



Министерство образования и науки 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)

---

# МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ

*Отчёт по рубежному контролю № 1*

*«Методы обработки данных»*

Выполнила:  
студентка группы ИУ5 – 23М

Галичий Д. А.

Преподаватель:

Гапанюк Ю. Е.

2020г.

# Рубежный контроль №1

## Тема: Методы обработки данных

Номер варианта: 3

Номер задачи: 1

Номер набора данных, указанного в задаче: 3

### Задача №1

Для заданного набора данных постройте основные графики, входящие в этап разведочного анализа данных с использованием библиотек Matplotlib и Seaborn. В случае наличия пропусков в данных удалите строки или колонки, содержащие пропуски. Какие графики Вы построили и почему? Какие выводы о наборе данных Вы можете сделать на основании построенных графиков? Проведите корреляционный анализ. Сделайте выводы о возможности построения моделей машинного обучения и о возможном вкладе признаков в модель.

### Импорт библиотек

In [3]:

```
# Импорт библиотек
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
import sklearn
```

### Загрузка данных

In [4]:

```
from sklearn.datasets import load_wine
wines = load_wine()
```

### Характеристики датасета

In [5]:

```
# Полная информация о наборе данных  
wines
```

Out[5]:

[illegible]

3). UCI Machine Learning Repository\n[https://archive.ics.uci.edu/ml]. Irvine, CA: University of California,\nSchool of Information and Computer Science. \n\n.. topic:: References\n\n\n (1) S. Aeberhard, D. Coomans and O. de Vel, \n Comparison of Classifiers in High Dimensional Settings, \n Tech. Rep. no. 92-02, (1992), Dept. of Computer Science and Dept. of \n Mathematics and Statistics, James Cook University of North Queensland. \n (Also submitted to Technometrics). \n\n The data was used with many others for comparing various \n classifiers. The classes are separable, though only RDA \n has achieved 100% correct classification. \n (RDA : 100%, QDA 99.4%, LDA 98.9%, 1NN 96.1% (z-transformed data)) \n (All results using the leave-one-out technique) \n\n (2) S. Aeberhard, D. Coomans and O. de Vel, \n "THE CLASSIFICATION PERFORMANCE OF RDA" \n Tech. Rep. no. 92-01, (1992), Dept. of Computer Science and Dept. of \n Mathematics and Statistics, James Cook University of North Queensland. \n (Also submitted to Journal of Chemometrics).\n',  
'feature\_names': ['alcohol',  
'malic\_acid',  
'ash',  
'alcalinity\_of\_ash',  
'magnesium',  
'total\_phenols',  
'flavanoids',  
'nonflavanoid\_phenols',  
'proanthocyanins',  
'color\_intensity',  
'hue',  
'od280/od315\_of\_diluted\_wines',  
'proline']}]

In [6]:

```
X, y, t_names, f_names = wines["data"], wines["target"], wines["target_names"], wines["feature_names"]
```

In [7]:

```
X.shape
```

Out[7]:

```
(178, 13)
```

In [8]:

```
y.shape
```

Out[8]:

```
(178,)
```

Существует 178 записей, с каждой записью связано 13 признаков.

In [9]:

```
t_names
```

Out[9]:

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

Целевой признак - принадлежность к одному из трёх классов вин.

In [10]:

```
# Названия признаков  
f_names
```

Out[10]:

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

In [11]:

```
#Первые 5 записей  
X[:5]
```

Out[11]:

```
array([[1.423e+01, 1.710e+00, 2.430e+00, 1.560e+01, 1.270e+02, 2.800e+00,  
        3.060e+00, 2.800e-01, 2.290e+00, 5.640e+00, 1.040e+00, 3.920e+00,  
        1.065e+03],  
 [1.320e+01, 1.780e+00, 2.140e+00, 1.120e+01, 1.000e+02, 2.650e+00,  
        2.760e+00, 2.600e-01, 1.280e+00, 4.380e+00, 1.050e+00, 3.400e+00,  
        1.050e+03],  
 [1.316e+01, 2.360e+00, 2.670e+00, 1.860e+01, 1.010e+02, 2.800e+00,  
        3.240e+00, 3.000e-01, 2.810e+00, 5.680e+00, 1.030e+00, 3.170e+00,  
        1.185e+03],  
 [1.437e+01, 1.950e+00, 2.500e+00, 1.680e+01, 1.130e+02, 3.850e+00,  
        3.490e+00, 2.400e-01, 2.180e+00, 7.800e+00, 8.600e-01, 3.450e+00,  
        1.480e+03],  
 [1.324e+01, 2.590e+00, 2.870e+00, 2.100e+01, 1.180e+02, 2.800e+00,  
        2.690e+00, 3.900e-01, 1.820e+00, 4.320e+00, 1.040e+00, 2.930e+00,  
        7.350e+02]])
```

