

# 机器学习实验---支持向量机

## 一、实验目的

1. 理解支持向量机算法原理，能实现支持向量机算法；
2. 针对特定应用场景及数据，实现二分类支持向量机算法，并尝试扩展到三分类任务。

## 二、实验内容

1. 从 UCI 数据库中下载一个二分类数据集，进行数据说明；
2. 用 80% 的数据训练，余下的做测试，计算分类准确度。

## 三、实验报告要求

1. 按实验内容撰写实验过程；
2. 报告中涉及到的代码，每一行需要有详细的注释；
3. 按自己的理解重新组织，禁止粘贴复制实验内容。

## 四、实验记录

```
import mglearn.plots
import numpy as np
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler
from sklearn.svm import LinearSVC
from sklearn.pipeline import Pipeline
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
warnings.simplefilter("ignore")

my_iris = load_iris() # 第一个数据集，鸢尾花数据集
#注：此处如果安装 scikit-learn 版本==1.2 的话导入数据会报错，需安装 scikit-learn 版本==1.1.1

# print(my_iris.keys())
# print(my_iris.feature_names)
# print(my_iris.target_names)
iris_X = my_iris.data[:,(2,3)] # petal length, petal width
iris_y = (my_iris.target == 2).astype(np.float64) # Iris virginica

random_X, random_y = mglearn.datasets.make_forge() # 第二个数据集，随机生成二分类数据集

# 合并两个数据集
two_data_sets_X = [iris_X, random_X]
two_data_sets_y = [iris_y, random_y]
```

```

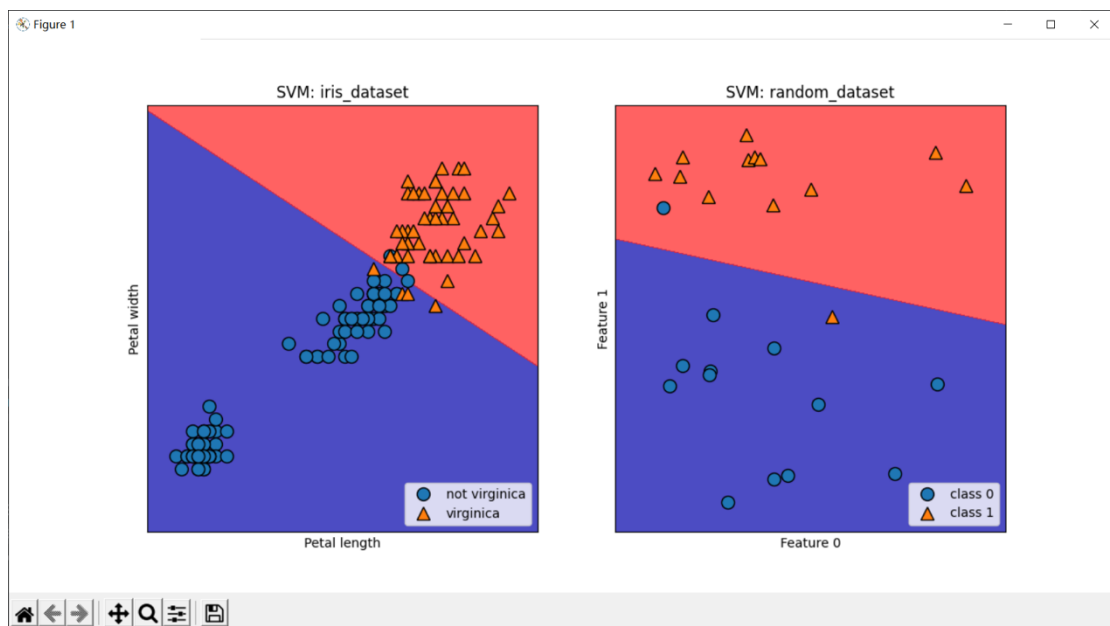
# SVM 模型
svm_clf =
Pipeline([("scaler",StandardScaler()),("linear_svc",LinearSVC(C=1,loss="hinge")),])

fig,axes = plt.subplots(1,2,figsize=(12,6))
for i,ax in zip([0,1],axes):
    X = two_data_sets_X[i]
    y = two_data_sets_y[i]
    svm_clf.fit(X,y)

mglearn.plots.plot_2d_separator(svm_clf,X,fill=True,eps=0.5,alpha=0.7,ax=ax)
# 不要忘记 ax 这个参数
mglearn.discrete_scatter(X[:,0],X[:,1],y,ax=ax) # 不要忘记 ax 这个参数
if i == 0:
    ax.set_title("SVM: iris_dataset")
    ax.set_xlabel("Petal length")
    ax.set_ylabel("Petal width")
    ax.legend(["not virginica","virginica"],loc="lower right")
else:
    ax.set_title("SVM: random_dataset")
    ax.set_xlabel("Feature 0")
    ax.set_ylabel("Feature 1")
    ax.legend(["class 0","class 1"],loc="lower right")
plt.show()

```

## 五、运行结果



## 六、实验小结

本次实验是理解支持向量机算法的原理并实现支持向量机算法，其输入是实例的特征向量，输出为实例的分类类别，以产生二分类输出，用于将数据分为两部分，因此也称为线性二分类器。实验中使用 `jupyterbook` 进行实验，并使用到了 `pandas`、`numpy`、`Matplotlib`、`sklearn` 等机器学习库，可对机器学习有初步理解。