

# 函数拟合

## 1. 函数定义

本实验的目标是使用两层 ReLU 神经网络拟合一个非线性函数。选择的函数为： $f(x)=x^3$ ，这是一个简单的多项式函数，适合验证神经网络对非线性函数的拟合能力。

## 2. 数据采集

为了训练和测试神经网络，我们生成了以下数据：

数据范围：从 0 到  $5\pi$ ，步长为 0.001，共生成 15708 个数据点。

数据集划分：

训练集：80% 的数据（12566 个样本）。

验证集：10% 的数据（1571 个样本）。

测试集：10% 的数据（1571 个样本）。

数据生成代码：

```
x = np.arange(0, 5 * np.pi, 0.001)
y = [targetFunc(i) for i in x]
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2,
random_state=1)
X_train, X_val, y_train, y_val = train_test_split(X_train, y_train,
test_size=0.125, random_state=1)
```

## 3. 模型描述

我们使用了一个两层的全连接神经网络（MLP），具体结构如下：

输入层：1 个神经元（输入  $x$ ）。

隐藏层：10 个神经元，使用 ReLU 激活函数。

输出层：1 个神经元（输出预测的  $y$ ）。

损失函数：L1 损失（MAE）。

优化器：Adam 优化器，学习率为 0.001。

训练轮数：100 轮。

模型的定义代码如下：

```
class MLP(nn.Module):
```

```

def __init__(self, in_features=1, out_features=1):
    super().__init__()
    self.FC1 = nn.Linear(in_features=in_features, out_features=10)
    self.relu = nn.ReLU()
    self.FC2 = nn.Linear(in_features=10,
out_features=out_features)

def forward(self, x):
    x = self.FC1(x)
    x = self.relu(x)
    outputs = self.FC2(x)
    return outputs

```

## 4. 拟合效果

### 4.1 训练过程

在训练过程中，我们记录了每一轮的训练损失和验证损失。以下是训练过程中的损失变化：

验证损失：从初始的 944.25 下降到 166.88。

训练损失和验证损失的下降趋势表明，模型能够有效地学习目标函数的特征。

```

0%|
poch0 Loss960.0045166015625
验证集损失: 944.250732421875
10%|
poch10 Loss532.6106567382812
验证集损失: 522.1928100585938
20%|
poch20 Loss469.5384826660156
验证集损失: 466.757080078125
30%|
poch30 Loss434.7850646972656
验证集损失: 432.2745361328125
40%|

```

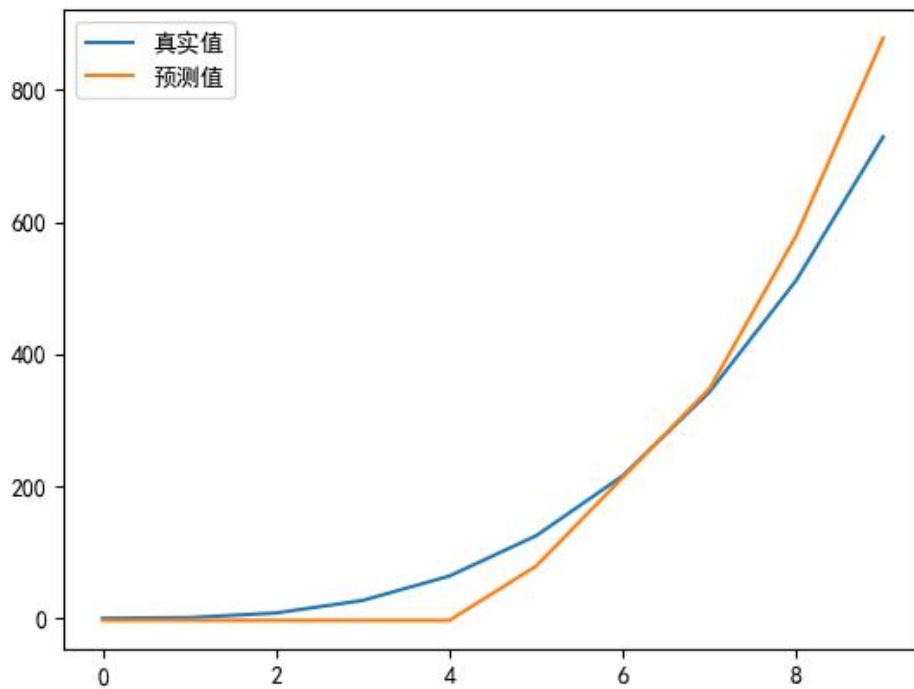
验证集损失: 299.0916442871094		
70%		70/100 [00:12<00:06, 5.00it/s]E
poch70 Loss256.0261535644531		
验证集损失: 250.64483642578125		
80%		80/100 [00:14<00:03, 5.36it/s]E
poch80 Loss211.21815490722656		
验证集损失: 205.54666137695312		
90%		90/100 [00:16<00:01, 5.24it/s]E
poch90 Loss172.41461181640625		
验证集损失: 166.88546752929688		
100%		100/100 [00:18<00:00, 5.54it/s]

## 4.2 测试结果

在测试集上，模型的性能如下：

测试损失：38.64。

拟合效果图：



从图中可以看出，模型的预测值与真实值非常接近，表明神经网络能够很好地拟合目标函数。

## 5. 总结

本实验通过构建一个两层的 ReLU 神经网络，成功拟合了非线性函数  $f(x)=x^3$ 。实验结果表明：神经网络能够有效地学习非线性函数的特征。通过合理的数据划

分和模型训练，可以避免过拟合问题。模型的预测值与真实值非常接近，验证了神经网络在函数拟合任务中的强大能力。