In [12]:

```
# Первые 5 значений целевых признаков  
y[:5]
```

Out[12]:

```
array([0, 0, 0, 0, 0])
```

In [13]:

```
# Преобразование набора данных в dataframe
w_data = pd.DataFrame(data = np.c_[X, y],
                      columns = f_names + ['target'])
```

In [14]:

```
w_data
```

Out[14]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonf
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	
...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	

178 rows × 14 columns

In [43]:

```
# Уникальные значения целевого признака
w_data['target'].unique()
```

Out[43]:

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

In [53]:

```
# Приведение целевого признака к типу int
new = w_data['target'].astype('int')
new
```

Out[53]:

```
0      0
1      0
2      0
3      0
4      0
...
173    2
174    2
175    2
176    2
177    2
```

Name: target, Length: 178, dtype: int32

In [55]:

```
# Объединение датасета с новым целочисленным целевым признаком
d_wines = w_data.join(new, how='left', lsuffix='_left', rsuffix='_right')
```

In [56]:

```
d_wines
```

Out[56]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonf
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	
...	...	...	...	...	...	...	...	
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	

178 rows × 15 columns

In [57]:

```
# Переименование столбца с целочисленным целевым признаком
d_wines.rename(columns={'target_right': 'target'}, inplace=True)
d_wines
```

Out[57]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonf
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	
...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	

178 rows × 15 columns

In [58]:

```
# Удаление старого столбца с целевым признаком
del d_wines['target_left']
```

In [59]:

```
d_wines
```

Out[59]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonf
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	
...	...	...	...	...	...	...	...	
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	

178 rows × 14 columns

In [60]:

```
w_data = d_wines
```

## Визуальное исследование датасета

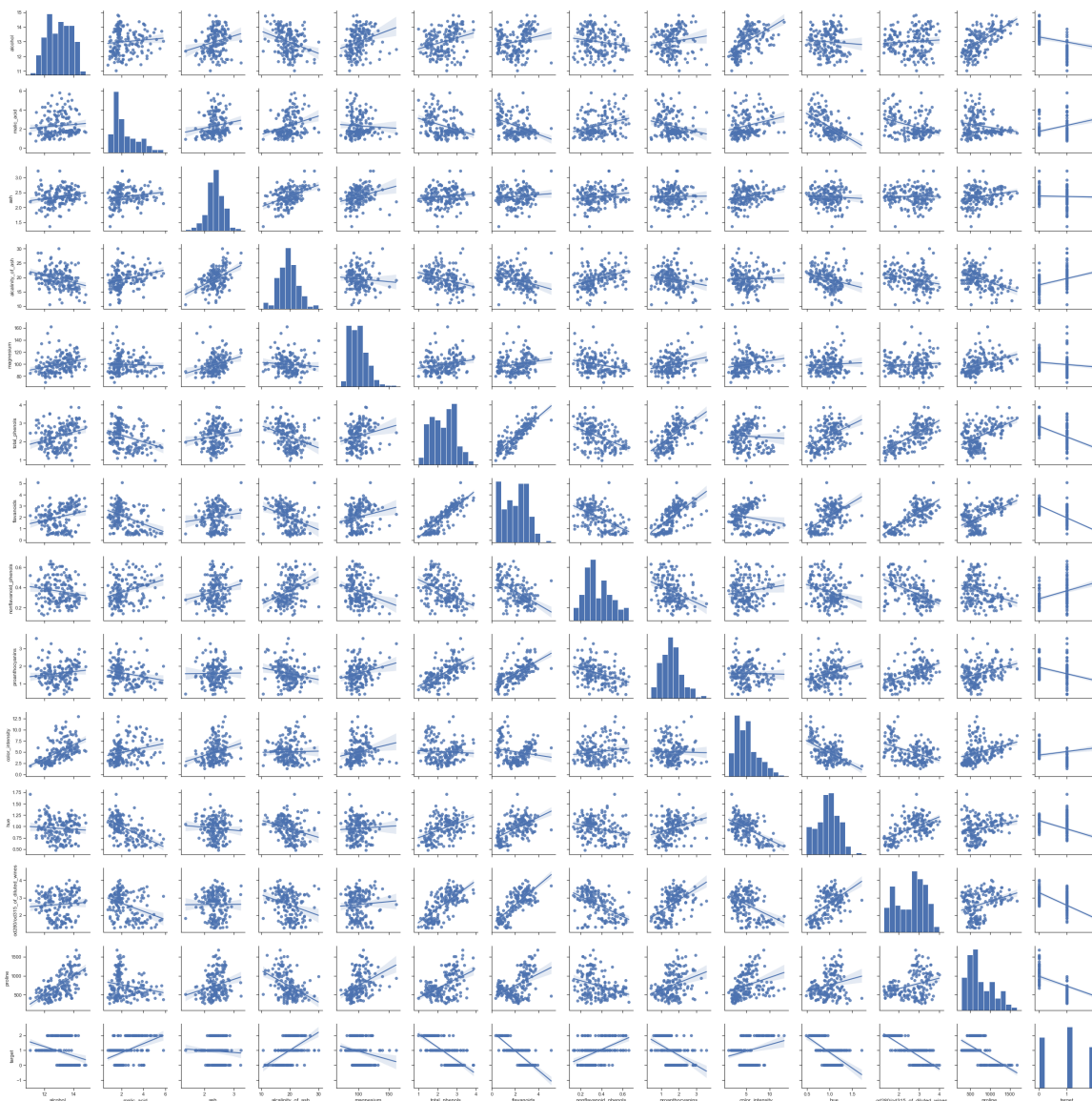
Комбинация гистограмм и диаграмм рассеивания для всего набора данных

