**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика с системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Технологии машинного обучения»

Отчёт по рубежному контролю №1

Выполнил: Проверил:

студент группы РТ5-61Б преподаватель каф. ИУ5

Агеев Алексей Гапанюк Ю.Е.

Подпись и дата: Подпись и дата:

Москва, 2023 г

## Задание

Вариант №1 (Задание №1)

Для заданного набора данных проведите корреляционный анализ. В случае наличия пропусков в данных удалите строки или колонки, содержащие пропуски. Сделайте выводы о возможности построения моделей машинного обучения и о возможном вкладе признаков в модель.

Для пары произвольных колонок данных построить график "Jointplot".

Набор данных: <https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_iris>

## Текстовое описание выбранного набора данных

Набор данных содержит следующие параметры: sepal length/width – длина/ширина чашелистика, petal length/width – длина/ширина лепестка.

## Основные характеристики датасета

import numpy as np  
import pandas as pd  
import seaborn as sns  
import matplotlib.pyplot as plt  
%matplotlib inline  
sns.set(style="ticks")  
from sklearn.datasets import load\_iris

# Набор данных для распознавания ирисов

iris = load\_iris()  
for x in iris:  
 print(x)

data  
target  
frame  
target\_names  
DESCR  
feature\_names  
filename  
data\_module

iris['target\_names']

array(['setosa', 'versicolor', 'virginica'], dtype='<U10')

iris['feature\_names']

['sepal length (cm)',  
 'sepal width (cm)',  
 'petal length (cm)',  
 'petal width (cm)']

iris['target'].shape

(150,)

data = pd.DataFrame(data= np.c\_[iris['data'], iris['target']],  
 columns= iris['feature\_names'] + ['target'])

data.head()

sepal length (cm) sepal width (cm) petal length (cm) petal width (cm)   
0 5.1 3.5 1.4 0.2 \  
1 4.9 3.0 1.4 0.2   
2 4.7 3.2 1.3 0.2   
3 4.6 3.1 1.5 0.2   
4 5.0 3.6 1.4 0.2   
  
 target   
0 0.0   
1 0.0   
2 0.0   
3 0.0   
4 0.0

# Размер датасета - 150 строк, 5 колонок  
data.shape

(150, 5)

total\_count = data.shape[0]  
print('Всего строк: {}'.format(total\_count))

Всего строк: 150

# Список колонок  
data.columns

Index(['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)',  
 'petal width (cm)', 'target'],  
 dtype='object')

# Список колонок с типами данных  
data.dtypes

sepal length (cm) float64  
sepal width (cm) float64  
petal length (cm) float64  
petal width (cm) float64  
target float64  
dtype: object

# 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))

sepal length (cm) - 0  
sepal width (cm) - 0  
petal length (cm) - 0  
petal width (cm) - 0  
target - 0

# Основные статистические характеристки набора данных  
data.describe()

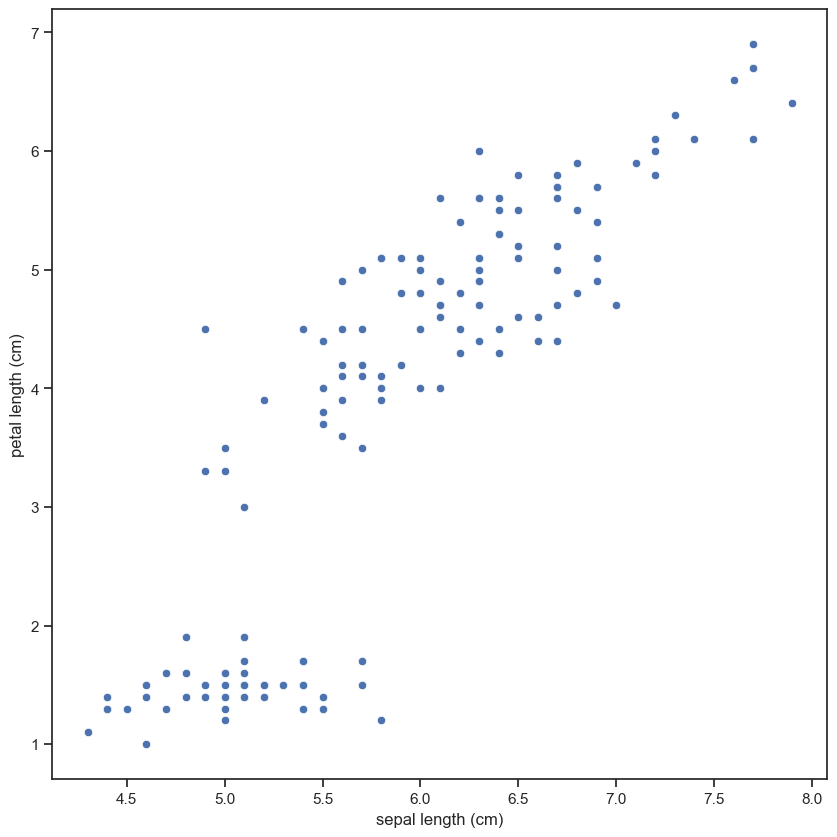
sepal length (cm) sepal width (cm) petal length (cm)   
count 150.000000 150.000000 150.000000 \  
mean 5.843333 3.057333 3.758000   
std 0.828066 0.435866 1.765298   
min 4.300000 2.000000 1.000000   
25% 5.100000 2.800000 1.600000   
50% 5.800000 3.000000 4.350000   
75% 6.400000 3.300000 5.100000   
max 7.900000 4.400000 6.900000   
  
