**1.打开操作环境，在代码编辑出编写代码**

**2.导入算法库**

**from** sklearn.datasets **import** load\_iris # 导入鸢尾花数据集 #from sklearn.svm import LinearSVC # 导入支持向量机分类算法库 **from** sklearn **import** svm # 导入支持向量机模块 **from** sklearn.model\_selection **import** train\_test\_split # 导入划分数据集函数库

**3.加载数据集并划分为训练集与测试集，可以适当修改选择的数据集特征或者划分的比例**

iris=load\_iris() # 加载鸢尾花数据集 X=iris.data[:, :2] # 选择鸢尾花数据集的前两列作为分类数据特征 y=iris.target # 将鸢尾花数据集集标签赋值给y x\_train,x\_test,y\_train,y\_test = train\_test\_split(X,y,test\_size = 0.4,random\_state = 0) # 划分数据集，其中测试集占数据集的40%，其余作为训练集

**4. 创建支持向量机分类模型并使用训练集训练模型，训练后预测测试集中数据的类别并输出**

#model = LinearSVC(C=1) # 在导入支持向量机分类算法库的条件下，使用支持向量机分类语法，定义参数，创建对象model model = svm.SVC(C=1) # 在导入支持向量机模块的条件下，使用支持向量机分类语法，定义参数，创建对象model model.fit(x\_train, y\_train) # 使用创建的支持向量机分类对象训练模型，传入的参数有训练集与训练集标签 print(model.score(x\_train, y\_train)) # 输出训练集数据和标签上的平均准确度。 predicted= model.predict(x\_test) # 使用训练的模型预测测试集 print(predicted) # 输出预测标签