

| 神经网络训练模型

| 1. 实验目的

为了验证模型的健康与否

| 2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]；Visual Studio Code。
- 硬件要求：笔记本/台式电脑1台^[2]。

| 3. 实验地址

例程目录：[\[安装目录\]\RflySimAPIs\7.RflySimPHM\0.ApiExps\e13_model_train](#)

[Model-training.ipynb](#)：神经网络训练模型

| 4. 实验内容或步骤

| 4.1 步骤1：打开文件

通过Visual Studio Code将Model-training.ipynb文件打开。

```
# GPU训练
import os
os.environ["CUDA_VISIBLE_DEVICES"] = "-1"

import numpy as np
import tensorflow as tf
from tensorflow import keras
from keras import layers, Sequential
import pandas as pd

#搭建模型
model = Sequential([
    layers.Dense(56,activation = 'relu'),
    layers.Dense(32,activation = 'relu'),
    layers.Dense(2,activation = 'softmax')
])

#定义输入数据维度
model.build(input_shape = (None,48))
model.summary()
```

4.2 步骤2：运行文件

对Model-training.ipynb文件进行逐步运行(shift + enter)，运行后可以得到最终的数据集文件，生成的文件会置于文件夹中的data文件夹中。

5. 关键知识点

5.1 关键知识点1：神经网络结构

输入层：data_process步骤得到的数据集，输入数据有110维；

第一层隐藏层：56维，激活函数为relu；

第二层隐藏层：32维，激活函数为relu；

输出层为2,0或1,0为故障，1为健康，激活函数为Softmax；

完成上文所述定义输入数据维度操作之后，需要为所建立的双层隐性神经网络模型定义编译器以及优化函数，具体方式可通过Keras提供的`losses.SparseCategoricalCrossentropy()`函数以及`optimizer.RMSprop()`函数实现完成，具体情况如下文代码。

```
1. model.compile(  
2. loss = keras.losses.SparseCategoricalCrossentropy(from_logits=True),  
3. optimizer = keras.optimizers.RMSprop(),  
4. metrics = ["accuracy"]  
5. )
```

6. 参考资料

1. [RflySim官方文档](#)
2. [Keras官方文档](#)
3. [神经网络基础教程](#)

7. 常见问题

Q1: 运行notebook时遇到依赖库缺失的问题怎么办?

A1: 请确保已安装所有必要的Python库，如keras、numpy、pandas等。可以通过pip install命令安装缺失的库。

Q2: 训练过程中出现内存不足错误如何解决?

A2: 这可能是由于数据集过大导致的。可以尝试减小批量大小(batch size)或者清理系统内存后再运行。

Q3: 模型准确率较低应该如何优化?

A3: 可以尝试调整网络结构、增加训练轮次、使用不同的优化器或调整学习率等方式来改善模型性能。

1. <https://rflysim.com/> ↩

2. 推荐配置请见: <https://rflysim.com/> ↩