清华大学电子工程系

媒体与认知

2023-2024 学年春季学期

作业 3

孙文辉 2024 年 5 月 26 日

理论部分

1 单选题（15 分）

1.1 D

1.2 C

1.3 D

1.4 D

1.5 B

2 计算题（15 分）

2.1 给定两个类别的样本分别为:

$$\omega_1 : \{(3,1),(2,2),(4,3),(3,2)\}$$
$$\omega_2 : \{(1,3),(1,2),(-1,1),(-1,2)\}$$

试利用 LDA，将样本特征维数压缩为一维。

$$S_j = \frac{1}{n_i}\sum_{k=1}^{N_i}\{(x_k^i - \mu_i)(x_k^i - \mu_i)^T\} \tag{1}$$

$$S_1 = \begin{bmatrix} 0.5 & 0.25 \\ 0.25 & 0.5 \end{bmatrix} \quad S_2 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{bmatrix} \tag{2}$$

类内均值为：

$$\mu_1 = \begin{bmatrix} 3.0 & 2.0 \end{bmatrix}, \quad \mu_2 = \begin{bmatrix} 0.0 & 2.0 \end{bmatrix} \tag{3}$$

总体均值为：

$$\mu = \begin{bmatrix} 1.5 & 2.0 \end{bmatrix} \tag{4}$$

类间散度矩阵为：

$$S_B = \begin{bmatrix} 2.25 & 0 \\ 0 & 0 \end{bmatrix} \tag{5}$$

类内散度矩阵为：

$$S_W = \begin{bmatrix} 0.75 & 0.375 \\ 0.375 & 0.5 \end{bmatrix} \tag{6}$$

由 $S_W^{-1}S_B$ 的特征值和特征向量可得：

$$S_W^{-1}S_B = \begin{bmatrix} 4.8 & 0 \\ -3.6 & 0 \end{bmatrix} \tag{7}$$

$$\lambda = 4.8, \quad \mathbf{w}^T = \begin{bmatrix} 0.8 & -0.6 \end{bmatrix} \tag{8}$$

即投影方向为 $\mathbf{w}^T = \begin{bmatrix} 0.8 & -0.6 \end{bmatrix}$，则样本在该方向上的投影为：

$$\begin{aligned}
\mathbf{w}\mathbf{x}_1^T &= 0.8 * 3 - 0.6 * 2 = 1.8 \\
\mathbf{w}\mathbf{x}_2^T &= 0.8 * 2 - 0.6 * 2 = 0.4 \\
\mathbf{w}\mathbf{x}_3^T &= 0.8 * 4 - 0.6 * 3 = 1.4 \\
\mathbf{w}\mathbf{x}_4^T &= 0.8 * 3 - 0.6 * 2 = 1.2 \\
\mathbf{w}\mathbf{x}_5^T &= 0.8 * 1 - 0.6 * 3 = -1 \\
\mathbf{w}\mathbf{x}_6^T &= 0.8 * 1 - 0.6 * 2 = -0.4 \\
\mathbf{w}\mathbf{x}_7^T &= 0.8 * (-1) - 0.6 * 1 = -1.4 \\
\mathbf{w}\mathbf{x}_8^T &= 0.8 * (-1) - 0.6 * 2 = -2
\end{aligned} \tag{9}$$

有：

$$\omega_1 : \{1.8, 0.4, 1.4, 1.2\}, \quad \omega_2 : \{-1, -0.4, -1.4, -2\} \tag{10}$$

2.2 模型训练通常需要大量的数据，假设某采集的数据集包含 80% 的有效数据和 20% 的无效数据。采用一种算法判断数据是否有效，其中无效数据被成功判别为无效数据的概率为 90%，而有效数据被误判为无效数据的概率为 5%。如果某条数据经过该算法被判别为无效数据，则根据贝叶斯定理，这条数据是无效数据的概率是多少？(提示：全概率公式 $P(Y) = \sum_{i=1}^{N} P(Y|X_i)P(X_i)$)

$$\begin{aligned} P(\text{无效数据}) &= \frac{P(\text{无效}) * P(\text{判为无效})}{P(\text{无效}) * P(\text{判为无效}) + P(\text{有效}) * P(\text{判为无效})} \\ &= \frac{0.2 * 0.9}{0.2 * 0.9 + 0.8 * 0.05} \\ &\approx 0.818 \end{aligned} \tag{11}$$