 petal width (cm) target   
count 150.000000 150.000000   
mean 1.199333 1.000000   
std 0.762238 0.819232   
min 0.100000 0.000000   
25% 0.300000 0.000000   
50% 1.300000 1.000000   
75% 1.800000 2.000000   
max 2.500000 2.000000

# Определим уникальные значения для целевого признака  
data['target'].unique()

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

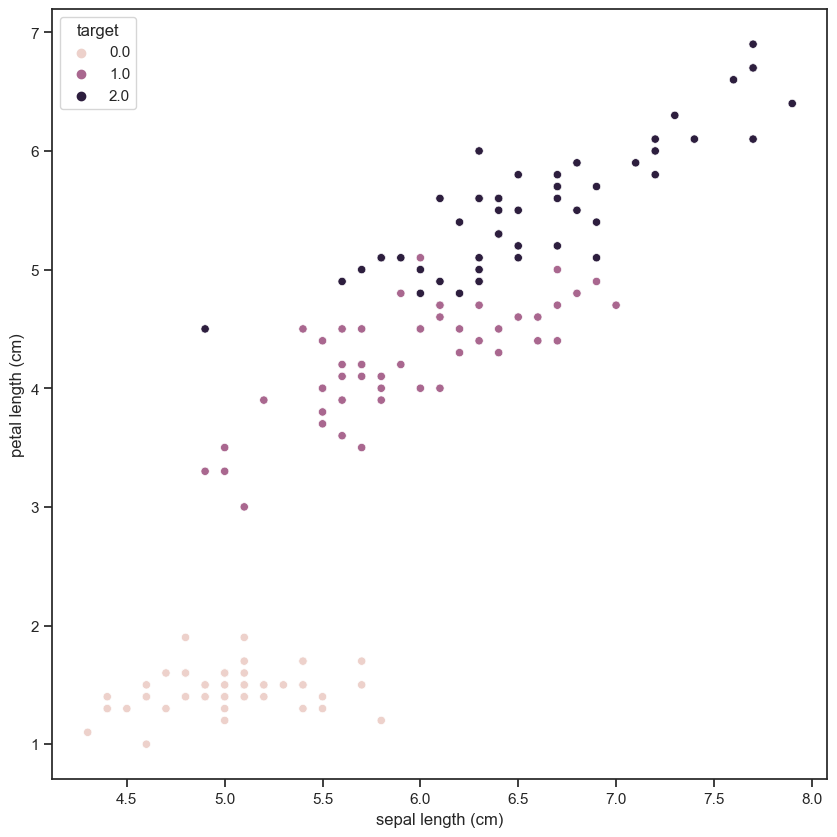
fig, ax = plt.subplots(figsize=(10,10))   
sns.scatterplot(ax=ax, x='sepal length (cm)', y='petal length (cm)', data=data)

