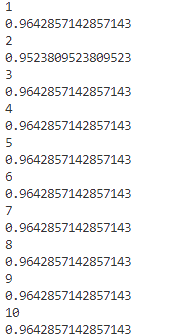
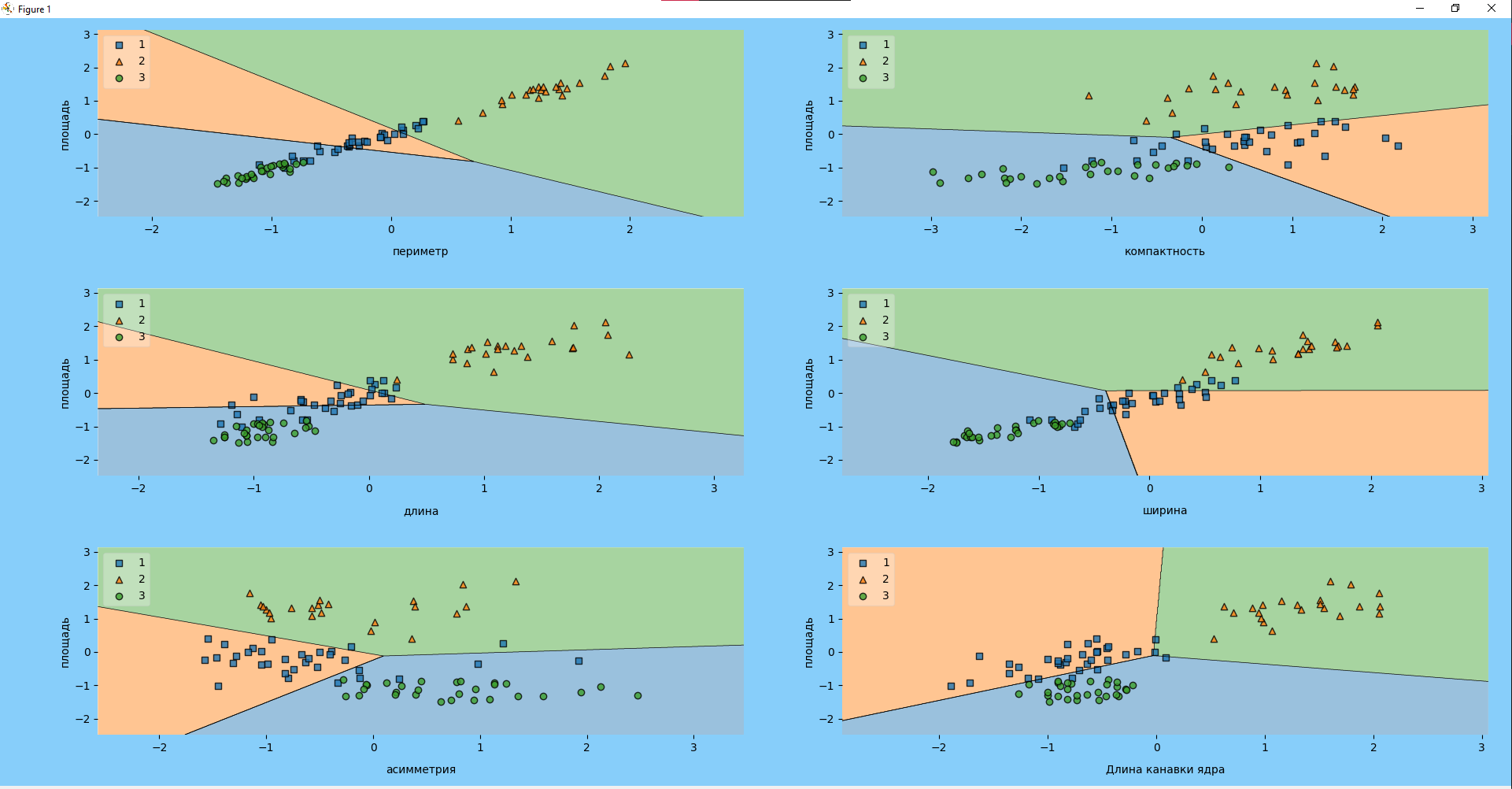
Используя метод логистической регрессии, находим оптимальную регуляризацию. Получим следующие результаты. На рисунке видно, что худшим параметром оказался 2.

****

Графики по парам признаков:

****

**Код программы**

import numpy as np

from sklearn.linear\_model import LogisticRegression

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

import matplotlib.pyplot as plt

from matplotlib.colors import ListedColormap

from mlxtend.plotting import plot\_decision\_regions

# from sklearn import datasets

from sklearn.svm import SVC

import pandas as pd

#логистич регресия

url = "C:/Users/fox/Desktop/ii/data/seeds\_dataset.data"

names = [

    "площадь",

    "периметр",

    "компактность",

    "длина",

    "ширина",

    "асимметрия",

    "Длина канавки ядра",

    "Сорт"

] #название атрибутов

dataset = pd.read\_csv(url, names=names) #чтение файла с данными

dataset.head()

X = dataset.iloc[:, :-1] #выбор вс

Y = dataset.iloc[:,7]

#print(dataset.shape)

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split( X, Y, test\_size=0.40,  random\_state = 1)

scaler  = StandardScaler()

scaler.fit(X\_train)

X\_train = scaler.transform(X\_train)

X\_test = scaler.transform(X\_test)

for i in range(1,11):

    print(i)

    model = LogisticRegression(C=i,random\_state=800, max\_iter=3000).fit(X\_train,Y\_train)

    model.predict(X\_test)

    #print(model.predict\_proba(X\_test))

    print(model.score(X\_test, Y\_test))

Y\_test=Y\_test.to\_numpy()

fig = plt.figure(figsize=(10,10), facecolor='#87CEFA')

for j in range(0,6):

    d = fig.add\_subplot(3,2,j+1)#сетка

    logistreg = LogisticRegression(C=1, solver='liblinear', random\_state=800)

    logistreg = logistreg.fit(X\_train[:,[j+1,0]], Y\_train)

    plot\_decision\_regions(X\_test[:,[j+1,0]], Y\_test, clf=logistreg, legend=2)

    plt.xlabel(dataset.columns[j+1],labelpad=10)

    plt.ylabel(dataset.columns[0])

    plt.tight\_layout(h\_pad=3)

plt.show()