2.3 设有两类正态分布的样本集，第一类均值为 $\mu_1 = [2, -1]^T$，第二类均值为 $\mu_2 = [1, 1]^T$。两类样本集的协方差矩阵和出现的先验概率都相等：$\Sigma_1 = \Sigma_2 = \Sigma = \begin{bmatrix} 4 & 2 \\ 2 & \frac{4}{3} \end{bmatrix}$，$p(\omega_1) = p(\omega_2)$。试计算分类界面，并对特征向量 $x = [6, 2]^T$ 分类。

由于各类协方差矩阵相等，先验概率相等，得线性判别函数为

$$g_{LDF}(x) = A_i^T x + b_i$$

其中 $A_i = \Sigma_0^{-1}\mu_i$,$b_i = -\frac{1}{2}\mu_i^T\Sigma_0^{-1}\mu_i$ 得：

$$g_1(x) = [3.5 \quad -6]x - 6.5$$
$$g_2(x) = [-0.5 \quad 1.5]x - 0.5$$

由

$$g(x) = g_1(x) - g_2(x) = 0$$

得分类界面为

$$4x_1 - 7.5x_2 = 6$$

$x = [6, 2]^T$ 时，g(x)=4>0, 故其应当分至第一类。

2.4 给定异或的样本集

$D = \left\{\left((0,0)^T, -1\right), \left((0,1)^T, 1\right), \left((1,0)^T, 1\right), \left((1,1)^T, -1\right)\right\}$ 该样本集是线性不可分的，可采用如下所示的多项式函数 $\phi(\mathbf{x})$ 将样本 $D = \{(\mathbf{x}_n, y_n)\}$ 映射为 $D_\phi = \{(\phi(\mathbf{x}_n), y_n)\}$，其中 $\phi(\mathbf{x})$ 满足

$$\phi_1(\mathbf{x}) = 2(x_1 - 0.5)$$
$$\phi_2(\mathbf{x}) = 4(x_1 - 0.5)(x_2 - 0.5)$$

(1) 给出映射后的样本集；

(2) 在映射后的样本集中，设计一个线性 SVM 分类器，给出支持向量及分类界面。

(1) $D_\phi = \{((-1,1)^T, -1), ((-1,-1)^T, 1), ((1,-1)^T, 1), ((1,1)^T, -1)\}$

(2) 其核函数为

$$K = \begin{bmatrix} 9 & 1 & 1 & 1 \\ 1 & 9 & 1 & 1 \\ 1 & 1 & 9 & 1 \\ 1 & 1 & 1 & 9 \end{bmatrix}$$

基函数为

$$\phi(x_1) = [1, 1, -\sqrt{2}, 1, -\sqrt{2}, \sqrt{2}]^T$$
$$\phi(x_2) = [1, 1, \sqrt{2}, 1, -\sqrt{2}, -\sqrt{2}]^T$$
$$\phi(x_3) = [1, 1, -\sqrt{2}, 1, \sqrt{2}, -\sqrt{2}]^T$$
$$\phi(x_4) = [1, 1, \sqrt{2}, 1, \sqrt{2}, \sqrt{2}]^T$$

目标函数分别对 α_i 求导并令导数等于 0，解得

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \frac{1}{8}$$

则均为支持向量。计算得到权值向量

$$w = [0, 0, 0, 0, 0, -\frac{\sqrt{2}}{2}]^T$$

代入任一一个样本数据，解得 b=0。

由此得到决策面

$$g(x) = \sqrt{2}x_2 + 0 = 0,\ x_2 = 0$$

2.5 使用 KMeans 算法对 2 维空间中的 6 个点 $(0,2)$,$(2,0)$,$(2,3)$,$(3,2)$,$(4,0)$,$(5,4)$ 进行聚类，距离函数选择欧氏距离 $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$。

