

哈尔滨工业大学

模式识别与机器学习实验报告

实验 四

题	目	多层感知机实验
学	院	未来技术学院
专	业	人工智能
学	号	2023112419
学	生	陈铠
任	课 教 师	刘扬

哈尔滨工业大学计算机科学与技术学院

2025 年秋季

一、 实验内容

（一）简要介绍

1.核心任务

多层感知机（**Multilayer Perceptron, MLP**）是一种基础的前馈神经网络结构，可以实现从输入特征到目标输出的非线性映射。

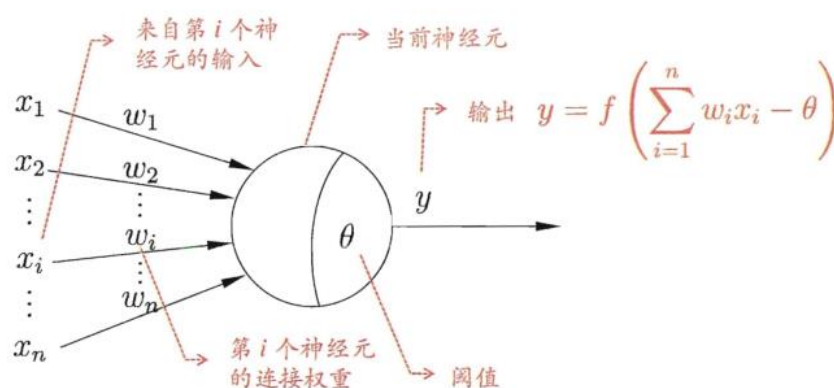
其核心任务是通过学习一组权重参数，使得网络能够准确地对输入样本进行分类或预测。

2.数学模型

多层感知机的基本单元是 M-P 神经元，其数学表达为：

$$y = f\left(\sum_{i=1}^n w_i x_i + b\right)$$

其中， x_i 为输入特征， w_i 为权重， b 为偏置， $f(\cdot)$ 为激活函数。



对于单个隐藏层 MLP，设输入层为 $x \in \mathbb{R}^n$ ，输入层为 $y \in \mathbb{R}^c$,隐藏层有 m 个神经元则第 j 个神经元的输出为

$$h_j = f\left(\sum_{i=1}^n w_{ij}^{(1)} x_i + b_j^{(1)}\right), j = 1, 2, \dots, m$$

第 k 个输出为

$$y_k = g\left(\sum_{j=1}^m w_{jk}^{(2)} h_j + b_k^{(2)}\right), k = 1, 2, \dots, c$$

矩阵形式即

$$\mathbf{h} = f(\mathbf{W}^{(1)T} \mathbf{x} + \mathbf{b}^{(1)})$$

$$\mathbf{y} = g(\mathbf{W}^{(2)T} \mathbf{h} + \mathbf{b}^{(2)})$$

其中 $\mathbf{W}^{(1)} \in \mathbb{R}^{n \times m}$ 是输入层到隐藏层的权重矩阵， $\mathbf{b}^{(1)} \in \mathbb{R}^m$ 是隐藏层偏置向量， $f(\cdot)$ 对向量每个元素独立作用。 $\mathbf{W}^{(2)} \in \mathbb{R}^{m \times c}$ 是隐藏层到输出层的权重矩阵， $\mathbf{b}^{(2)} \in \mathbb{R}^c$ 是输出层偏置向量， $g(\cdot)$ 是输出激活函数，常使用 softmax。若有多层隐藏层，则

$$\begin{aligned} \mathbf{h}^{(1)} &= f^{(1)}(\mathbf{W}^{(1)T} \mathbf{x} + \mathbf{b}^{(1)}) \\ \mathbf{h}^{(2)} &= f^{(2)}(\mathbf{W}^{(2)T} \mathbf{h}^{(1)} + \mathbf{b}^{(2)}) \\ &\vdots \\ \mathbf{y} &= g(\mathbf{W}^{(L+1)T} \mathbf{h}^{(L)} + \mathbf{b}^{(L+1)}) \end{aligned}$$