In [31]:

```
sns.pairplot(w_data, kind="reg")
```

Out[31]:

<seaborn.axisgrid.PairGrid at 0x26b01cd5b70>



По построенным графикам можно увидеть зависимость, близкую к линейной, между параметрами 'total\_phenols' и 'flavanoids'. Построим диаграмму рассеивания для этих параметров с учётом целевого признака.

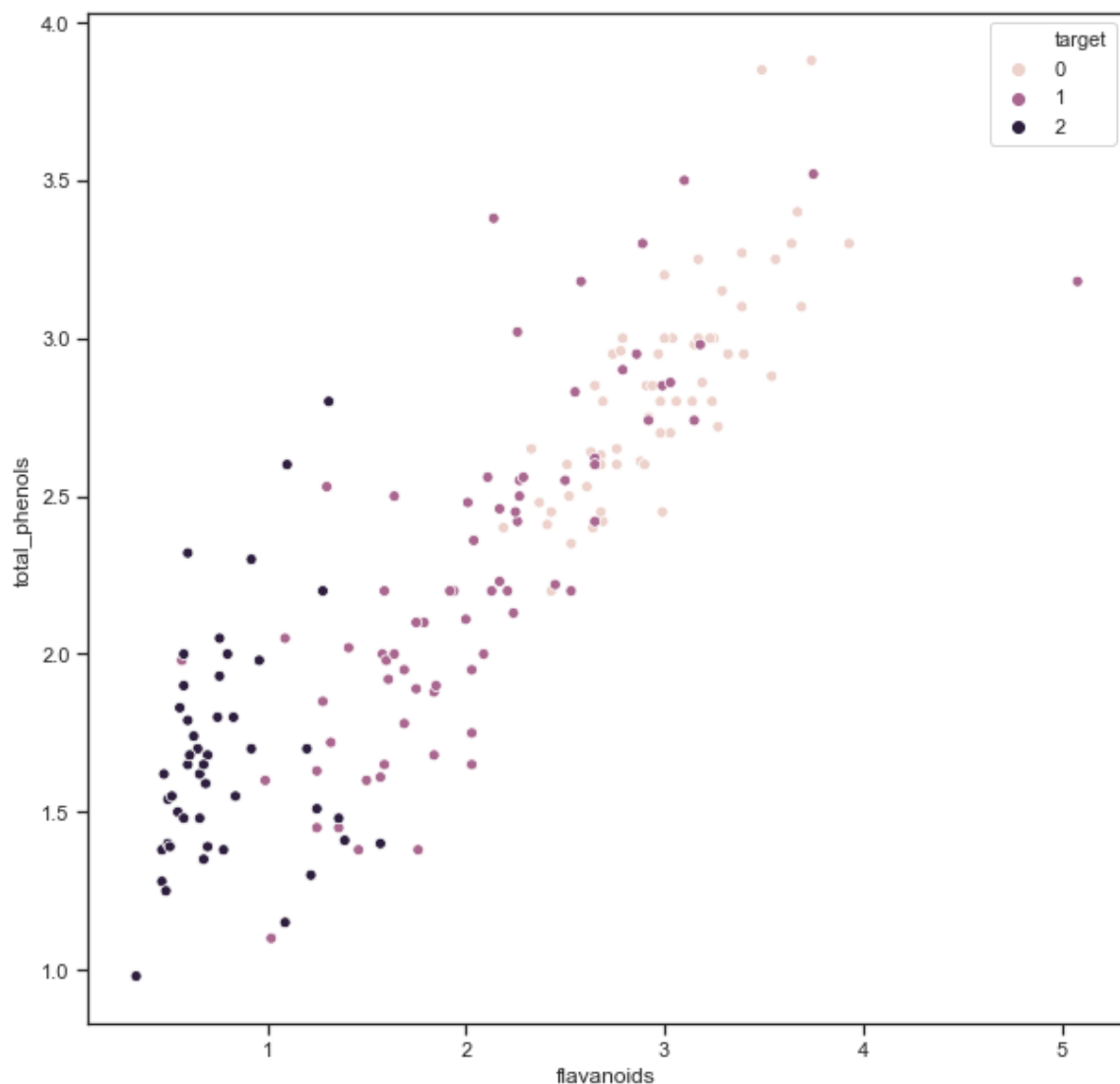
