Разведочный анализ данных. Исследование и визуализация данных

1) Текстовое описание набора данных

В качестве набора данных мы будем использовать набор данных по распознаванию вин

Эти данные представляют собой результаты химического анализа вин, выращенных в одном и том же регионе Италии тремя разными культиваторами. Было проведено тринадцать различных измерений, проведенных для различных компонентов, содержащихся в трех типах вина.

- Алкоголь
- Яблочная кислота
- Пепел
- Щелочность золы
- Магний
- Общие фенолы
- Флаваноиды
- Нефлаваноидные фенолы
- Проантоцианы
- Интенсивность цвета
- Оттенок
- **OD280 / OD315** разбавленных вин
- Пролайн

Загрузка данных

Загрузим файлы датасета в помощью библиотеки **Pandas.**

In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
from sklearn.datasets import *
wine = load_wine()
```

In [2]:

```
data = pd.DataFrame(data= np.c_[wine['data'], wine['target']],
                    columns= list(wine['feature_names']) + ['target'])
```

2) Основные характеристики датасета

In [4]:

```
# Первые 5 строк датасета
data.head()
```

Out[4]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	c
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	c
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	

In [5]:

```
# Размер датасета (строки, колонки)
data.shape
```

Out[5]:

```
(178, 14)
```

In [6]:

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

```
Всего строк: 178
```

In [7]:

```
# Список колонок
data.columns
```

Out[7]:

```
Index(['alcohol', 'malic_acid', 'ash', 'alcalinity_of_ash', 'magnesium',
       'total_phenols', 'flavanoids', 'nonflavanoid_phenols',
       'proanthocyanins', 'color_intensity', 'hue',
       'od280/od315_of_diluted_wines', 'proline', 'target'],
      dtype='object')
```

In [8]:

```
# Список колонок с типами данных
data.dtypes
```

Out[8]:

```
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 [9]:

```
# Проверим наличие пустых значений
# Цикл по колонкам датасета
for col in data.columns:
    # Количество пустых значений - все значения заполнены
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
alcohol - 0
malic_acid - 0
ash - 0
```

```
alcalinity_of_ash - 0
magnesium - 0
total_phenols - 0
flavanoids - 0
nonflavanoid_phenols - 0
proanthocyanins - 0
color_intensity - 0
hue - 0
od280/od315_of_diluted_wines - 0
proline - 0
target - 0
```

In [10]:

```
# Датасет
data
```

Out[10]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82
...	...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	0.52	1.06
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	0.43	1.41
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	0.43	1.35
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	0.53	1.46
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	0.56	1.35

178 rows × 14 columns

In [11]:

```
# Основные статистические характеристки набора данных
data.describe()
```

Out[11]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	pro
count	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	
mean	13.000618	2.336348	2.366517	19.494944	99.741573	2.295112	2.029270	0.361854	
std	0.811827	1.117146	0.274344	3.339564	14.282484	0.625851	0.998859	0.124453	
min	11.030000	0.740000	1.360000	10.600000	70.000000	0.980000	0.340000	0.130000	
25%	12.362500	1.602500	2.210000	17.200000	88.000000	1.742500	1.205000	0.270000	
50%	13.050000	1.865000	2.360000	19.500000	98.000000	2.355000	2.135000	0.340000	
75%	13.677500	3.082500	2.557500	21.500000	107.000000	2.800000	2.875000	0.437500	
max	14.830000	5.800000	3.230000	30.000000	162.000000	3.880000	5.080000	0.660000	

In [12]:

```
# Определим уникальные значения для пола
data['target'].unique()
```

Out[12]:

