计算机与信息工程学院实验报告（五）

姓名：王赫 学号：1828070097 专业：数据科学与大数据技术(明德计划) 年级：2018级

课程：机器学习与数据挖掘 主讲教师：罗慧敏 辅导教师：\_\_\_\_\_\_\_

实验时间：2020年11月 25日 下午17时至18时，实验地点：606

实验题目： 用线性核与高斯核训练支持向量机

实验目的： 掌握支持向量机的原理及应用

实验环境（硬件和软件） Anaconda/Jupyter notebook/Pycharm

实验内容：

使用Sklearn，在西瓜集3.0上分别使用线性核和高斯核训练一个SVM，并比较其支持向量的差别。

实验步骤：

1. 数学公式推导
2. 编程实现
3. 数据记录
4. 实验总结

实验数据记录：

**一、**已经给定部分代码，补充完整的代码，需要补充代码的地方已经用红色字体标注，包括：

**（1）#补充构建SVM模型及训练代码**

**（2）#补充预测代码**

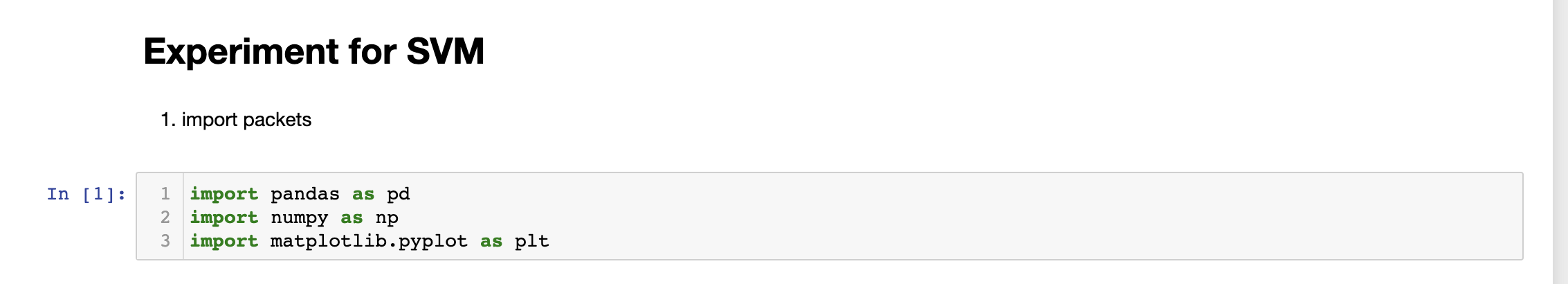
**二、**将补充完整的代码提交，并提交实验结果；（**也可以自己重写这部分的代码提交**）

data\_file\_watermelon\_3a = "watermelon\_3a.csv"

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt



df = pd.read\_csv(data\_file\_watermelon\_3a, header=None, )

df.columns = ['id', 'density', 'sugar\_content', 'label']

df.set\_index(['id'])

X = df[['density', 'sugar\_content']].values

y = df['label'].values



########## SVM training and comparison

# based on linear kernel as well as gaussian kernel

from sklearn import svm

for fig\_num, kernel in enumerate(('linear', 'rbf')):

**#补充构建SVM模型及训练代码**

# SVM model Creating and Training

svc = svm.SVC(C=500.0, kernel=kernel)

svc.fit(X, y)

#给定新的样本X\_test，预测其标签

X\_test = [[0.719,0.103]]

**#补充预测代码**

# Predict with this SVM model

pre = svc.predict(X\_test)

# get support vectors

sv = svc.support\_vectors\_

##### draw decision zone

plt.figure(fig\_num)

plt.clf()

# plot point and mark out support vectors

plt.scatter( X[:,0], X[:,1], edgecolors='k', c=y, cmap=plt.cm.Paired, zorder=10)

plt.scatter(sv[:,0], sv[:,1], edgecolors='k', facecolors='none', s=80, linewidths=2, zorder=10)

# plot the decision boundary and decision zone into a color plot

x\_min, x\_max = X[:, 0].min() - 0.2, X[:, 0].max() + 0.2

y\_min, y\_max = X[:, 1].min() - 0.2, X[:, 1].max() + 0.2

XX, YY = np.meshgrid(np.arange(x\_min, x\_max, 0.02), np.arange(y\_min, y\_max, 0.02))