(1) 起始聚类中心选择 (0,0) 和 (4,3)，计算聚类中心；

(2) 起始聚类中心选择 (1,4) 和 (3,1)，计算聚类中心。

(1) 起始聚类中心为 (0,0) 和 (4,3)，迭代如下：

迭代次数	聚类 1	新中心	聚类 2	新中心
1	(0,2),(2,0)	(1,1)	(2,3),(3,2),(4,0),(5,4)	(3.5,2.25)
2	(0,2),(2,0)	(1,1)	(2,3),(3,2),(4,0),(5,4)	(3.5,2.25)

故聚类中心为 (1,1) 和 (3.5,2.25)

(2) 起始聚类中心为 (1,4) 和 (3,1)，迭代如下：

迭代次数	聚类 1	新中心	聚类 2	新中心
1	(0,2),(2,3)	(1,2.5)	(2,0),(3,2),(4,0),(5,4)	(3.5,1.5)
2	(0,2),(2,3)	(1,2.5)	(2,0),(3,2),(4,0),(5,4)	(3.5,1.5)

故聚类中心为 (1,2.5) 和 (3.5,1.5)

编程部分

3 编程作业报告

这里选择助教提供的模型

3.1 代码补全

代码补全详见附件代码

3.2 程序验证

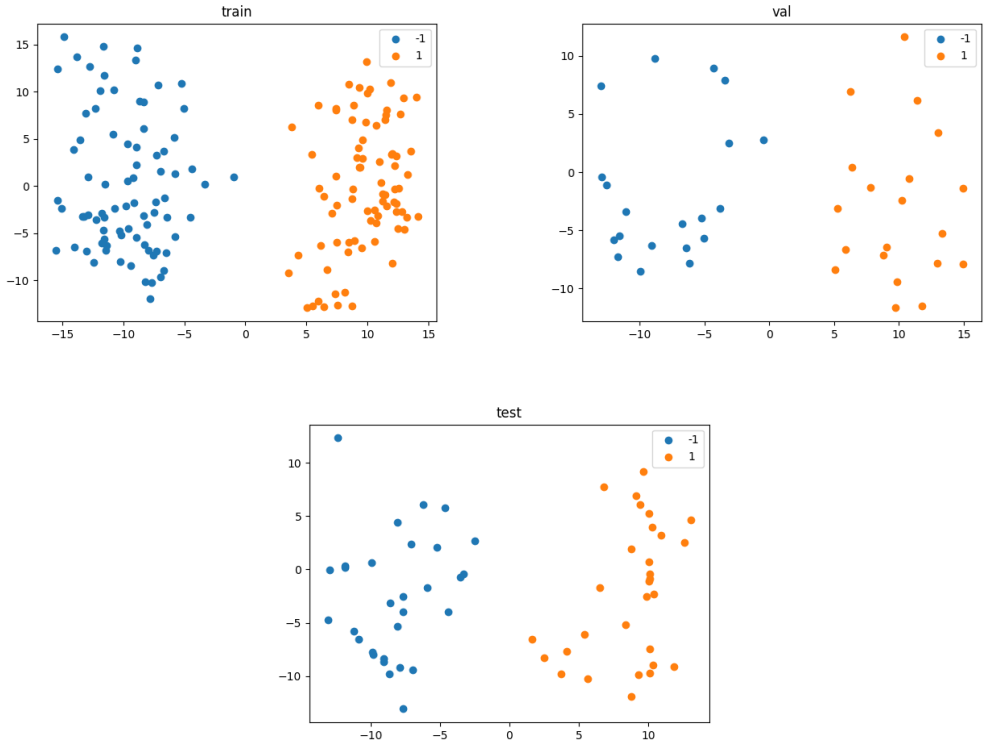
运行 *check.py*

```
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code> python check.py
Linear successully tested!
Hinge successfully tested!
SVM_HINGE successfully tested!
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code>
```

图 1: 程序运行结果

3.3 数据预处理

将降维后的 3 个数据集分别对应的可视化结果如下：

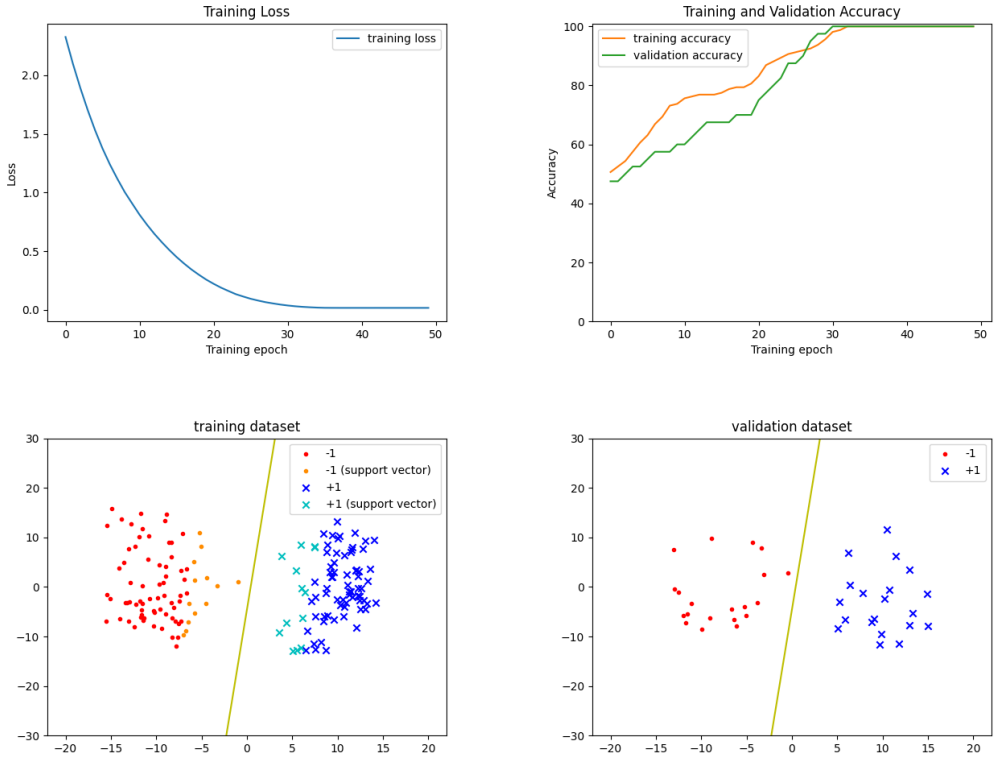


可以看出，经过数据降维之后，数据集的维度降低，但数据的分布特征基本保持不变。从图中可以较为明显的将两种数据进行区分

3.4 训练，验证和测试

模型训练

loss 曲线，分类准确率，训练集及验证集可视化结果如下：



模型测试

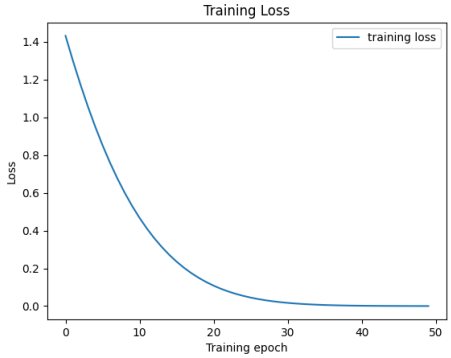
```
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code> python test_svm.py
Test accuracy = 100.0%
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code>
```

图 2: 测试结果

从测试结果可看出，模型在 30 轮左右即达到了较低的 loss 与较高的验证集正确率。在测试集上的准确率为 100%，表明模型的泛化能力较好。

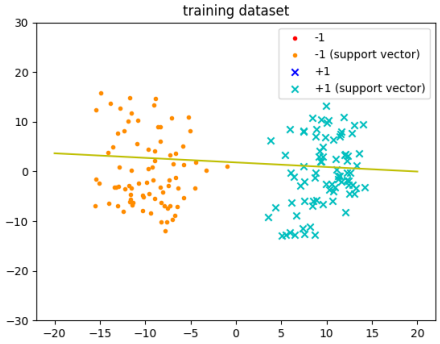
3.5 调整正则化系数 C

C = 1e-6

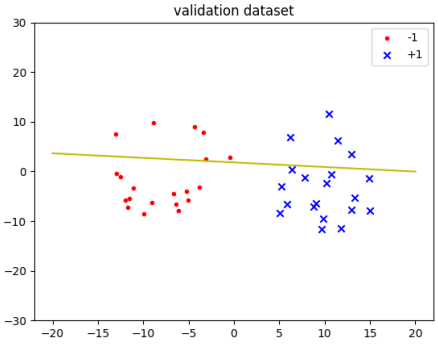


(a) loss 曲线

(b) 分类准确率



(c) 训练集可视化



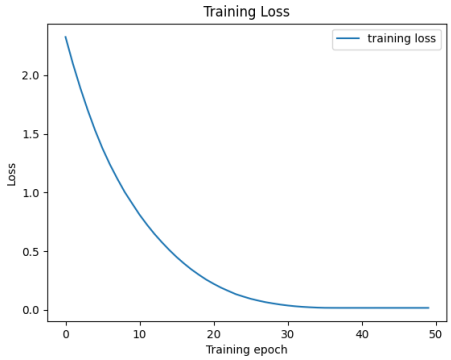
(d) 验证集可视化

此时模型在测试集上的准确率为：

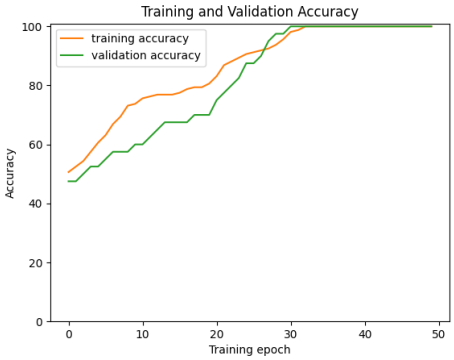
```
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code> python test_svm.py
Test accuracy = 58.3%
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code>
```

图 3: 测试准确率

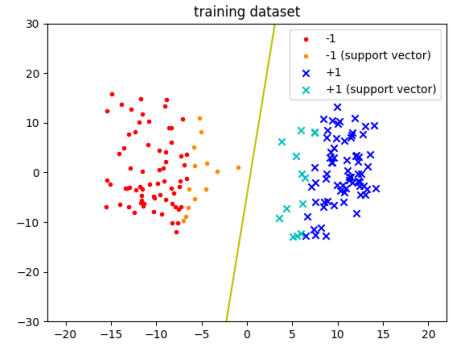
C=1e-3



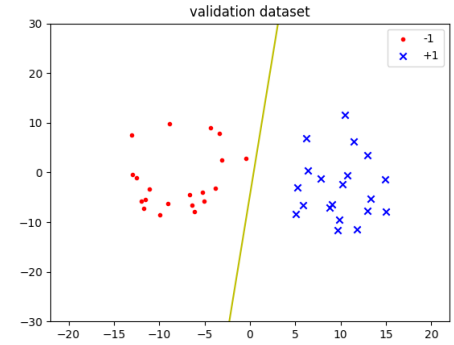
(a) loss 曲线



(b) 分类准确率



(c) 训练集可视化



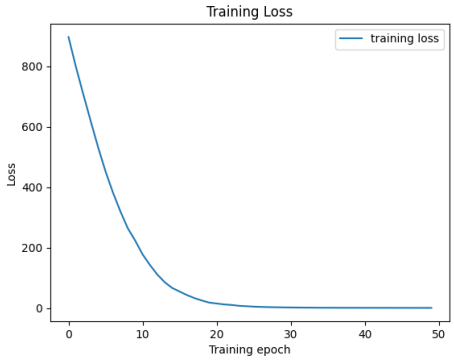
(d) 验证集可视化

此时模型在测试集上的准确率为：

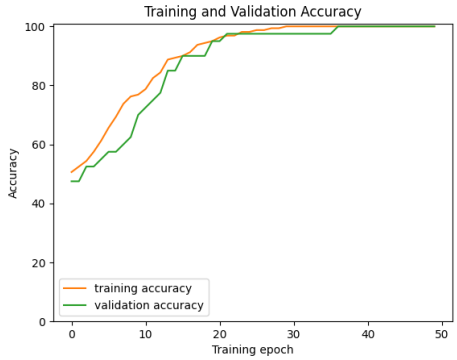
```
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code> python test_svm.py
Test accuracy = 100.0%
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code>
```

图 4: 测试准确率

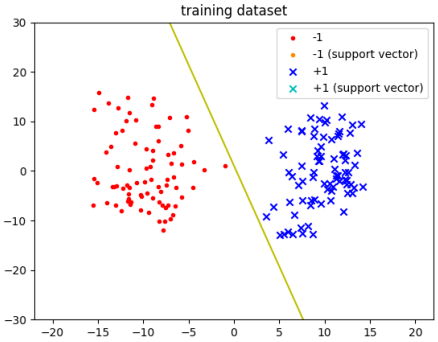
C=1



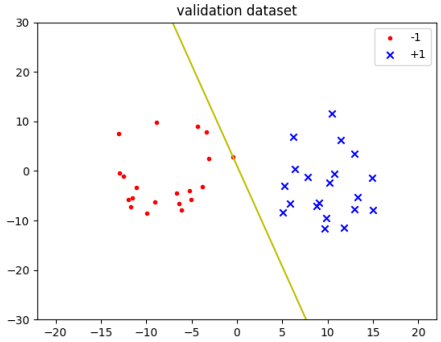
(a) loss 曲线



(b) 分类准确率



(c) 训练集可视化



(d) 验证集可视化

此时模型在测试集上的准确率为：

```
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code> python test_svm.py
Test accuracy = 98.3%
PS F:\Learning\2023_2024spring\machine_learning\HW\hw3\hw3_release\code>
```

图 5: 测试准确率

C 对分类效果的影响：

- C=1e-6 时，即 C 过小，模型欠拟合，训练集和验证集准确率较低，loss 较高。

- C=1e-3 时，模型拟合较好，训练集和验证集准确率较高，loss 较低。
- C=1 时，即 C 过大，模型过拟合，训练集准确率较高，验证集准确率较低，loss 较低，表现为分类平面过于倾斜。

4 总结建议

本次编程作业是使用 svm 对数据进行分类，借助助教的指示，完成代码并不难，其中最复杂也最关键的点在与代码中各个变量之间的维度匹配。如一开始在计算 output 时，我没有看见助教给的提示，直接使用了 $y = torch.matmul(W, x.t()) + b$ ，与提示中的代码恰好做了转置，之后一路均没有问题，在写完后发现结果均正确，但是想将 output 改为正常的转置时，发现后续代码怎么也不对，经过仔细检查发现是因为在计算 loss 时的 tensor 维度不匹配，花费了相当多的时间。当然这种写法似乎比起原先的更为复杂，因此我最终保留了较为简洁的原版。在编程时，应该仔细检查每一步的维度是否匹配，以免出现不必要的错误。同时也很感谢助教在作业中给出的提示，让我能够更好的理解代码的运行逻辑。