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

3) Визуальное исследование датасета

Диаграмма рассеяния

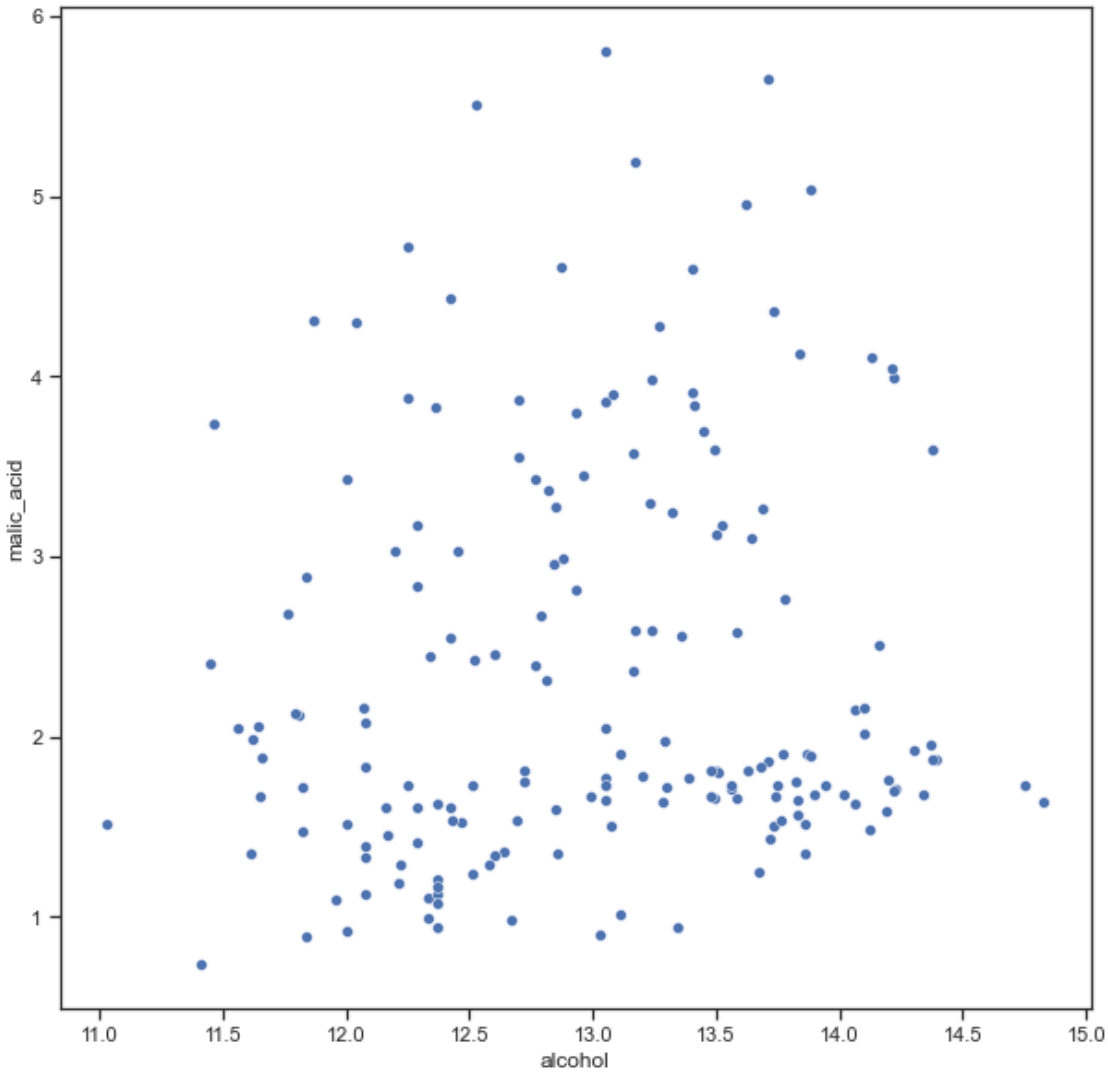
Позволяет построить распределение двух колонок данных и визуально обнаружить наличие зависимости.