Зависимость между другими признаками модели сложна и неочевидна, поэтому выявить её визуально с помощью вышеуказанных графиков не представляется возможным.

In [62]:

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

Out[62]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x26b0d0ee710>



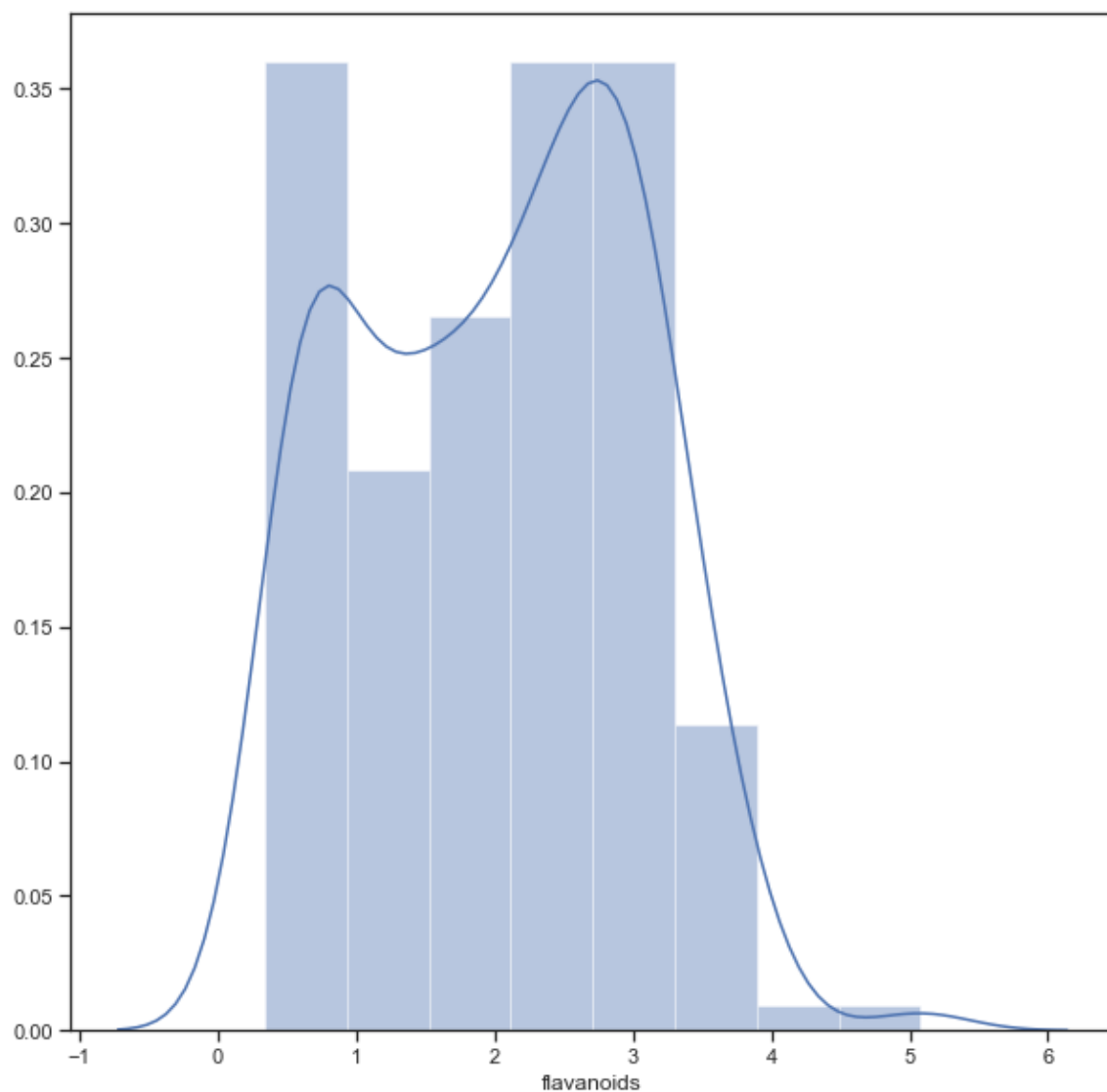
Построим гистограмму для оценки плотности вероятности распределения данных.