<Axes: xlabel='sepal length (cm)', ylabel='petal length (cm)'>



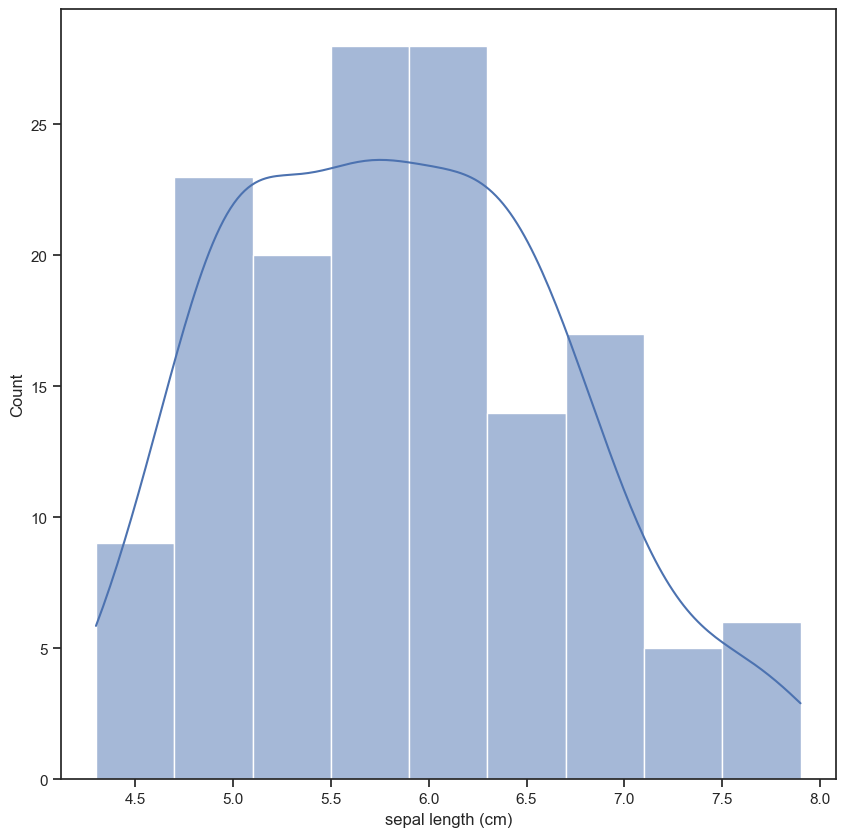
fig, ax = plt.subplots(figsize=(10,10))   
sns.scatterplot(ax=ax, x='sepal length (cm)', y='petal length (cm)', data=data, hue='target')

<Axes: xlabel='sepal length (cm)', ylabel='petal length (cm)'>



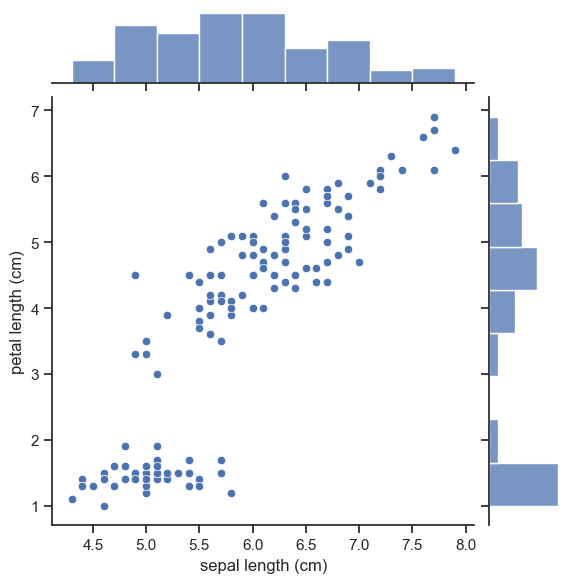
fig, ax = plt.subplots(figsize=(10,10))   
sns.histplot(data['sepal length (cm)'], kde=True)

<Axes: xlabel='sepal length (cm)', ylabel='Count'>



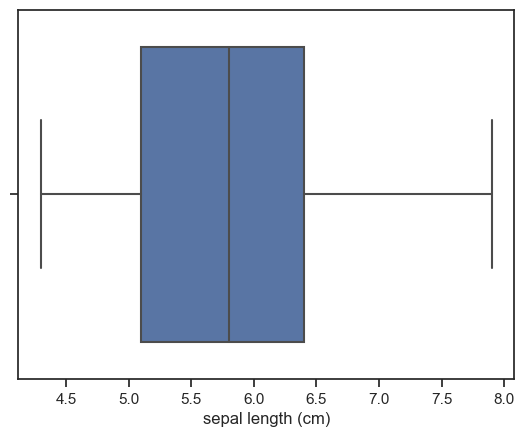
sns.jointplot(data=data, x='sepal length (cm)', y='petal length (cm)')

