

# 同濟大學

TONGJI UNIVERSITY

## 函数拟合

深度学习第 2 次平时作业 2.2

授课老师 程大伟

专 业 数据科学和大数据技术

学生姓名 涂中浩

学 号 2251320

## 1 拟合函数和模型定义

选择正弦函数进行拟合：

$$f(x) = \sin(x)$$

定义模型为有两个全连接层的神经网络

两个隐藏层大小分别为 64、32

```
def forward(self, x):
    """前向传播"""
    self.z1 = np.dot(x, self.W1) + self.b1
    self.a1 = self.relu(self.z1)
    self.z2 = np.dot(self.a1, self.W2) + self.b2
    self.a2 = self.relu(self.z2)
    self.z3 = np.dot(self.a2, self.W3) + self.b3
    return self.z3
```

## 2 数据采集

编写 generate\_data 函数调用函数采集数据

```
def generate_data(domain, noise_level=0.0, num_train=1000, num_test=200):
    """
    生成训练数据和测试数据。

    :param domain: 定义域, 即 x 的范围, 元组类型 (x_min, x_max)
    :param noise_level: 噪声强度, 默认为 0, 即不添加噪声
    :param num_train: 训练集的样本数
    :param num_test: 测试集的样本数
    :return: 返回训练数据和测试数据
    """
    # 创建 函数 类的实例
    func = fun(noise_level=noise_level)

    # 生成训练集
    x_train = np.random.uniform(domain[0], domain[1], num_train)
    y_train = func(x_train)

    # 生成测试集
    x_test = np.random.uniform(domain[0], domain[1], num_test)
    y_test = func(x_test)

    return (x_train, y_train), (x_test, y_test)
```

采用随机采样的方式获取数据

### 3 拟合效果

模型拟合效果如下

```
Epoch 0, Loss (RMSE): 0.6830415783686106
Epoch 100, Loss (RMSE): 0.6629739890664503
Epoch 200, Loss (RMSE): 0.6629739590407072
Epoch 300, Loss (RMSE): 0.6629739377333441
Epoch 400, Loss (RMSE): 0.6629739164995102
Epoch 500, Loss (RMSE): 0.6629738952660054
Epoch 600, Loss (RMSE): 0.6629738739739166
Epoch 700, Loss (RMSE): 0.6629738525789425
Epoch 800, Loss (RMSE): 0.6629738310390203
Epoch 900, Loss (RMSE): 0.662973809330617
Training Loss: 0.4395345545677529
Test Loss: 0.5028833938998686
```

再训练过程中在前几个 Epoch（例如，Epoch 0 到 Epoch 100）损失（RMSE）有所下降，但从 Epoch 100 开始，训练损失几乎没有明显变化，基本保持在一个较低的值。这意味着模型可能已经在训练数据上收敛。

测试损失为 0.5029，略高于训练损失。模型可能会存在一定程度的过拟合。

RMSE 值为 0.44 左右，表明模型能够相对准确地拟合数据，但仍有改进的空间。