In [65]:

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

Out[65]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x26b0d5019b0>



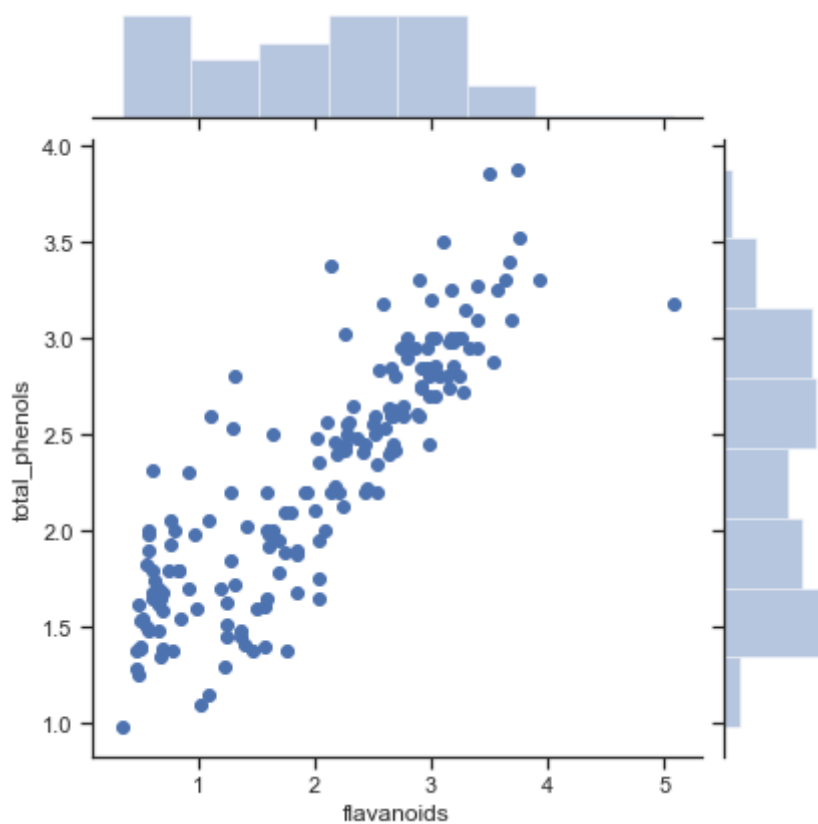
Построим комбинацию гистограммы и диаграммы рассеивания для тех же признаков.

