题目：鸢尾花的数据有三类，试用SVM多分类方法（一对一，一对多）实现其分类。

要求：

1.把两种多分类方法实现的代码和运行结果放到文档中，并分析他们的异同点（不少于200字），命名规则：学号+姓名+A机房.docx

2.上传到群文件夹20250715作业中。

SVM一对一分类：

from sklearn import datasets

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

from sklearn.svm import SVC

from sklearn.multiclass import OneVsOneClassifier

from sklearn.metrics import classification\_report

# 加载鸢尾花数据集

iris = datasets.load\_iris()

X = iris.data

y = iris.target

# 划分训练集和测试集

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# 使用一对一策略

ovo\_clf = OneVsOneClassifier(SVC(kernel='linear'))

ovo\_clf.fit(X\_train, y\_train)

# 预测和评估

y\_pred\_ovo = ovo\_clf.predict(X\_test)

print("一对一策略分类结果：")

print(classification\_report(y\_test, y\_pred\_ovo, target\_names=iris.target\_names))

运行结果：

(base) fang50253@MacBook-Pro express % python3 -u "/Users/fang50253/Desktop/Files/Documents/NJFU\_My\_Github/N

JFU\_CS\_Note/24-25-2暑假软件企业实训/丁琛老师/2025-07-15/作业代码.py"

一对一策略分类结果：

precision recall f1-score support

setosa 1.00 1.00 1.00 19

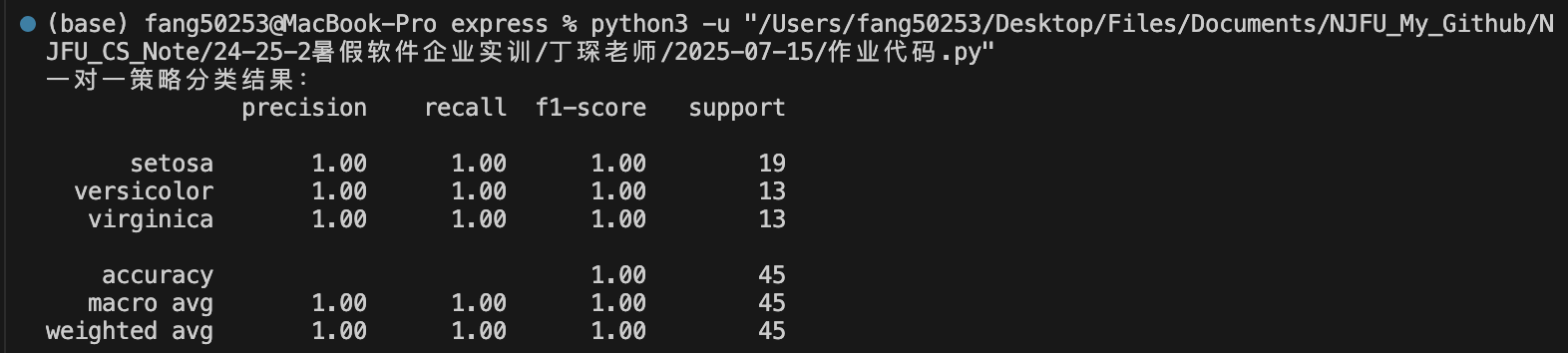
versicolor 1.00 1.00 1.00 13

virginica 1.00 1.00 1.00 13

accuracy 1.00 45

macro avg 1.00 1.00 1.00 45

weighted avg 1.00 1.00 1.00 45



SVM一对多分类：

from sklearn.multiclass import OneVsRestClassifier

# 使用一对多策略

ovr\_clf = OneVsRestClassifier(SVC(kernel='linear'))

ovr\_clf.fit(X\_train, y\_train)

# 预测和评估

y\_pred\_ovr = ovr\_clf.predict(X\_test)

print("一对多策略分类结果：")

print(classification\_report(y\_test, y\_pred\_ovr, target\_names=iris.target\_names))

一对多策略分类结果：

precision recall f1-score support

setosa 1.00 1.00 1.00 19

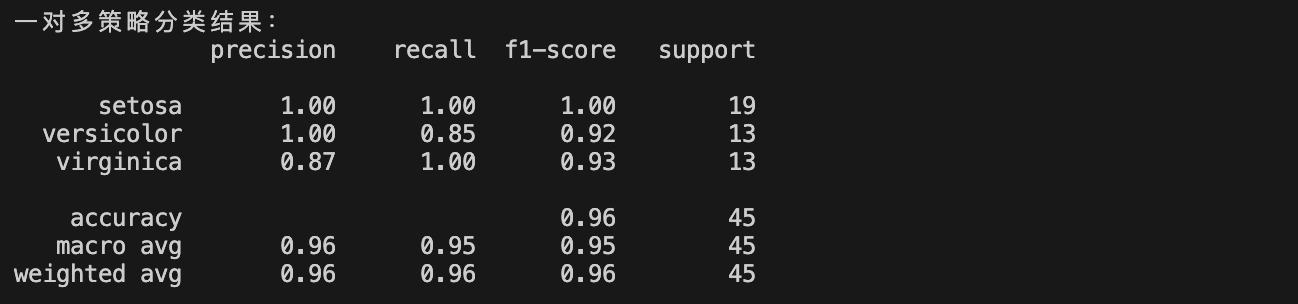
versicolor 1.00 0.85 0.92 13

virginica 0.87 1.00 0.93 13

accuracy 0.96 45

macro avg 0.96 0.95 0.95 45

weighted avg 0.96 0.96 0.96 45



相同点：两者都实现了对鸳鸯花的分类，并取得了相似的结果

不同点：一对一策略需要为每一对类别建立一个二分类器，其余两类合并为负类，构建分类器。其中一对一分类分类效果更细致，精度通常更高。一对多分类的处理速度会比一对一分类的处理速度更快。