3. 优化方法

对于分类任务，输出层通常使用 softmax 激活函数，损失函数采用交叉熵：

$$L(\mathbf{W}, \mathbf{b}) = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^c y_{ik} \log \hat{y}_{ik}$$

反向传播通过链式法则计算梯度：

输出层梯度为

$$\frac{\partial L}{\partial \mathbf{W}^{(L+1)}} = \mathbf{h}^{(L)} (\boldsymbol{\delta}^{(L+1)})^T, \frac{\partial L}{\partial \mathbf{b}^{(L+1)}} = \boldsymbol{\delta}^{(L+1)}$$

隐藏层梯度为

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{W}^{(l)}} &= \mathbf{h}^{(l-1)} (\boldsymbol{\delta}^{(l)})^T, \frac{\partial L}{\partial \mathbf{b}^{(l)}} = \boldsymbol{\delta}^{(l)} \\ \mathbf{W}^{(l)} &\leftarrow \mathbf{W}^{(l)} - \eta \frac{\partial L}{\partial \mathbf{W}^{(l)}}, \mathbf{b}^{(l)} \leftarrow \mathbf{b}^{(l)} - \eta \frac{\partial L}{\partial \mathbf{b}^{(l)}} \end{aligned}$$

（二）实验研究内容

与线性分类器对比

不同数据量，不同超参数，比较实验效果

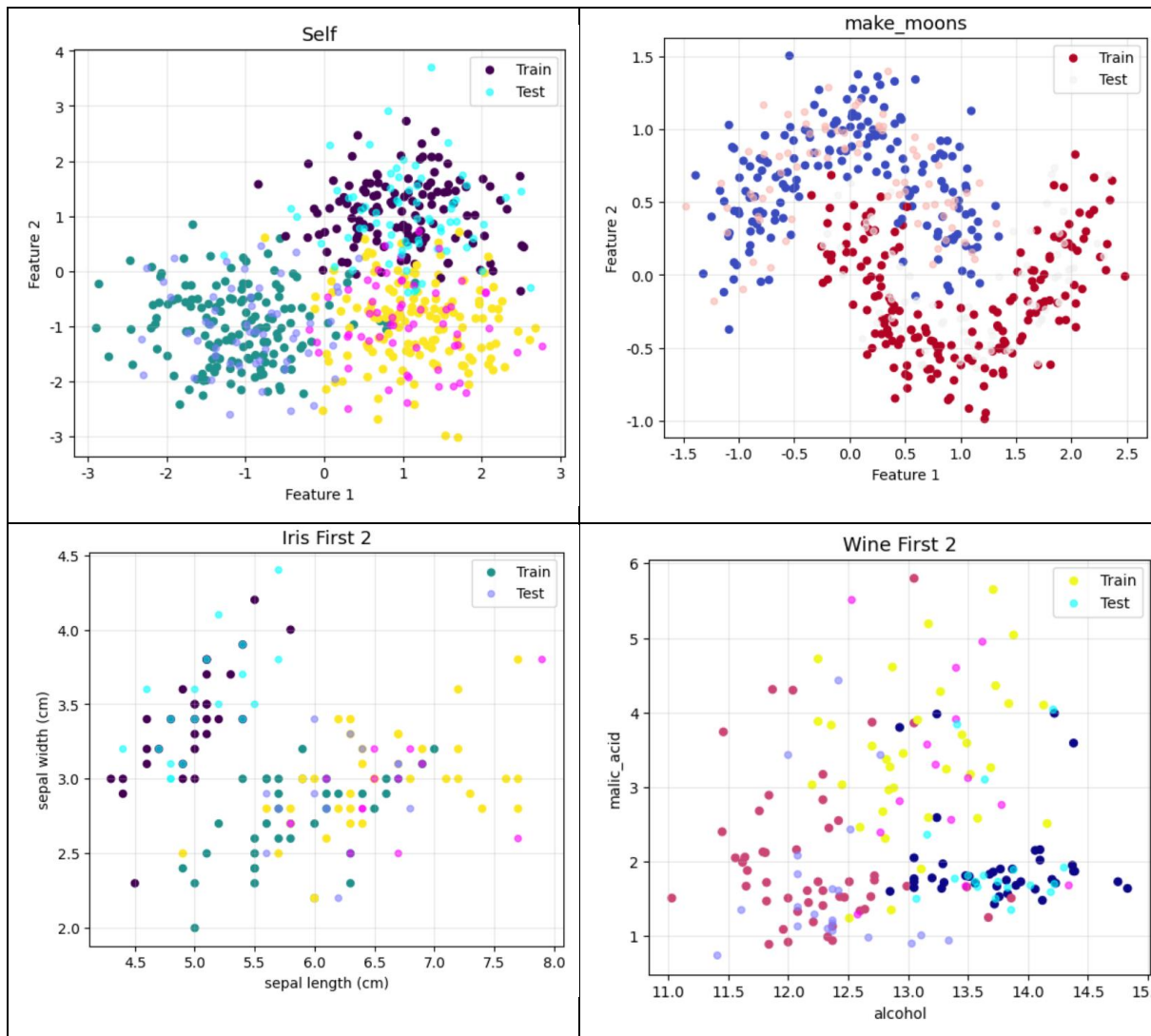
二、 实验环境

google colab 云平台

jupyter notebook

三、 实验结果及分析

本次实验共使用四种数据集，分别为使用高斯分布的类别为 3 的数据集、sklearn.datasets 库中的 make_moons 函数生成的数据集、load_iris 鸢尾花数据集、load_wine 葡萄酒数据集。

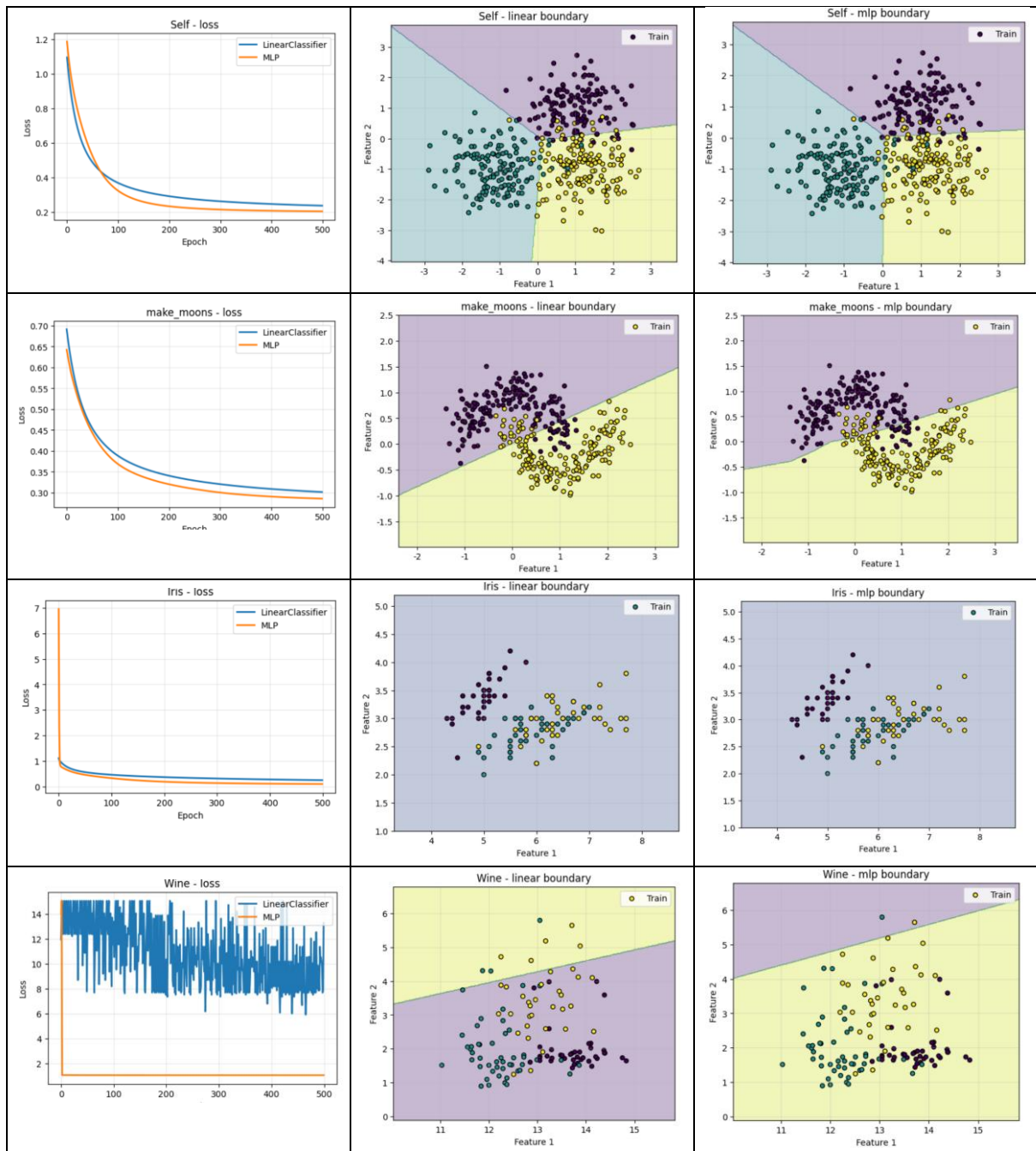


dataset	samples	dimension	categories	test size
self	600	2	3	0.3
make_moons	500	2	2	0.3
Iris	150	4	3	0.3
wines	178	13	3	0.3

线性分类器和 MLP 对比

以下为两种模型的训练损失和验证集准确率，迭代次数均为 500

dataset	lin loss	lin acc	mlp loss	mlp acc
self	0.2375	0.878	0.2046	0.872
make_moons	0.3016	0.873	0.2858	0.867
Iris	0.2567	1.000	0.1055	1.000
wines	13.2690	0.667	1.0860	0.389



从表中可以看出，在线性可分的数据集（如 Iris）中，两种模型均能取得极高的准确率，说明线性模型已足以完成分类任务。而在具有非线性边界特征的数据集（如 make_moons、自建数据集）上，MLP 能通过非线性激活函数（ReLU）捕捉复杂边界特征，表现出更好的拟合能力与灵活性。

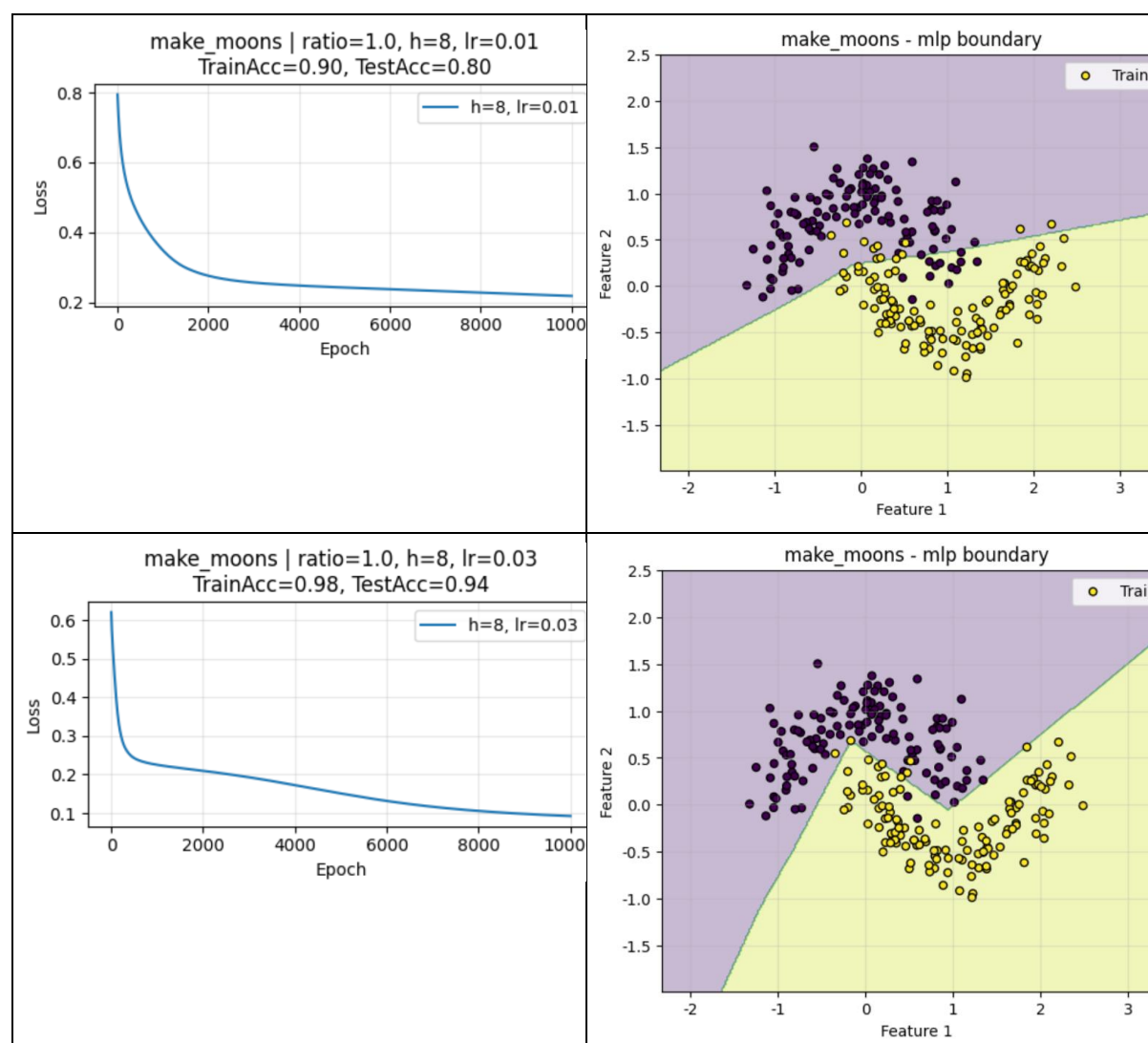
但是，在 Wine 数据集上，MLP 的表现不如线性分类器，主要原因可能包括：数据量较小，模型较深时容易过拟合；超参数（学习率、隐藏层大小）未充分调优；特征归一化或输入分布对网络训练的影响较大。总体上，MLP 在非线性数据中更具优势，但其性能依赖于合理的网络结构和超参数配置。

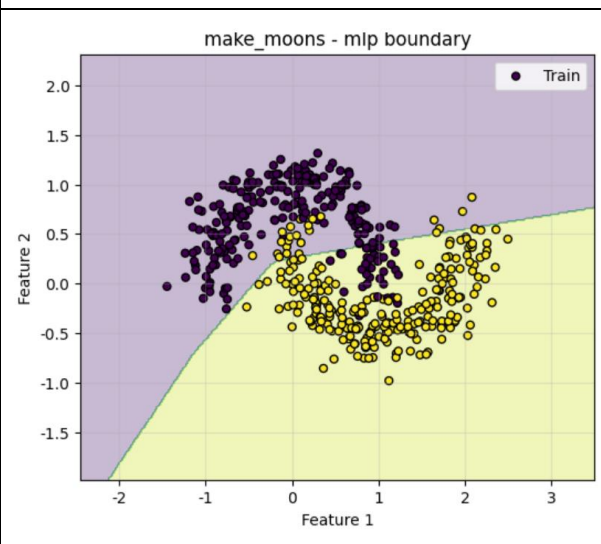
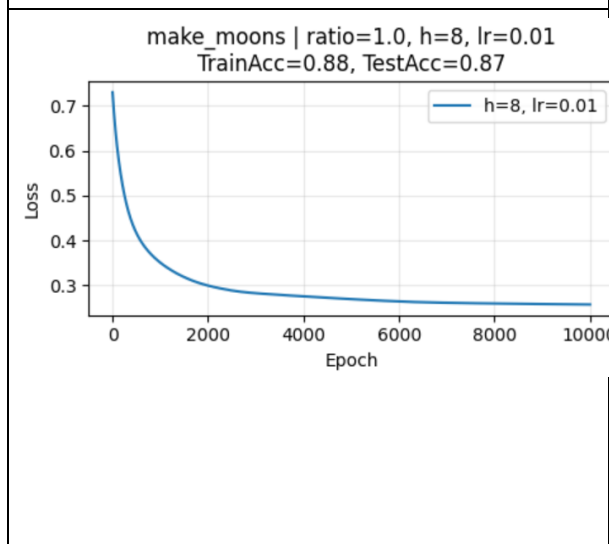
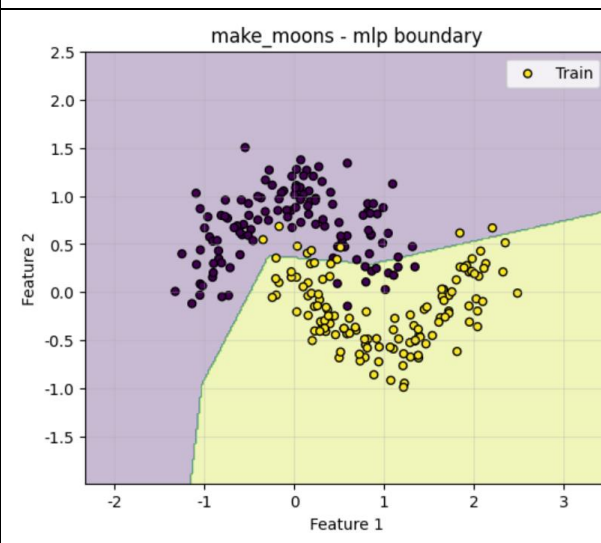
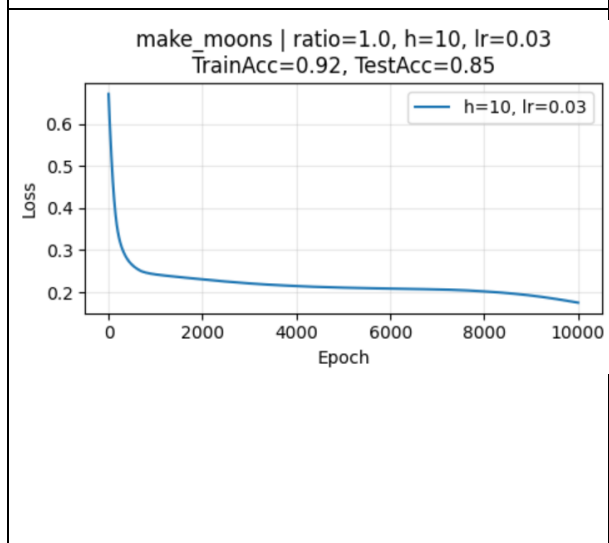
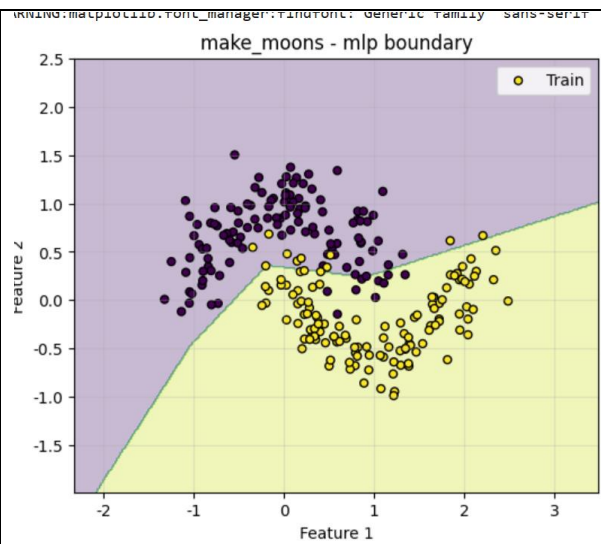
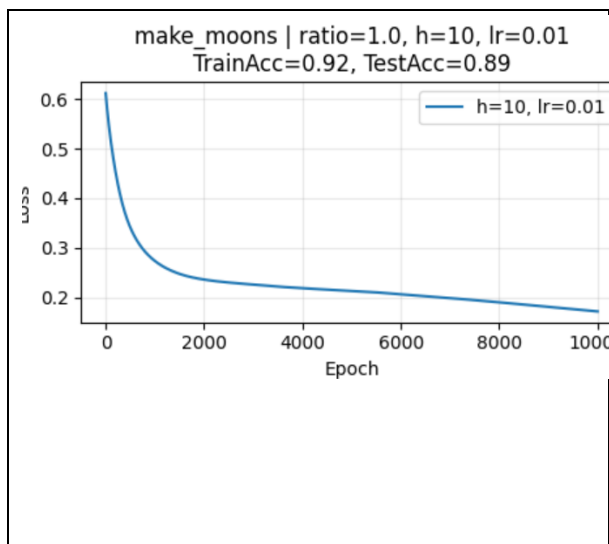
MLP 超参数对比

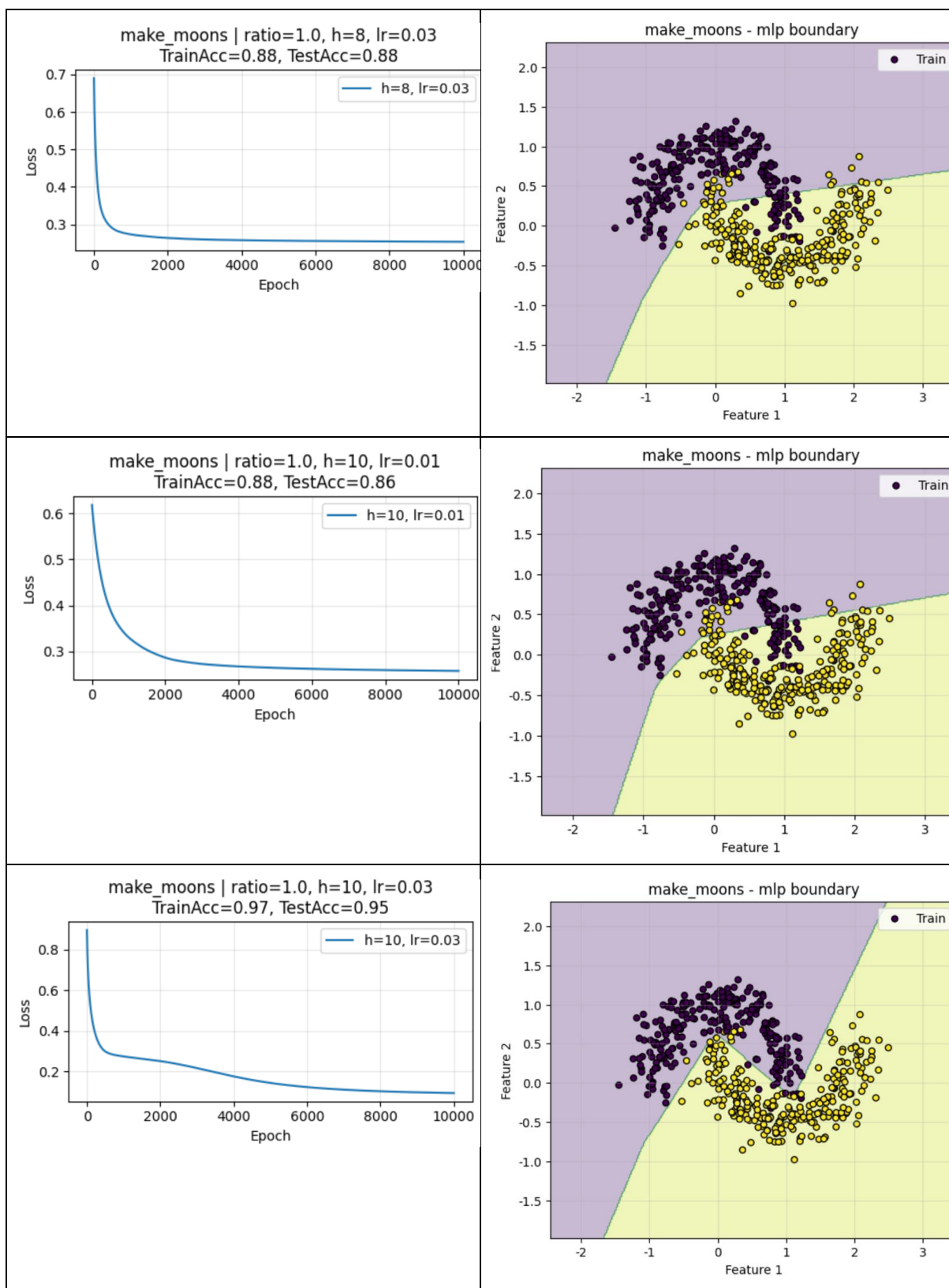
以下从数据集大小 n ，隐藏层维度 d ，学习率 lr 研究 MLP。

迭代次数均为 10000 次，使用 make_moons

n	d	lr	loss	make_moons acc
500	8	0.01	0.2178	0.800
500	8	0.03	0.0925	0.943
500	10	0.01	0.1713	0.886
500	10	0.03	0.1752	0.848
1000	8	0.01	0.2573	0.871
1000	8	0.03	0.2535	0.876
1000	10	0.01	0.2575	0.862
1000	10	0.03	0.0934	0.952







根据 `make_moons` 数据集的对比结果，学习率(lr)从 0.01 提升到 0.03，模型收敛更快、loss 更低、准确率更高 合适的学习率能加速收敛，但过大会引起震荡甚至性能下降隐藏层维度(d)从 8 到 10，准确率变化不大，模型容量略增能提升表示能力，但对小规模任务收益有限数据量(n)从 500 增加到 1000，模型收敛更稳定、准确率整体提升，

更多训练样本可提升泛化性能，验证了数据规模的重要性。

综上，MLP 对学习率敏感度最高，其次是数据量，而隐藏层维度在该任务中影响相对较小。合理的超参数选择能显著改善模型性能与收敛速度。

四、 结论

1. 线性分类器 在特征线性可分的任务中具有良好表现，训练简单、收敛快。
2. 多层感知机（MLP） 通过引入非线性激活函数与隐藏层结构，能够学习更复杂的决策边界，对非线性任务具有明显优势。
3. MLP 的性能依赖于 数据量、学习率、网络结构等超参数 的综合调优；在小数据集上，若未正则化或调参不当，容易出现过拟合或收敛不稳。
4. 实验验证了神经网络模型在非线性分类任务中的有效性，也说明了深度学习中模型复杂度与数据规模的匹配原则。

五、 参考文献

- [1]（美）SHELDON AXLER 著；杜现坤，刘大艳，马晶译. 线性代数应该这样学 第3版[M]. 北京：人民邮电出版社, 2016.10.
- [2]周志华著. 机器学习[M]. 北京：清华大学出版社, 2016.01.
- [3]谢文睿，秦州编著. 机器学习公式详解[M]. 北京：人民邮电出版社, 2021.03.
- [4]李航著. 统计学习方法 第2版[M]. 北京：清华大学出版社, 2019.05.