In [13]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='alcohol', y='malic_acid', data=data)
```

Out[13]:

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

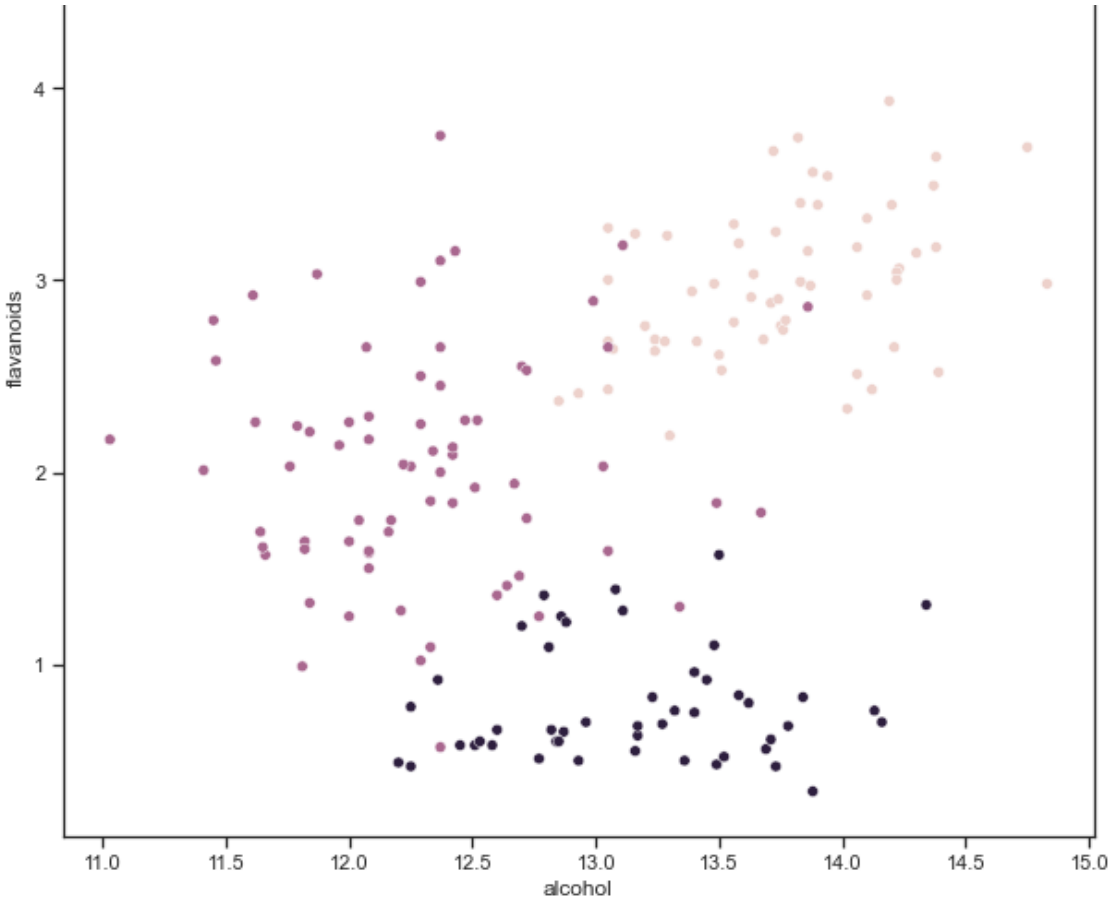


In [14]:

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

Out[14]:

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



Гистограмма

Позволяет оценить плотность вероятности распределения данных.

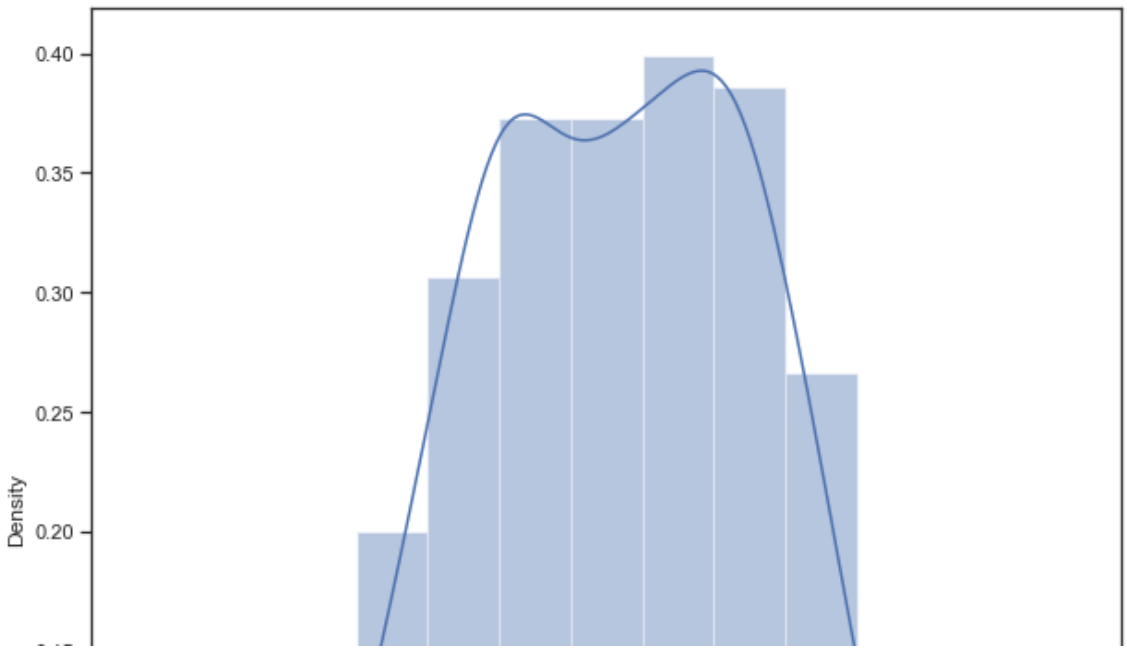
In [15]:

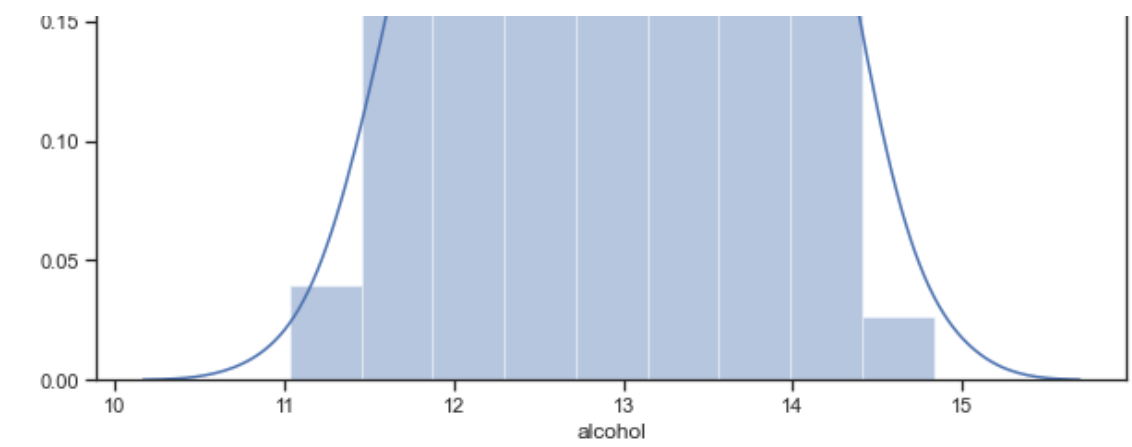
```
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(data['alcohol'])
```

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

Out[15]:

```
<AxesSubplot:xlabel='alcohol', ylabel='Density'>
```





Jointplot

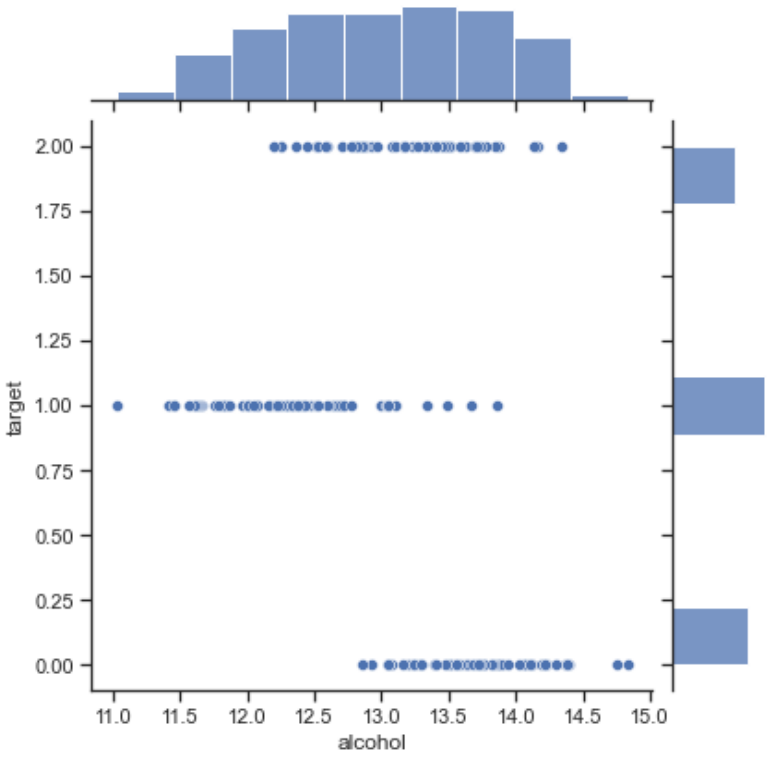
Комбинация гистограмм и диаграмм рассеивания.

In [16]:

```
sns.jointplot(x='alcohol', y='target', data=data)
```

Out[16]:

```
<seaborn.axisgrid.JointGrid at 0x2216bb1a5e0>
```

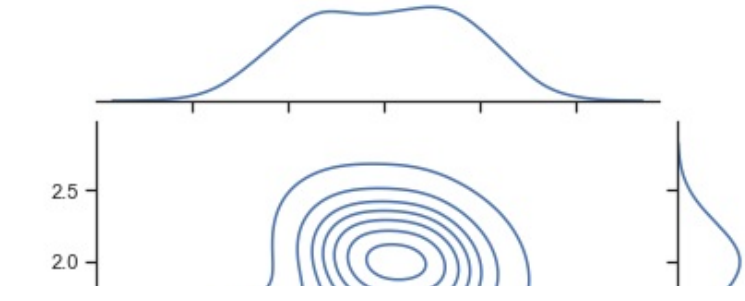


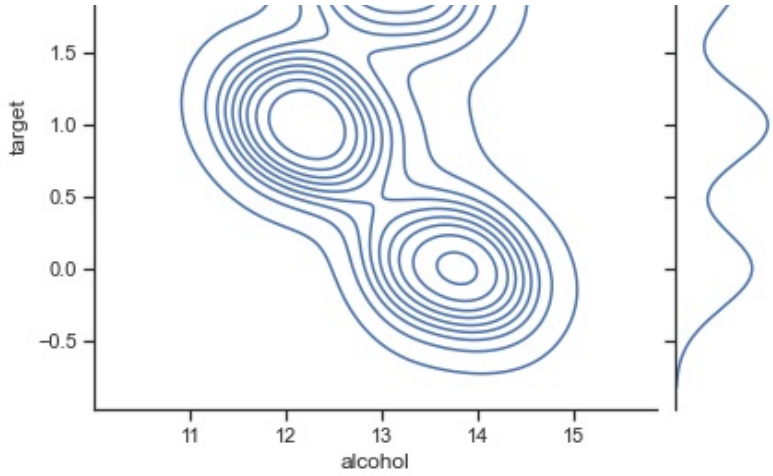
In [17]:

```
sns.jointplot(x='alcohol', y='target', data=data, kind="kde")
```

Out[17]:

```
<seaborn.axisgrid.JointGrid at 0x2216bce3fa0>
```





"Парные диаграммы"

Комбинация гистограмм и диаграмм рассеивания для всего набора данных.

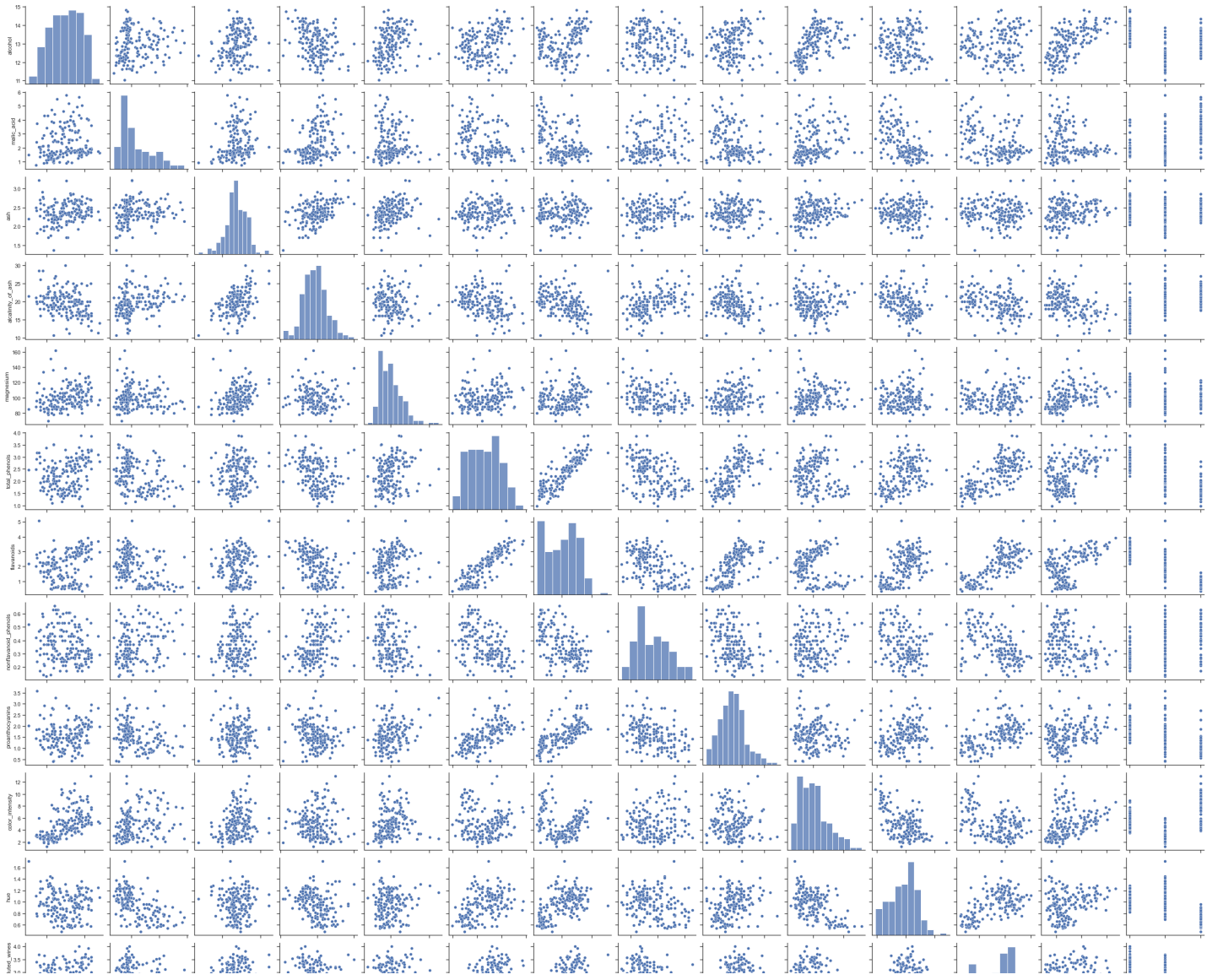
Выводится матрица графиков. На пересечении строки и столбца, которые соответстуют двум показателям, строится диаграмма рассеивания. В главной диагонали матрицы строятся гистограммы распределения соответствующих показателей.

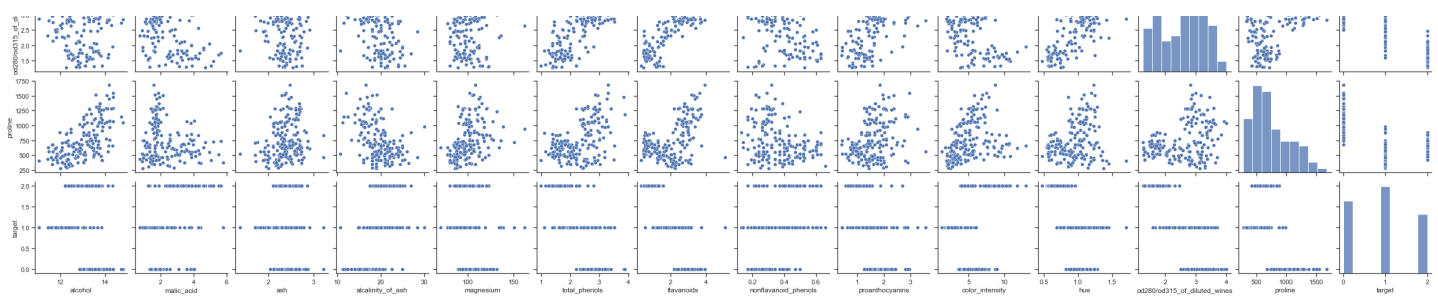
In [18]:

```
sns.pairplot(data)
```

Out[18]:

```
<seaborn.axisgrid.PairGrid at 0x2216be37400>
```





In [23]:

```
sns.pairplot(data, hue='target')
```

Out[23]:

```
<seaborn.axisgrid.PairGrid at 0x22173ebc6d0>
```

Violin plot

Отображает одномерное распределение вероятности, по краям отображаются распределения плотности.

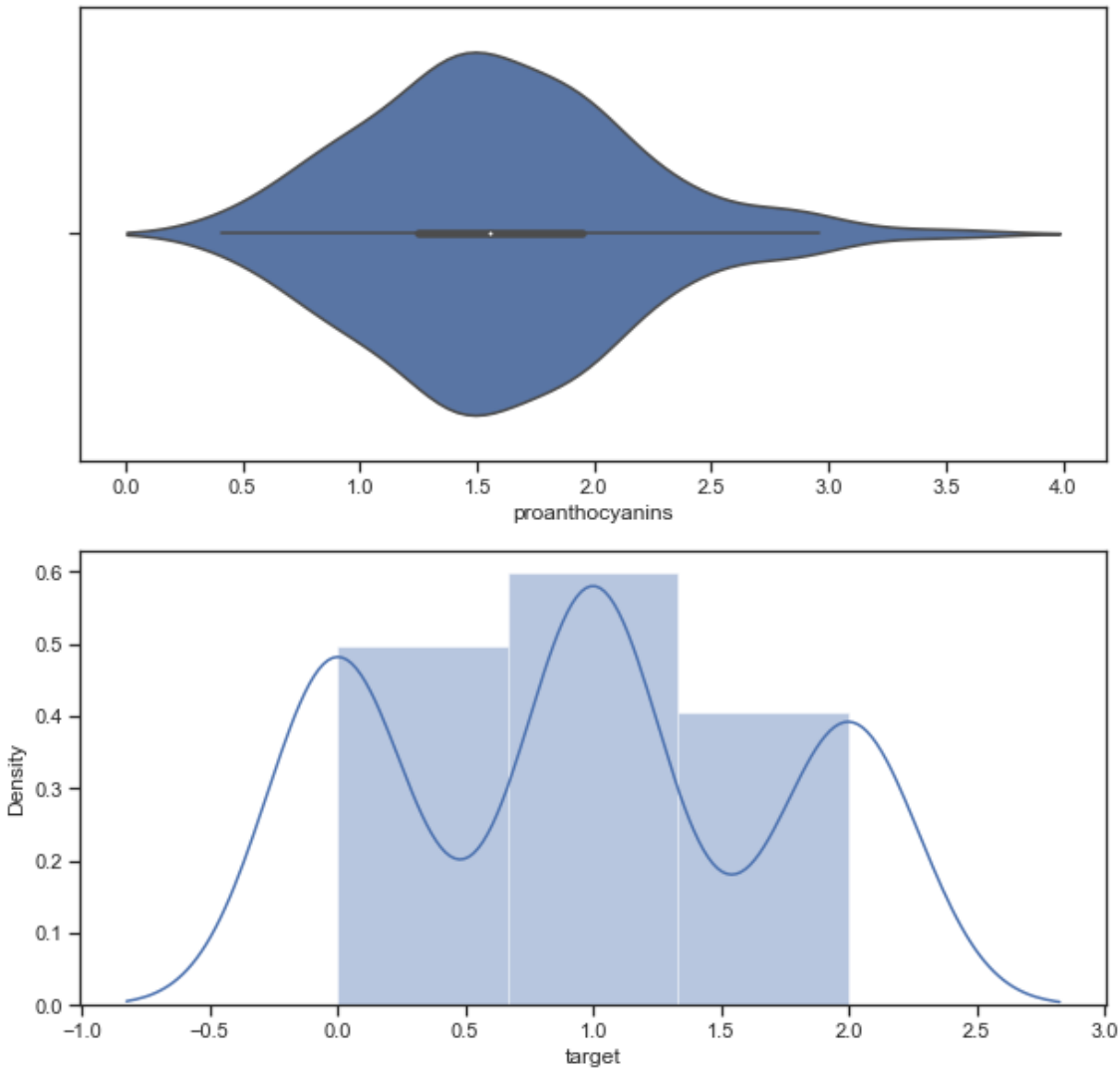
In [19]:

```
fig, ax = plt.subplots(2, 1, figsize=(10,10))
sns.violinplot(ax=ax[0], x=data['proanthocyanins'])
sns.distplot(data['target'], ax=ax[1])
```

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

Out[19]:

```
<AxesSubplot:xlabel='target', ylabel='Density'>
```



4) Информация о корреляции признаков

Построим матрицу корреляции с помощью разных методов

In [20]:

```
data.corr()
```

Out[20]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflav
alcohol	1.000000	0.094397	0.211545	-0.310235	0.270798	0.289101	0.236815	
malic_acid	0.094397	1.000000	0.164045	0.288500	-0.054575	-0.335167	-0.411007	
ash	0.211545	0.164045	1.000000	0.443367	0.286587	0.128980	0.115077	

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

In [21]:

```
data.corr(method='kendall')
```

Out[21]:

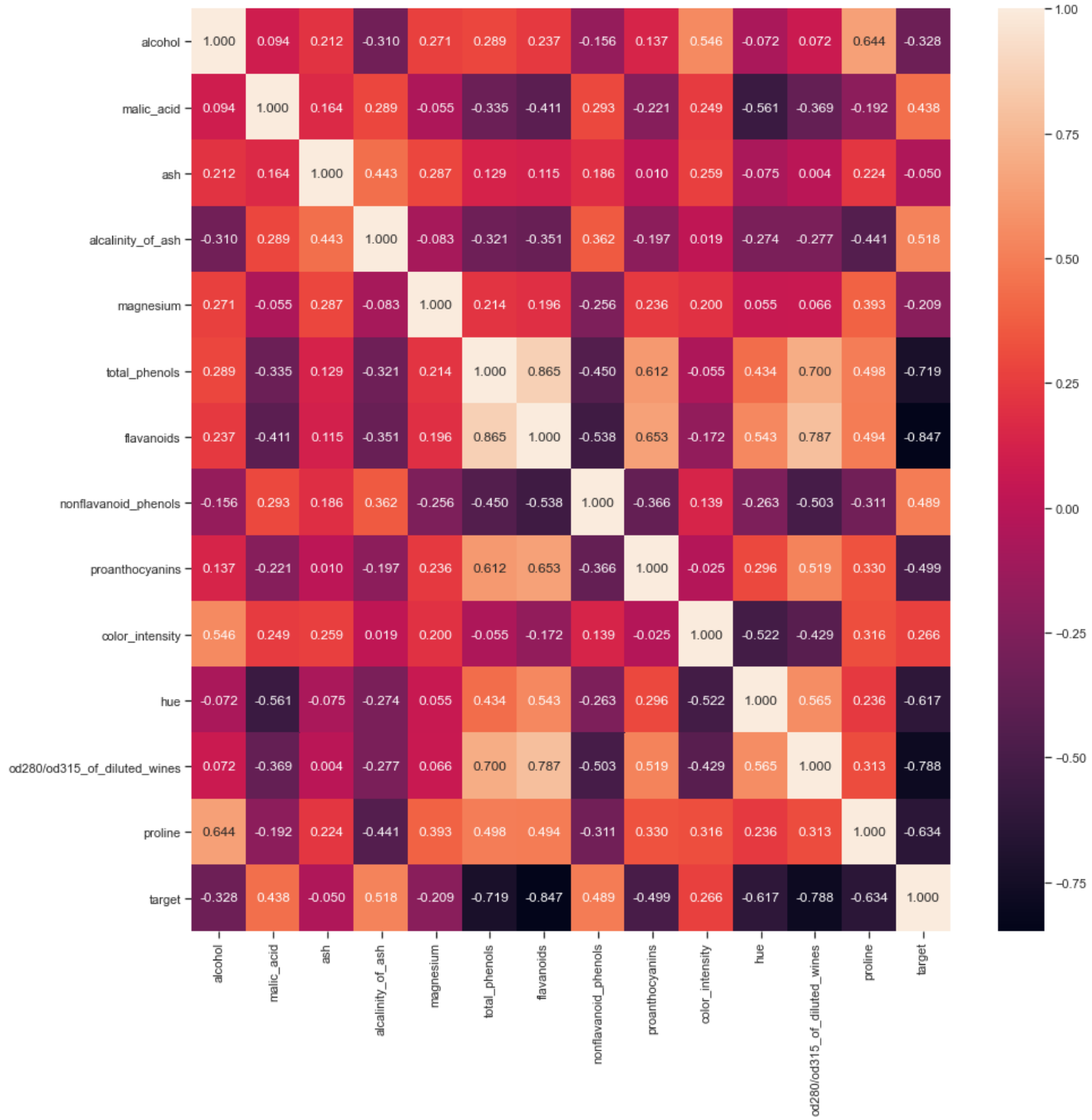
	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflav
alcohol	1.000000	0.093844	0.170154	-0.212978	0.250506	0.209099	0.191087	
malic_acid	0.093844	1.000000	0.158178	0.210119	0.050869	-0.174929	-0.211918	
ash	0.170154	0.158178	1.000000	0.258352	0.254246	0.089855	0.049474	
alcalinity_of_ash	-0.212978	0.210119	0.258352	1.000000	-0.121005	-0.256669	-0.309865	
magnesium	0.250506	0.050869	0.254246	-0.121005	1.000000	0.172195	0.161603	
total_phenols	0.209099	-0.174929	0.089855	-0.256669	0.172195	1.000000	0.701999	
flavanoids	0.191087	-0.211918	0.049474	-0.309865	0.161603	0.701999	1.000000	
nonflavanoid_phenols	-0.109554	0.175129	0.098937	0.278091	-0.158361	-0.310443	-0.378099	
proanthocyanins	0.133526	-0.168714	0.018240	-0.171404	0.117871	0.466517	0.534615	
color_intensity	0.434353	0.195607	0.187786	-0.057281	0.241781	0.028264	0.028674	
hue	-0.021717	-0.388707	-0.037234	-0.239210	0.023760	0.289210	0.354372	
od280/od315_of_diluted_wines	0.061513	-0.162909	-0.006341	-0.226253	0.034307	0.478267	0.520448	
proline	0.449387	-0.044660	0.171574	-0.313218	0.343016	0.280203	0.263661	
target	-0.238984	0.247494	-0.038085	0.449402	-0.184992	-0.590404	-0.725255	

In [24]:

```
fig, ax = plt.subplots(figsize=(15,15))
sns.heatmap(data.corr(), annot=True, fmt='.3f')
```

Out[24]:

<AxesSubplot:>



Выводы о коррелирующих признаках

На основе нашей корреляционной матрице, визуализированной с помощью тепловой карты, определим признаки которые коррелируют с нашим целевым признаком.

Использованы следующие обозначения:

- **Alcohol -** Крепость
- **Malic_acid -** Яблочная кислота

- **Ash -** неорганические вещества
- **Alcalinity_of_ash -** Щелочность неорганических веществ
- **Magnesium -** Магний
- **Total_phenols -** Содержание полифенолов
- **Flavanoids -** Содержание Флавоноиды
- **Nonflavanoid_phenols -** Нефлаваноидные фенолы
- **Proanthocyanins -** Проантоцианидины
- **Color_intensity -** Интенсивность цвета
- **Hue -** Оттенок
- **od280/od315_of_diluted_wines - OD280 / OD315** разбавленных вин **(**метод определения концентрации белка**.)**
- **proline -** Пролин

Отрицательный коэффициент корреляции показывает**,** что две переменные могут быть связаны таким образом**,** что при возрастании значений одной из них значения другой убывают**.**

- Наиболее коррелируемым признаком является содержание флавоноиды **- (-0.847)**
- Вторым по коэффициенту корреляции является концентрации белка **- (-0.788)**
- Так как между собой содержание флавоноиды и концентрация белка также коррелируют **(0.787),** мы можем оставить в модели только один из двух признаков
- Исключим из модели слабокоррерилирующие признаки такие как интенсивность цвета **(0.266),** неорганические вещества **(-0.050),** содержание магния **(-0.209)**
- Хорошо коррелируемым признаком является содержание пфенола**(-0.719),** но мы не можем добавить его в модель так как этот признак корелирует с признаком содержания флавоноиды
- Целевой признак хорошо коррелирует с признаками пролин **(-0.634),** оттенок **(-0.617),** щелочность неорганических веществ **(0.518),** Проантоцианидины **(-0.499)** оставим их в модели
- Из модели исключаем такие признаки корреляции как крепость **(-0.328),** яблочная кислота **(-0.438),** Нефлаваноидные фенолы **(0.489)** так как они коррелируют с признаками пролин**,** оттенок**,** флавониды соотвественно