Z = svc.decision\_function(np.c\_[XX.ravel(), YY.ravel()])

Z = Z.reshape(XX.shape)

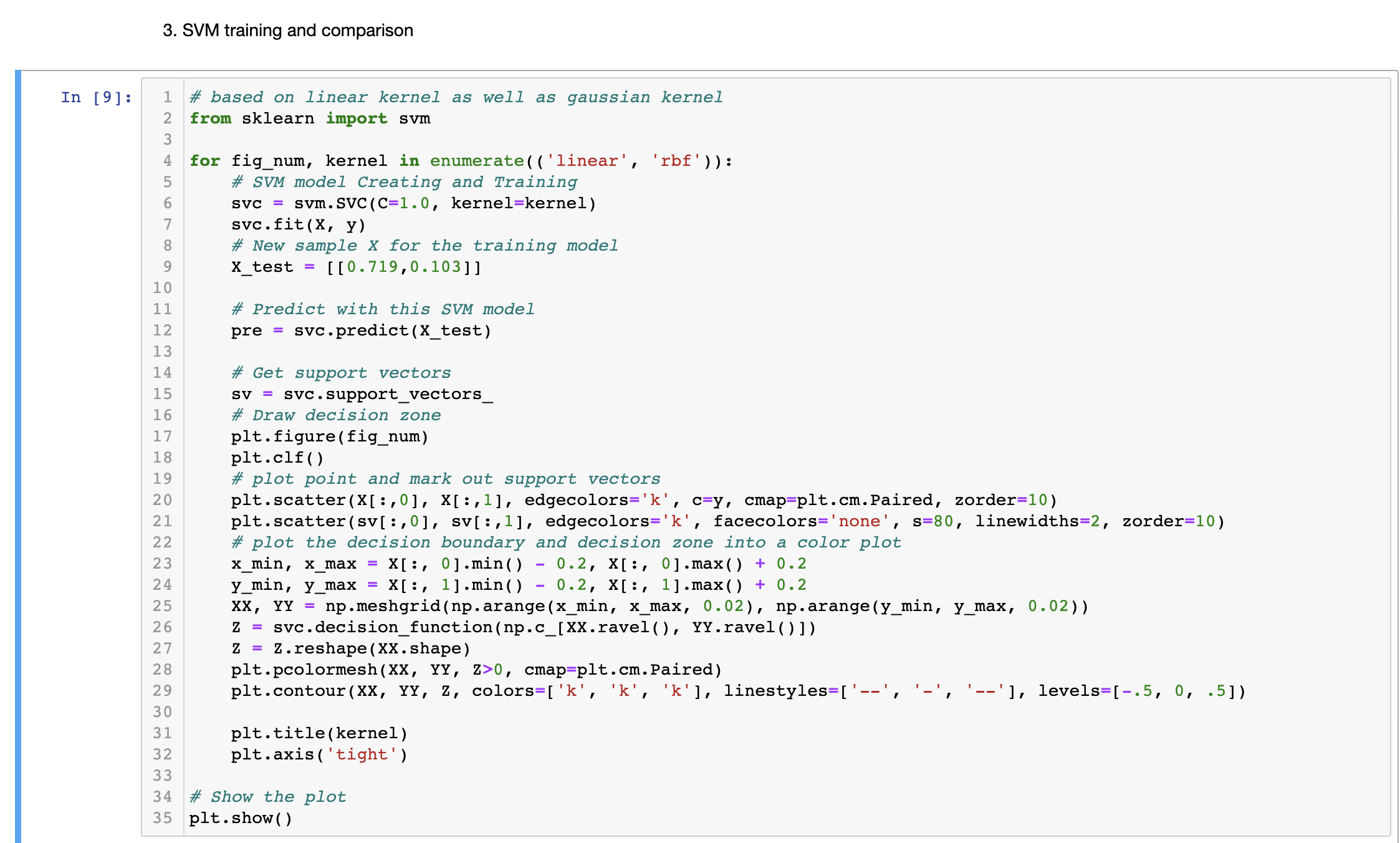
plt.pcolormesh(XX, YY, Z>0, cmap=plt.cm.Paired)

