	33.2	439391699	17.75	7464022049	2

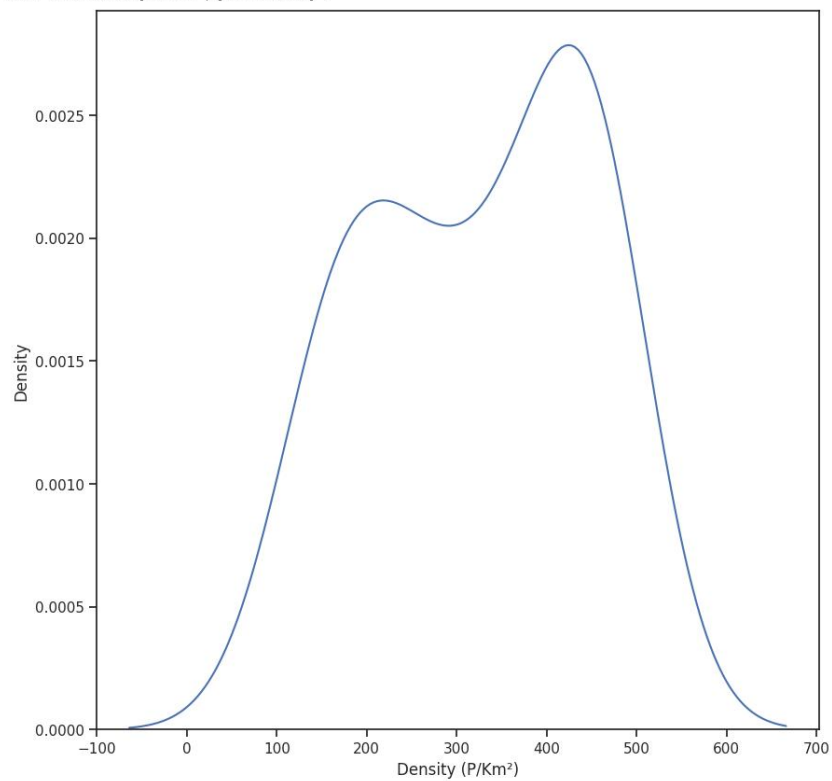
```
In [4]: data.dtypes
```

```
Out[4]: Year                int64
Population                int64
Yearly % Change           float64
Yearly Change             int64
Migrants (net)            int64
Median Age                float64
Fertility Rate            float64
Density (P/Km2)           int64
Urban Pop %              float64
Urban Population          int64
Country's Share of World Pop float64
World Population          int64
India Global Rank         int64
dtype: object
```

✓ Рассмотрим распределение рейтинга производительности CPU

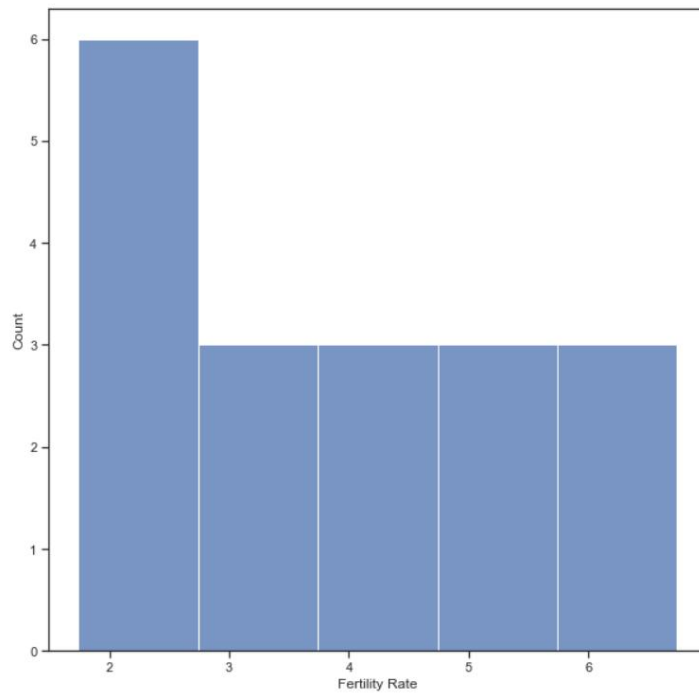
```
fig, ax = plt.subplots(figsize=(10,10))  
sns.kdeplot(data=data, x="Density (P/Km²)")
```