In [66]:

```
sns.jointplot(x='flavanoids', y='total_phenols', data=w_data)
```

Out[66]:

<seaborn.axisgrid.JointGrid at 0x26b0d501b00>



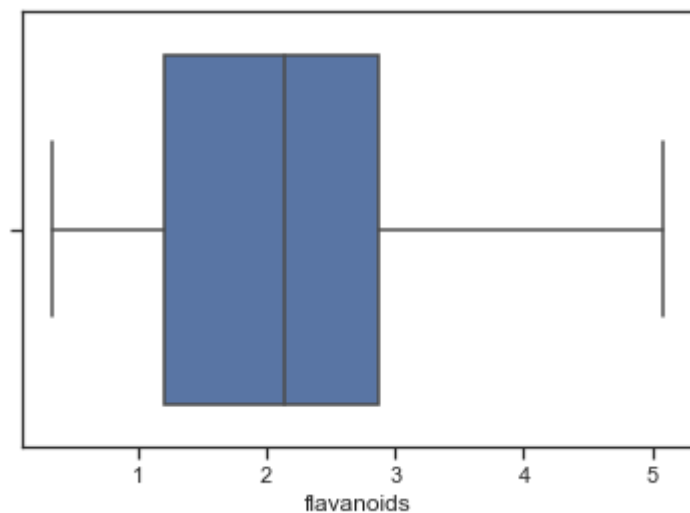
Для колонки данных 'flavanoids' построим график 'Ящик с усами', чтобы отобразить одномерное распределение вероятности.

In [68]:

```
sns.boxplot(x=w_data['flavanoids'])
```

Out[68]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x26b0da80358>



## Сбор информации о корреляции признаков

Построим корреляционную матрицу

In [69]:

```
w_data.corr()
```

Out[69]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium
alcohol	1.000000	0.094397	0.211545	-0.310235	0.270798
malic_acid	0.094397	1.000000	0.164045	0.288500	-0.054575
ash	0.211545	0.164045	1.000000	0.443367	0.286587
alcalinity_of_ash	-0.310235	0.288500	0.443367	1.000000	-0.083333
magnesium	0.270798	-0.054575	0.286587	-0.083333	1.000000
total_phenols	0.289101	-0.335167	0.128980	-0.321113	0.214401
flavanoids	0.236815	-0.411007	0.115077	-0.351370	0.195784
nonflavanoid_phenols	-0.155929	0.292977	0.186230	0.361922	-0.256294
proanthocyanins	0.136698	-0.220746	0.009652	-0.197327	0.236441
color_intensity	0.546364	0.248985	0.258887	0.018732	0.199950
hue	-0.071747	-0.561296	-0.074667	-0.273955	0.055398
od280/od315_of_diluted_wines	0.072343	-0.368710	0.003911	-0.276769	0.066004
proline	0.643720	-0.192011	0.223626	-0.440597	0.393351
target	-0.328222	0.437776	-0.049643	0.517859	-0.209179

По матрице корреляций делаем следующие выводы:

- Целевой признак наиболее сильно коррелирует с параметрами 'flavanoids' (0.847) 'od280/od315\_of\_diluted\_wines' (0.788) и 'total\_phenols' (0.719). Эти признаки обязательно следует оставить в модели;
- Целевой признак также коррелирует с параметрами 'proline' (0.633), 'hue' (0.617) и 'alcalinity\_of\_ash' (0.517). Эти признаки также будут полезны для модели;
- Целевой признак слабо коррелирует с признаками 'ash' (0.049), 'magnesium' (0.209) и 'color\_intensity' (0.265). Данные признаки можно исключить при построении модели, так как они не несут информативность;
- Признаки 'total\_phenols' и 'flavanoids' сильно коррелируют между собой (0.864). Возможно при необходимости исключить из модели один из них;
- Также можно сделать вывод, что, выбирая из признаков 'total\_phenols' и 'flavanoids', лучше выбрать 'flavanoids', потому что он сильнее коррелирован с целевым признаком.

Для более наглядного представления матрицы корреляции можно воспользоваться тепловой картой.

In [71]:

```
sns.heatmap(w_data.corr(), annot=True, fmt='.1f')
```

Out[71]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x26b0edc69e8>

