机器学习实验报告

姓名：卜梦煜　　学号：1120192419　　班级：07111905

1. 实验名称

使用 SVM 方法实现数据分类

2. 实验目的

理解 SVM 实现原理与实现方法，理解多分类任务的步骤，使用 SVM 模型完成多分类模型的构建与结果评估。

3. 实验内容

使用 SVM 模型完成 Iris 数据集上的多分类任务，并使用多分类任务的评价指标对实验结果进行分析。

4. 实验环境

python 3.9.4，sklearn 0.24.2，numpy 1.20.3

5. 实验过程

本实验选用 Python 语言实现 SVM 模型。数据集准备与代码设计实现步骤如下。

5.1 数据集准备

下载 Iris 数据集，阅读“iris.names”说明文件得到各列含义，并根据该说明将错误数据修正。

5.2 加载数据集与数据集划分

在加载数据集前编写转换函数，将数据集第 5 列的文本标签映射为数字标签，调用 numpy.loadtxt()方法加载数据集，存入二维数组中。

调用 numpy.split() 方法将数据集分割为训练集、测试集。调用 sklearn.model_selection.train_test_split()方法，将训练集、验证集的样本和标签分开，并固定随机数种子，方便后续对参数 C 和 gamma 取值的分析。

5.3 SVM 模型搭建与训练

利用 sklearn.pipline.Pipline()搭建 SVM 模型，流水线包括归一化层和 SVM 层。归一化层使用 sklearn.preprocessing.StandardScaler() 函数完成归一化操作。SVM 层使用

sklearn.svm.SVC()函数构建 SVM 模型，并选用高斯核函数，分类形式为多分类“ovr”，设置软间隔系数 C=1.0，核函数系数 gamma=5.0。

将训练集的样本和标签输入，训练得到模型。

5.4 实验结果评价

调用 model.predict()方法得到预测的标签值。

考虑到数据集性质，选择评价方式为多分类模型常用的评价指标，即：precision、micro-P 、 macro-P ， recall 、 micro-R 、 macro-R ， F1 、 micro-F1 、 macro-F1 。 调 用 sklearn.metrics.classification_report()函数，将预测标签值与实际标签值对比，计算出各项指标。

6. 实验结果与分析

6.1 实验结果

参数 C=1.0，gamma=5.0 时，训练集结果如下：

	precision	recall	f1-score	support
class 1	1.00	1.00	1.00	39
class 2	0.97	1.00	0.99	36
class 3	1.00	0.98	0.99	45
accuracy			0.99	120
macro avg	0.99	0.99	0.99	120
weighted avg	0.99	0.99	0.99	120

测试集结果如下：

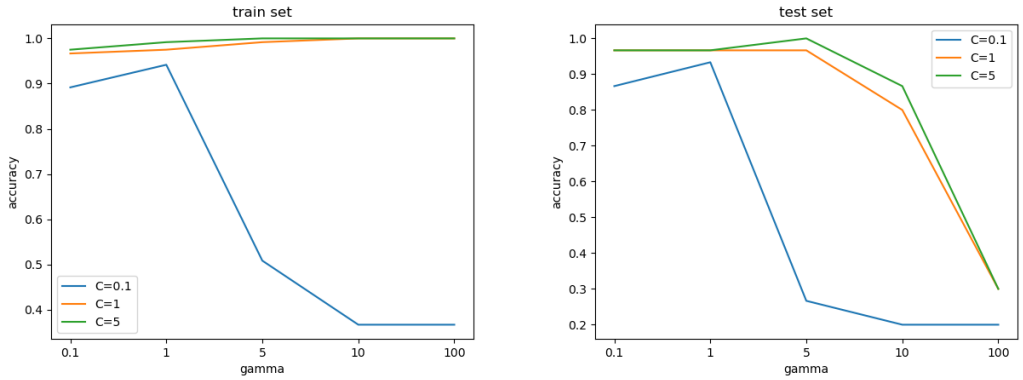
	precision	recall	f1-score	support
class 1	0.91	1.00	0.95	10
class 2	1.00	1.00	1.00	13
class 3	1.00	0.86	0.92	7
accuracy			0.97	30
macro avg	0.97	0.95	0.96	30
weighted avg	0.97	0.97	0.97	30

由结果可知，该组参数设置下，可实现较为准确的预测。

6.2 参数分析

对 SVM 模型，可调参数为软间隔系数 C、核函数系数 gamma、核函数类型。

（1）固定核函数为高斯核函数，C、gamma 分别取一系列值时，训练集、测试集的准确率（按 accuracy 算）如下：



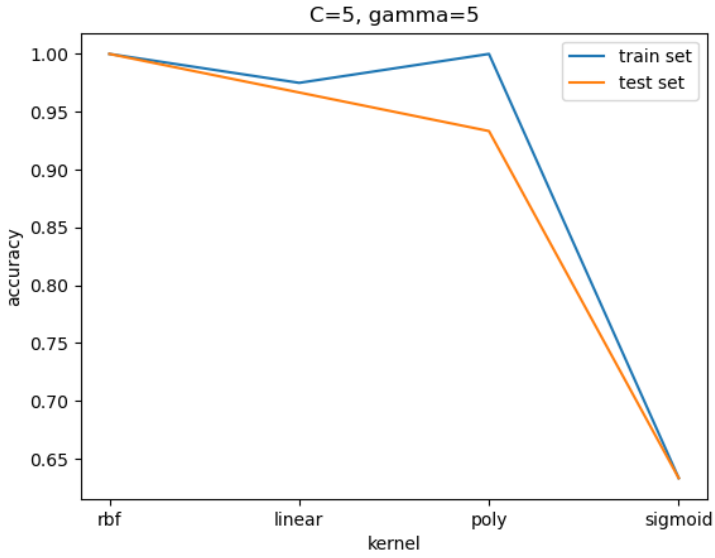
图中 C、gamma 变化时，训练集、测试集上准确率变化符合经验规律：

1）软间隔系数 C 反映对错分样本的惩罚力度。C 越小，惩罚力度越小，错分样本增加，可能出现欠拟合问题，倾向于选择简单的分类界面；C 越大，惩罚力度越大，更容易出现过拟合问题，倾向于选择复杂的分类界面。

2）核函数系数 gamma 在低维样本向高维映射时起作用，反映高维映射的复杂程度。gamma 越小，映射的维度越低，分类边界越简单；gamma 越大，映射的维度越高，训练集结果越好，但容易引起过拟合问题，泛化能力降低。通常$\text{gamma} = \frac{1}{n_feature}$。

3）对本模型，实验结果表明，C=5，gamma=5 时效果最好。

（2）固定 C=5, gamma=5, 使用不同高斯核函数，训练集、测试集的准确率（按 accuracy 算）如下：



由图可知，对 Iris 数据集，选用高斯核的模型泛化性能效果最好。

7. 心得体会

通过本次实验，我有以下收获：

（1）深入理解了 SVM 模型原理，对 SVM 模型可调参数、泛化能力、决策边界有了更直观的理解。

（2）初步掌握了 sklearn 库关于 SVM 的库函数，能够使用 sklearn、numpy、pandas、matplotlib 等库完成 SVM 模型从读取数据、搭建模型到结果可视化的全过程。