↳ <Axes: xlabel='Density (P/Km²)', ylabel='Density'>



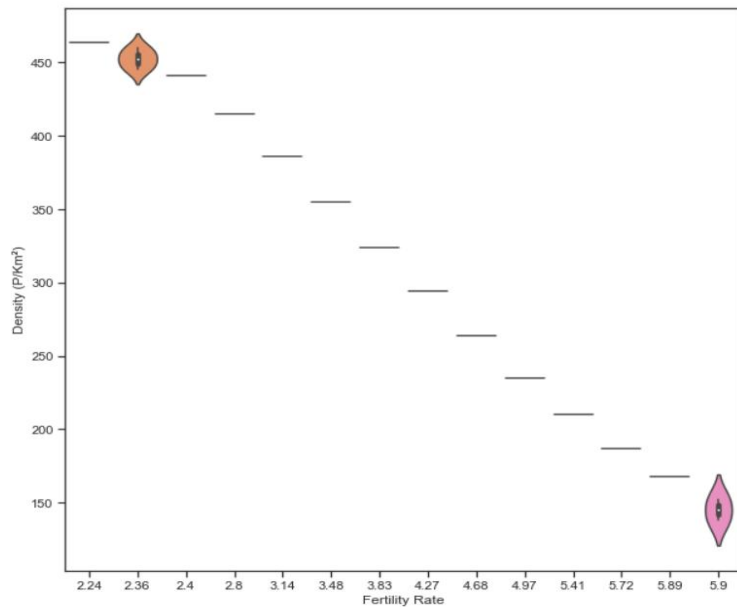
```
In [8]: fig, ax = plt.subplots(figsize=(10,10))
sns.histplot(data['Fertility Rate'], discrete=True)
```

```
Out[8]: <AxesSubplot:xlabel='Fertility Rate', ylabel='Count'>
```

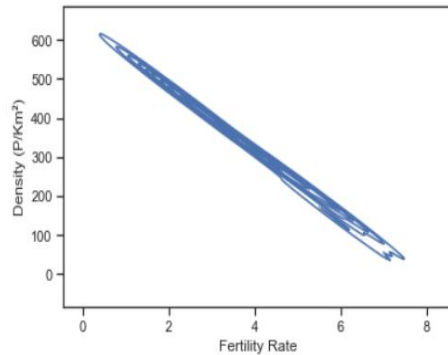


```
In [9]: fig, ax = plt.subplots(figsize=(10,10))
sns.violinplot(x='Fertility Rate', y='Density (P/Km²)', data=data)
```

```
Out[9]: <AxesSubplot:xlabel='Fertility Rate', ylabel='Density (P/Km²)'>
```

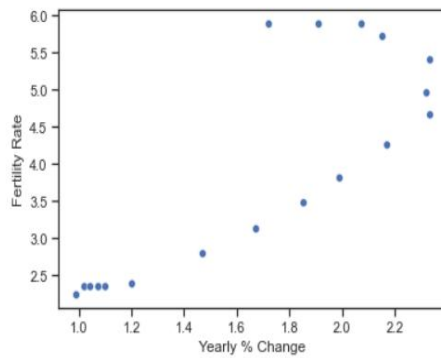


```
Out[11]: <AxesSubplot:xlabel='Fertility Rate', ylabel='Density (P/Km²)'
```



```
In [12]: sns.scatterplot(x='Yearly % Change', y='Fertility Rate', data=data)
```

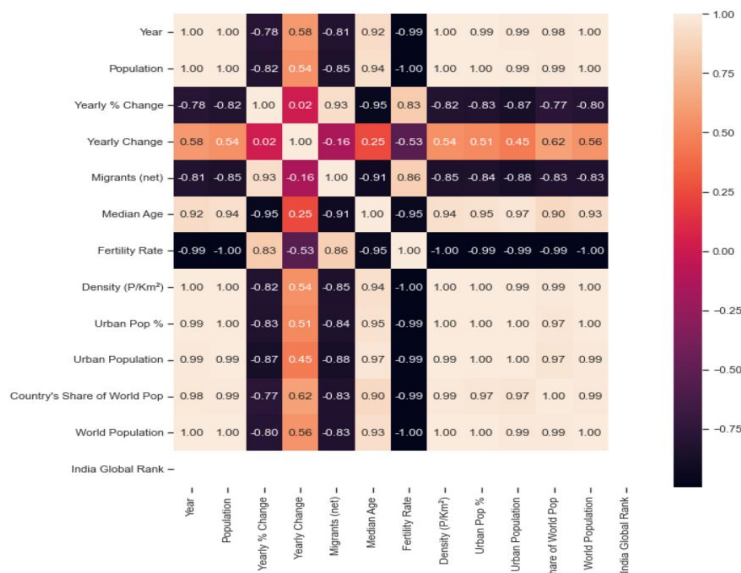
```
Out[12]: <AxesSubplot:xlabel='Yearly % Change', ylabel='Fertility Rate'>
```



```
In [13]: fig, ax = plt.subplots(figsize=(10,10))
sns.heatmap(data.corr(method='pearson'), annot=True, fmt='.2f')
fig.suptitle('Корреляционные матрицы')
```

```
Out[13]: Text(0.5, 0.98, 'Корреляционные матрицы')
```

### Корреляционные матрицы



```
In [14]: sns.pairplot(data)
plt.show()
```