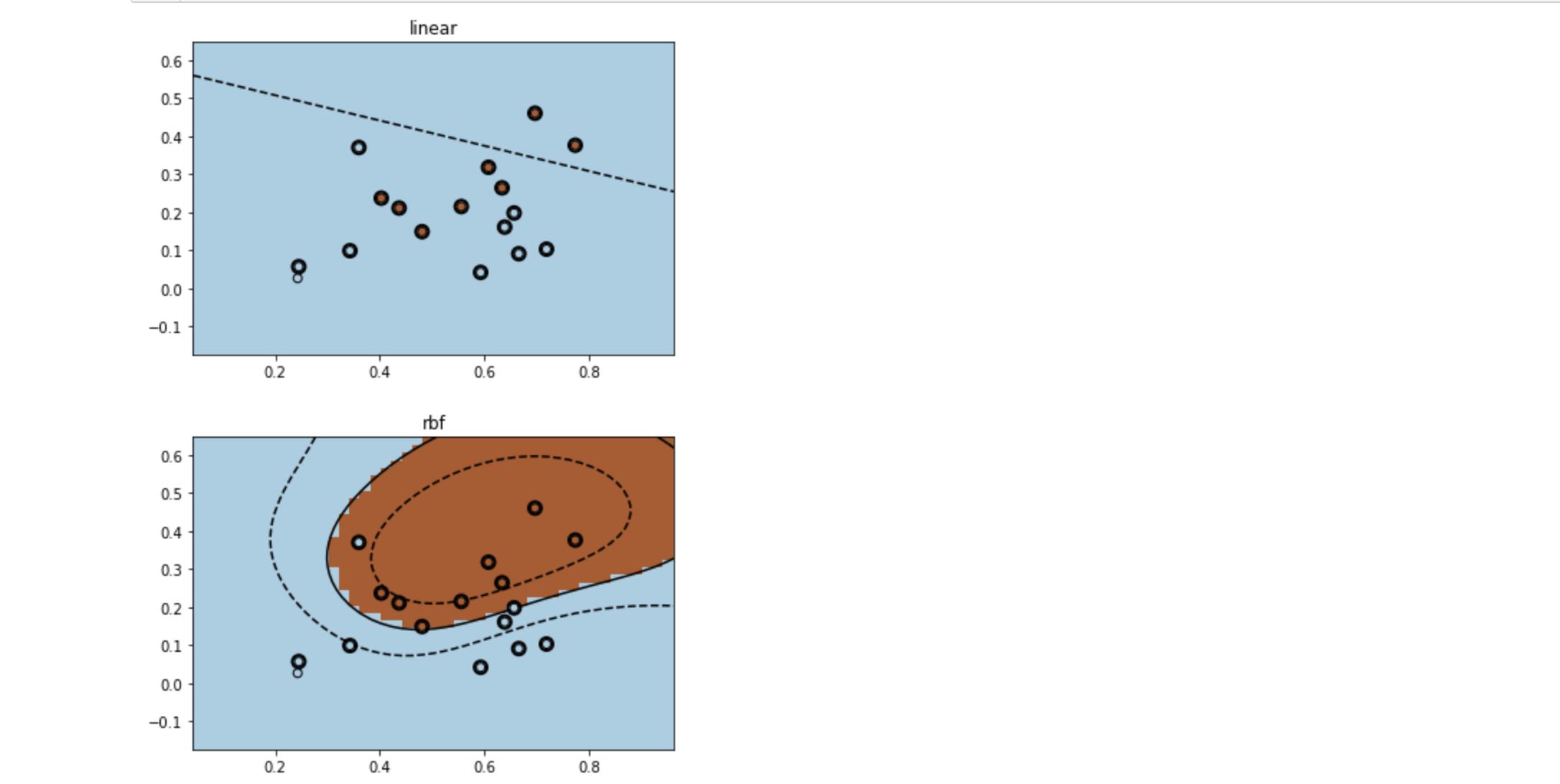
plt.contour(XX, YY, Z, colors=['k', 'k', 'k'], linestyles=['--', '-', '--'], levels=[-.5, 0, .5])

plt.title(kernel)