<seaborn.axisgrid.JointGrid at 0x241562ed190>



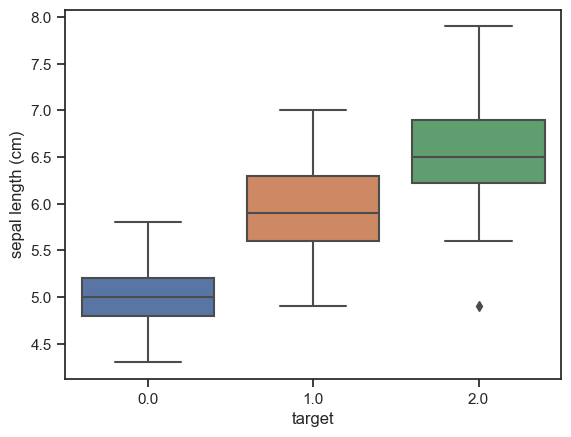
# По горизонтали  
sns.boxplot(x=data['sepal length (cm)'])

<Axes: xlabel='sepal length (cm)'>



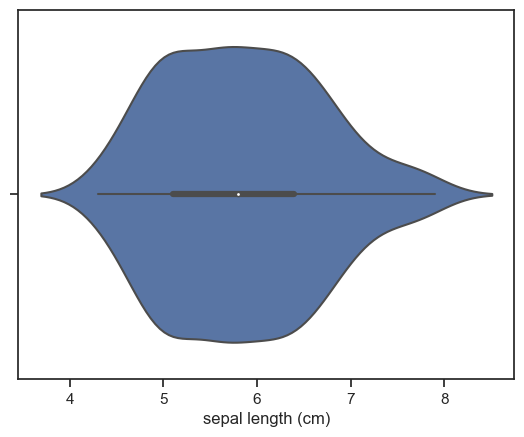
sns.boxplot(x='target', y='sepal length (cm)', data=data)

<Axes: xlabel='target', ylabel='sepal length (cm)'>



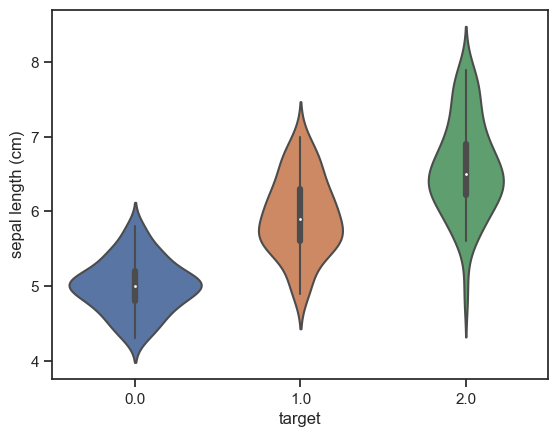
sns.violinplot(x=data['sepal length (cm)'])

<Axes: xlabel='sepal length (cm)'>



# Распределение параметра sepal length (cm) сгруппированные по target.  
sns.violinplot(x='target', y='sepal length (cm)', data=data)

<Axes: xlabel='target', ylabel='sepal length (cm)'>



data.corr()

sepal length (cm) sepal width (cm) petal length (cm)   
sepal length (cm) 1.000000 -0.117570 0.871754 \  
sepal width (cm) -0.117570 1.000000 -0.428440   
petal length (cm) 0.871754 -0.428440 1.000000   
petal width (cm) 0.817941 -0.366126 0.962865   
target 0.782561 -0.426658 0.949035   
  
 petal width (cm) target   
sepal length (cm) 0.817941 0.782561   
sepal width (cm) -0.366126 -0.426658   
petal length (cm) 0.962865 0.949035   
petal width (cm) 1.000000 0.956547   
target 0.956547 1.000000

На основе корреляционной матрицы можно сделать следующие выводы:

* Целевой признак наиболее сильно коррелирует с параметрами sepal length cm (0.78), petal length cm (0.949), petal width cm (0.956). Эти признаки обязательно следует оставить в модели.
* Целевой признак отчасти отрицательно коррелирует с sepal width (cm) (-0.427). Этот признак также стоит оставить в модели.