# Лабораторная работа №1 по курсу "Технологии машинного обучения"

## Выполнила Попова Дарья, студентка группы РТ5-61Б

## Разведочный анализ данных. Исследование и визуализация данных.

## Текстовое описание набора данных

В качестве датасета будем использовать игрушечный набор данных Wine recognition dataset от Scikit-learn.

В датасете приведены различные характеристики различных вин, такие как: флавоноиды, содержание алкоголя, фенолы, осадок в бокале, интенсивность цвета и другие.

Датасет состоит из 1 файла, файл содержит следующие колонки с данными:

- Alcohol (содержание спирта)
- Magnesium (содержание магния)
- Malic acid (яблочная кислота)
- Total phenols (общее содержание фенолов)
- Ash (осадок)
- Alcalinity of ash (щёлочность осадка)
- Proanthocyanins (проантоцианидины)
- Flavanoids (флавоноиды)
- Nonflavanoid phenols (нефлавоноидные фенолы)
- Color intensity (интенсивность цвета)
- Hue (оттенок)
- OD280/OD315 of diluted wines (показатель OD280/OD315 определения содержания протеинов для разбавленных вин)
- Proline (пролин)

## Загрузка библиотек и датасета

In [2]:

```
# загрузим библиотеки numpy и pandas
import numpy as np
import pandas as pd
from sklearn.datasets import *
```

In [3]:

```
# загрузим данные
wine = load_wine()
```

# Основные характеристики набора данных

In [4]:

```
# посмотрим на классификацию
wine['target_names']
```

Out[4]:

```
array(['class_0', 'class_1', 'class_2'], dtype='<U7')
```

In [5]:

```
# выведем атрибуты (названия колонок)
wine['feature_names']
```

Out[5]:

```
['alcohol',
 'malic_acid',
 'ash',
 'alcalinity_of_ash',
 'magnesium',
 'total_phenols',
 'flavanoids',
 'nonflavanoid_phenols',
 'proanthocyanins',
 'color_intensity',
 'hue',
 'od280/od315_of_diluted_wines',
 'proline']
```

In [7]:

```
wine['data'].shape
# увидим, что у нас в наличии 178 образцов данных(instances) и 13 атрибутов
```

Out[7]:

```
(178, 13)
```

In [6]:

```
# преобразование в PandasDataframe
my_data = pd.DataFrame(data=np.c_[wine['data'], wine['target']], columns=wine['feature_names']+['target'])
```

In [17]:

```
# верхние пять строк датасета
my_data.head()
```

Out[17]:

| | alcohol | malic_acid | ash | alcalinity_of_ash | magnesium | total_phenols | flavanoids | nonflavanoid_phenols | proanthocyanins | color_intensity | hue | od |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 14.23 | 1.71 | 2.43 | 15.6 | 127.0 | 2.80 | 3.06 | 0.28 | 2.29 | 5.64 | 1.04 | |
| **1** | 13.20 | 1.78 | 2.14 | 11.2 | 100.0 | 2.65 | 2.76 | 0.26 | 1.28 | 4.38 | 1.05 | |
| **2** | 13.16 | 2.36 | 2.67 | 18.6 | 101.0 | 2.80 | 3.24 | 0.30 | 2.81 | 5.68 | 1.03 | |
| **3** | 14.37 | 1.95 | 2.50 | 16.8 | 113.0 | 3.85 | 3.49 | 0.24 | 2.18 | 7.80 | 0.86 | |
| **4** | 13.24 | 2.59 | 2.87 | 21.0 | 118.0 | 2.80 | 2.69 | 0.39 | 1.82 | 4.32 | 1.04 | |

In [19]:

```
my_data.dtypes
# все колонки с типами данных
```

Out[19]:

```
alcohol                         float64
malic_acid                      float64
ash                             float64
alcalinity_of_ash               float64
magnesium                       float64
total_phenols                   float64
flavanoids                      float64
nonflavanoid_phenols            float64
proanthocyanins                 float64
color_intensity                 float64
hue                             float64
od280/od315_of_diluted_wines    float64
proline                         float64
target                          float64
dtype: object
```

In [8]:

```
# проверим наличие пустых значений
my_data.isnull().any()
```

Out[8]:

```
alcohol                         False
malic_acid                      False
ash                             False
alcalinity_of_ash               False
magnesium                       False
total_phenols                   False
flavanoids                      False
nonflavanoid_phenols            False
proanthocyanins                 False
color_intensity                 False
hue                             False
od280/od315_of_diluted_wines    False
proline                         False
target                          False
dtype: bool
```