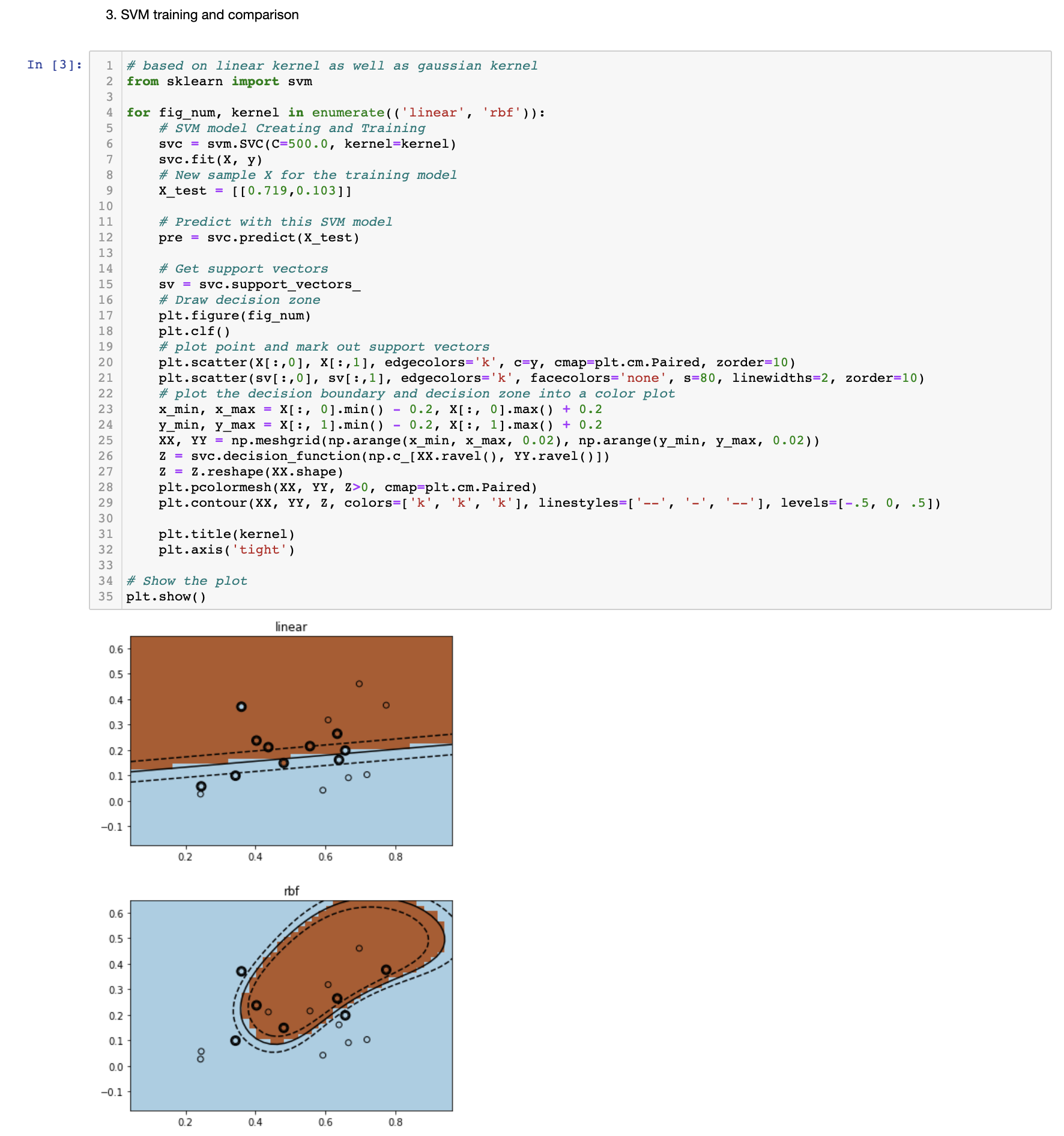
plt.axis('tight')

plt.show()





这里C的取值过小，将C的取值调整至500，测试结果为：



问题讨论：

本次实验主要进行了基于sklearn学习库中封装好的SVM类，调用SVC方法进行SVM模型构建与学习。

但是在实现过程中，如何编码对SVM模型构建以及最大距离具体求值尚未掌握。

这里需要注意正则化参数C的取值，如果C取值过小，则分类不明显，C取值越大，分类效果更好。但是分类学习有个上限，当达到上限值是，无论C值如何增大，最终的结果变化很小。