函数拟合实验报告

2251737 陈德凯

2025.3.20

# 目 录

[函数拟合实验报告 I](#_Toc193407143)

[目 录 I](#_Toc193407144)

[1 函数定义 1](#_Toc193407145)

[2 数据采集 1](#_Toc193407146)

[3 模型描述 1](#_Toc193407147)

[3.1 激活函数描述 1](#_Toc193407148)

[3.2 反向传播修正参数 1](#_Toc193407149)

[4 拟合效果 2](#_Toc193407150)

# 函数定义

使用较为平滑且复杂的函数：

作为实验对象。

# 数据采集

选择导数较小的作为采样区间，采集10000个点生成x数据，再将x代入到函数中取10000个y点。以此得出10000个坐标值。

# 模型描述

在模型构建中仅使用numpy实现

## 激活函数描述

ReLU函数定义为，它是一个分段线性的非负函数。ReLU及其组合能够表示任何分段线性函数，这是因为通过调整权重和偏置，ReLU单元可以在输入空间中创建不同的线性区域。

考虑单个ReLU单元，其中是权重向量，是偏置项。通过选择适当的和，我们可以让这个单元在特定区间内为线性函数，在其他区间内为零。这意味着，单个ReLU单元可以用来构造分段线性函数。

多个ReLU单元的线性组合可以用来逼近更加复杂的函数。具体来说，如果我们有一个两层ReLU网络，第一层由若干ReLU单元组成，第二层是对这些单元输出的加权和，则该网络可以表示为：其中，是隐藏层神经元的数量，,,分别是第个神经元的权重、输入权重和偏置，是最终的偏置项。

在代码中，具体表现为前向传播。

代码如下：

predition = np.dot(np.dot(x, w1)+b1, w2) + b2

## 反向传播修正参数

对于每个参数（在这里指 ），我们使用梯度下降法更新它们：

其中，是学习率，是损失函数，是损失函数关于的梯度

参数更新：

使用激活值和误差来更新，np.dot(a.T, dz)计算了的梯度

除以样本数量len(x)是为了归一化梯度，避免批量大小对学习率的影响

参数更新：

偏置的更新直接基于每批次样本的平均误差

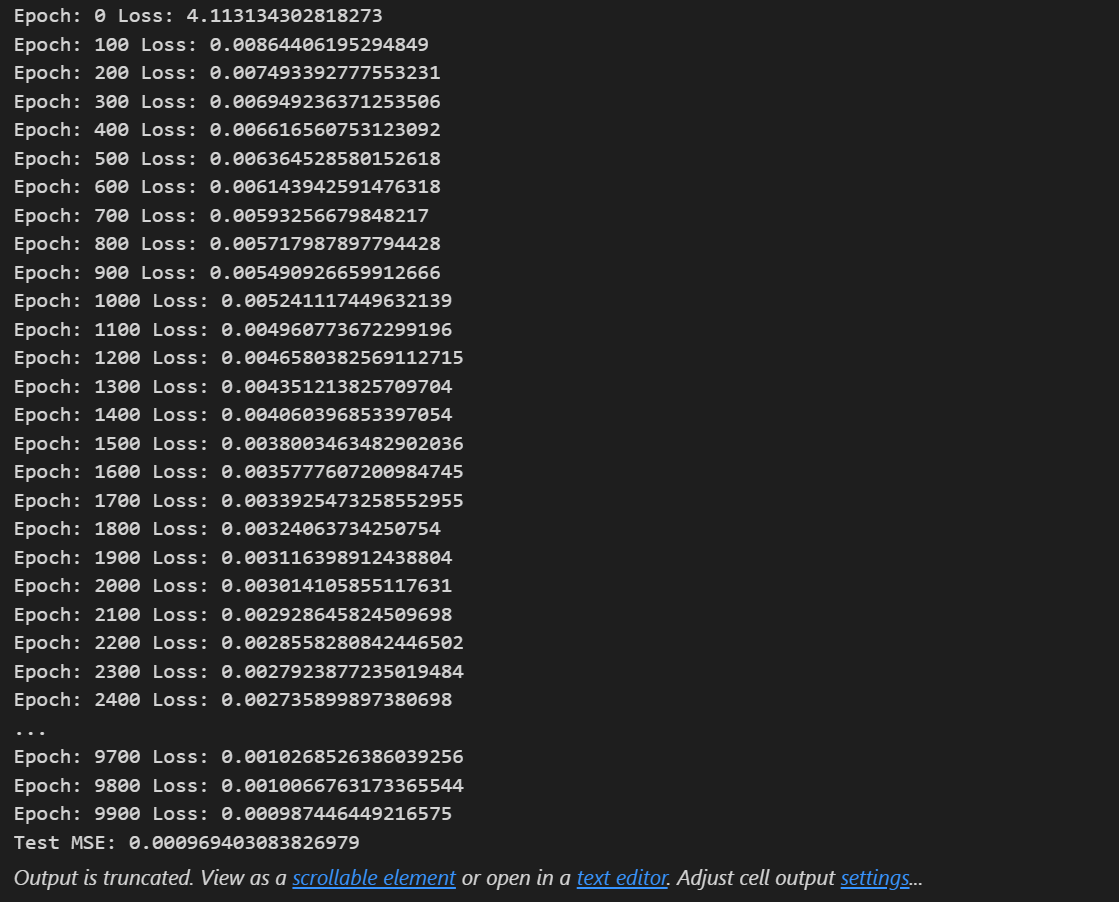
np.mean(dz, axis=0, keepdims=True)计算了所有样本上的平均梯度，

同理，除以样本数量进行归一化，计算平均梯度

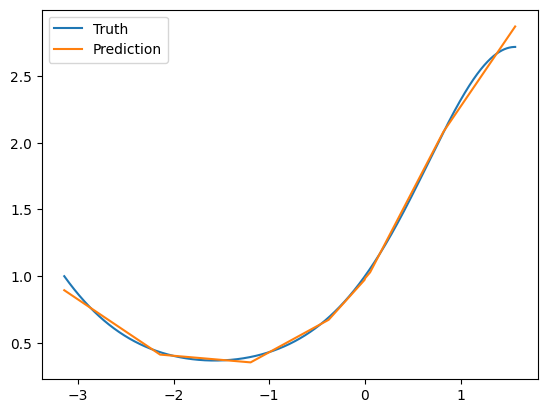
其中，dz是对应的层结果与真实值误差

dz2通过np.dot(dz2, w2.T) \* relu\_derivative(z1)计算得出。

# 拟合效果



每100代的损失情况



与函数拟合情况（蓝线为函数图像，橙线为预测）