In [23]:

```
# основные показатели
my_data.describe()
```

Out[23]:

| | alcohol | malic_acid | ash | alcalinity_of_ash | magnesium | total_phenols | flavanoids | nonflavanoid_phenols | proanthocyanins | color_inter |
|---|---|---|---|---|---|---|---|---|---|---|
| **count** | 178.000000 | 178.000000 | 178.000000 | 178.000000 | 178.000000 | 178.000000 | 178.000000 | 178.000000 | 178.000000 | 178.00 |
| **mean** | 13.000618 | 2.336348 | 2.366517 | 19.494944 | 99.741573 | 2.295112 | 2.029270 | 0.361854 | 1.590899 | 5.05 |
| **std** | 0.811827 | 1.117146 | 0.274344 | 3.339564 | 14.282484 | 0.625851 | 0.998859 | 0.124453 | 0.572359 | 2.31 |
| **min** | 11.030000 | 0.740000 | 1.360000 | 10.600000 | 70.000000 | 0.980000 | 0.340000 | 0.130000 | 0.410000 | 1.28 |
| **25%** | 12.362500 | 1.602500 | 2.210000 | 17.200000 | 88.000000 | 1.742500 | 1.205000 | 0.270000 | 1.250000 | 3.22 |
| **50%** | 13.050000 | 1.865000 | 2.360000 | 19.500000 | 98.000000 | 2.355000 | 2.135000 | 0.340000 | 1.555000 | 4.69 |
| **75%** | 13.677500 | 3.082500 | 2.557500 | 21.500000 | 107.000000 | 2.800000 | 2.875000 | 0.437500 | 1.950000 | 6.20 |
| **max** | 14.830000 | 5.800000 | 3.230000 | 30.000000 | 162.000000 | 3.880000 | 5.080000 | 0.660000 | 3.580000 | 13.00 |

In [26]:

```
my_data['target'].unique()
# определим уникальные значения для целевого признака
```

Out[26]:

```
array([0., 1., 2.])
```

# Визуальное исследование датасета

In [30]:

```
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
sns.set(style="ticks")
```

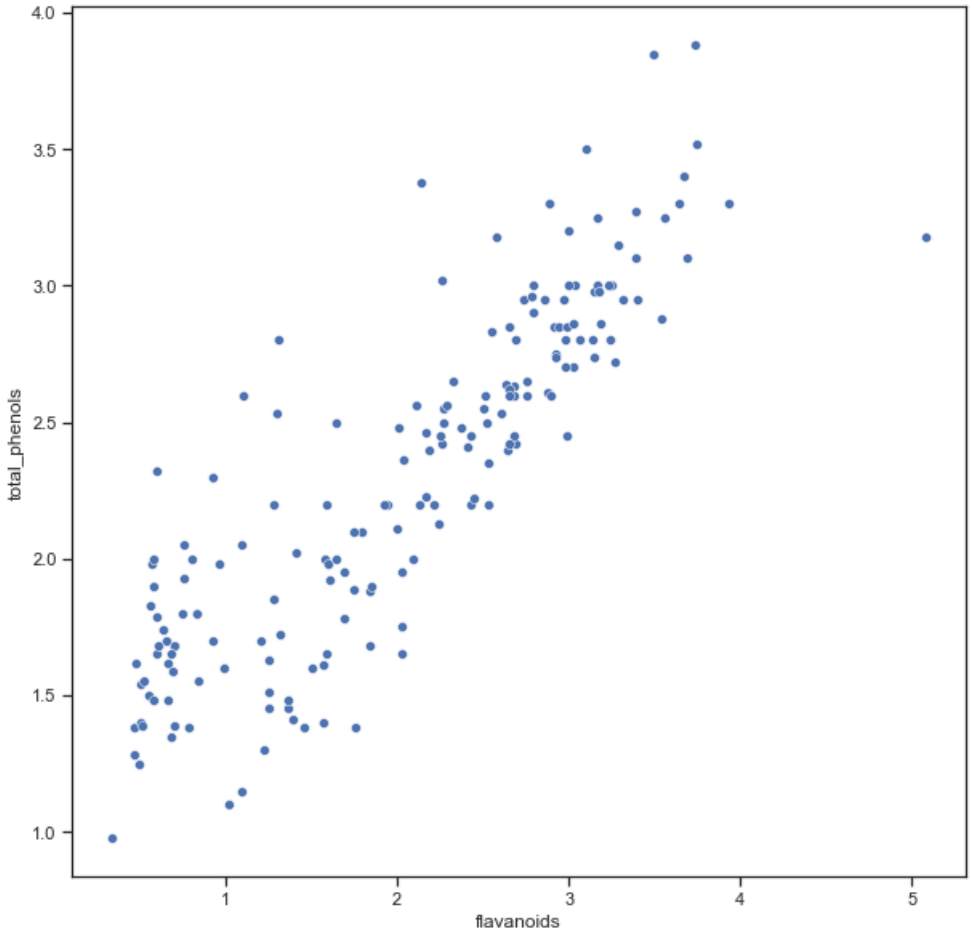
## Диаграмма рассеяния

In [47]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='flavanoids', y='total_phenols', data=my_data)
```

Out[47]:

```
<AxesSubplot:xlabel='flavanoids', ylabel='total_phenols'>
```




## Гистограмма

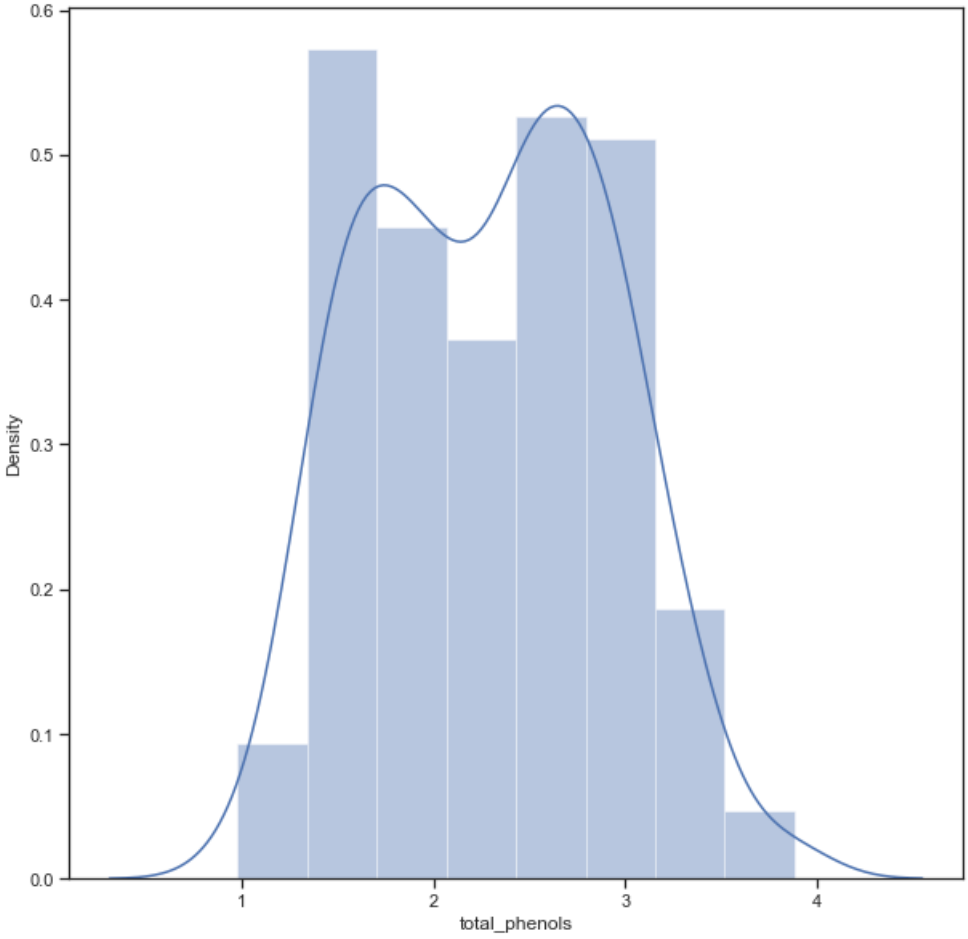
In [43]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(my_data['total_phenols'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a depre
cated function and will be removed in a future version. Please adapt your code to use either `displot` (a figu
re-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[43]:

```
<AxesSubplot:xlabel='total phenols', ylabel='Density'>
```




## KDE

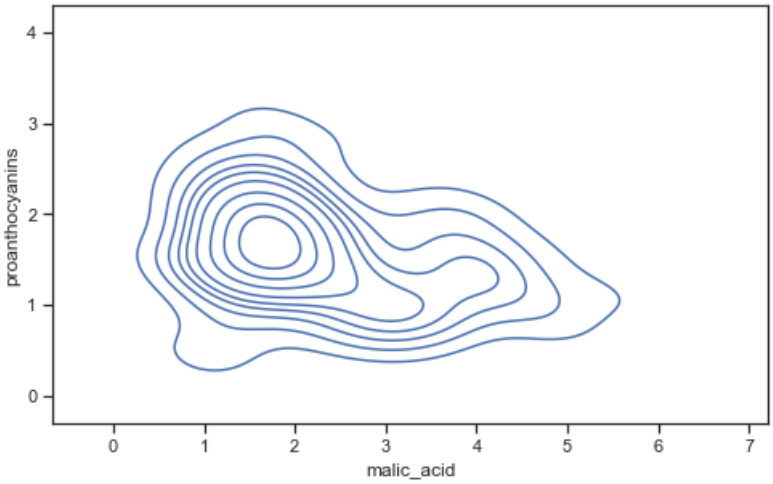
In [65]:

```
f, ax = plt.subplots(figsize=(8, 8))
ax.set_aspect("equal")

# Draw a contour plot to represent each bivariate density
sns.kdeplot(
    data=my_data,
    x="malic_acid",
    y="proanthocyanins",
    thresh=.1,
)
```

Out[65]:

```
<AxesSubplot:xlabel='malic_acid', ylabel='proanthocyanins'>
```



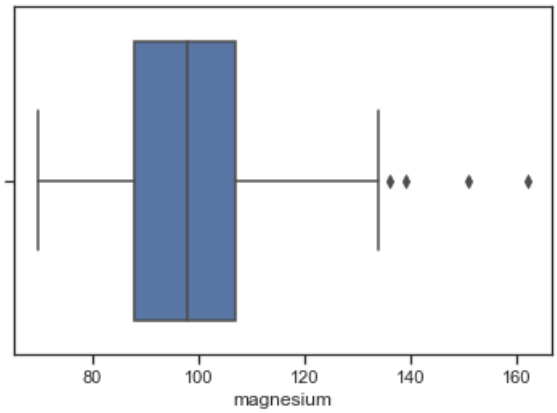

## Ящик с усами

In [53]:

```
sns.boxplot(x=my_data['magnesium'])
```

Out[53]:

```
<AxesSubplot:xlabel='magnesium'>
```




## Catplot

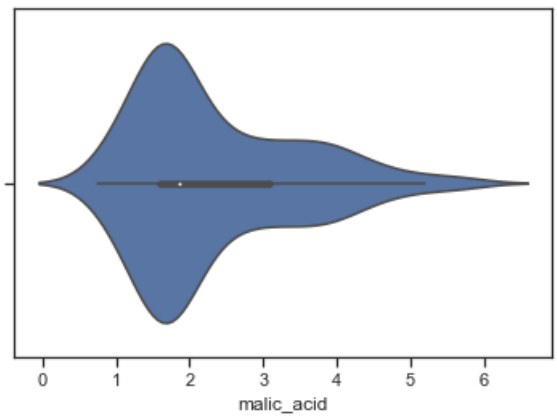
По краям отображаются распределения плотности.

In [54]:

```
sns.violinplot(x=my_data['malic_acid'])
```

Out[54]:

```
<AxesSubplot:xlabel='malic_acid'>
```




# Информация о корреляции признаков

In [44]:

```
# построим корреляционную матрицу
```

```
my_data.corr()
```

Out[44]:

| | alcohol | malic_acid | ash | alcalinity_of_ash | magnesium | total_phenols | flavanoids | nonflavanoid_phenols | proanth |
|---|---|---|---|---|---|---|---|---|---|
| **alcohol** | 1.000000 | 0.094397 | 0.211545 | -0.310235 | 0.270798 | 0.289101 | 0.236815 | -0.155929 | |
| **malic_acid** | 0.094397 | 1.000000 | 0.164045 | 0.288500 | -0.054575 | -0.335167 | -0.411007 | 0.292977 | - |
| **ash** | 0.211545 | 0.164045 | 1.000000 | 0.443367 | 0.286587 | 0.128980 | 0.115077 | 0.186230 | |
| **alcalinity_of_ash** | -0.310235 | 0.288500 | 0.443367 | 1.000000 | -0.083333 | -0.321113 | -0.351370 | 0.361922 | - |
| **magnesium** | 0.270798 | -0.054575 | 0.286587 | -0.083333 | 1.000000 | 0.214401 | 0.195784 | -0.256294 | |
| **total_phenols** | 0.289101 | -0.335167 | 0.128980 | -0.321113 | 0.214401 | 1.000000 | 0.864564 | -0.449935 | |
| **flavanoids** | 0.236815 | -0.411007 | 0.115077 | -0.351370 | 0.195784 | 0.864564 | 1.000000 | -0.537900 | |
| **nonflavanoid_phenols** | -0.155929 | 0.292977 | 0.186230 | 0.361922 | -0.256294 | -0.449935 | -0.537900 | 1.000000 | - |
| **proanthocyanins** | 0.136698 | -0.220746 | 0.009652 | -0.197327 | 0.236441 | 0.612413 | 0.652692 | -0.365845 | |
| **color_intensity** | 0.546364 | 0.248985 | 0.258887 | 0.018732 | 0.199950 | -0.055136 | -0.172379 | 0.139057 | - |
| **hue** | -0.071747 | -0.561296 | -0.074667 | -0.273955 | 0.055398 | 0.433681 | 0.543479 | -0.262640 | |
| **od280/od315_of_diluted_wines** | 0.072343 | -0.368710 | 0.003911 | -0.276769 | 0.066004 | 0.699949 | 0.787194 | -0.503270 | |
| **proline** | 0.643720 | -0.192011 | 0.223626 | -0.440597 | 0.393351 | 0.498115 | 0.494193 | -0.311385 | |
| **target** | -0.328222 | 0.437776 | -0.049643 | 0.517859 | -0.209179 | -0.719163 | -0.847498 | 0.489109 | - |

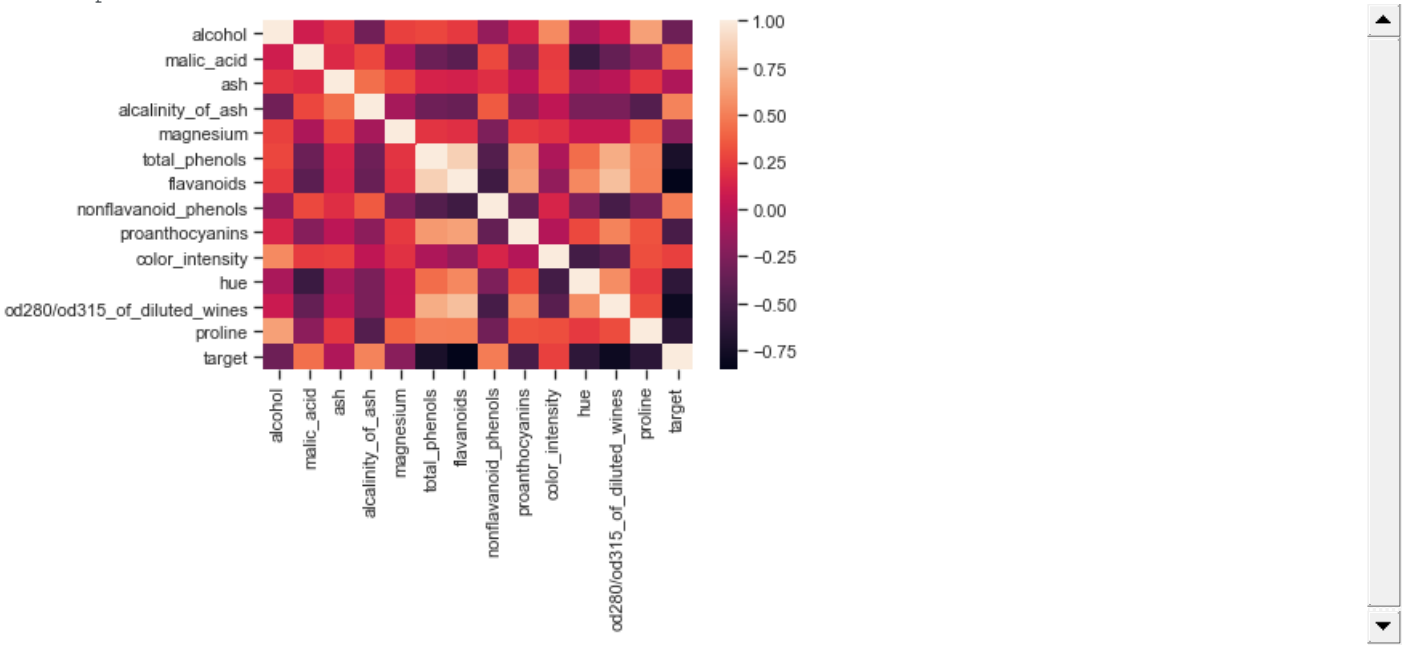
## Тепловая карта

In [58]:

```
sns.heatmap(my_data.corr(), annot=True)
```

Out[58]:

```
<AxesSubplot:>
```




In [ ]: