

实验一

实验要求

使用 pytorch 或者 tensorflow 手写一个前馈神经网络，用于近似函数：

$$y = \sin(x) + \cos(x) + \sin(x) \cos(x), \quad x \in [0, 2\pi)$$

并研究网络深度、学习率、网络宽度、激活函数对模型性能的影响。

实验步骤

- 网络框架**：要求选择 pytorch 或 tensorflow 其中之一，依据官方网站的指引安装包。若你需要使用 GPU，可能还需安装 CUDA 驱动。本次实验仅利用 CPU 也可以完成，但仍强烈推荐大家安装 GPU 版本，以满足后续实验需求。
- 数据生成**：本次实验的数据集仅需使用程序自动生成，即在 $[0, 2\pi)$ 范围内随机 sample 样本作为 x 值，并计算 $y = \sin(x) + \cos(x) + \sin(x) \cos(x)$ 作为 y 值。要求生成三个**互不相交**的数据集分别作为**训练集、验证集、测试集**。训练只能在训练集上完成，实验调参只能在验证集上完成。
- 模型搭建**：采用 pytorch 或 tensorflow 所封装的 module 编写模型，例如 `torch.nn.Linear()`, `torch.nn.ReLU()` 等，无需手动完成底层 forward、backward 过程。
- 模型训练**：将生成的训练集输入搭建好的模型进行前向的 loss 计算和反向的梯度传播，从而训练模型，同时也建议使用网络框架封装的 optimizer 完成参数更新过程。训练过程中记录模型在训练集和验证集上的损失，并绘图可视化。
- 调参分析**：将训练好的模型在验证集上进行测试，以 Mean Square Error(MSE) 作为网络性能指标。然后，对网络深度、学习率、网络宽度、激活函数等模型超参数进行调整，再重新训练、测试，并分析对模型性能的影响。
- 测试性能**：选择你认为最合适的（例如，在验证集上表现最好的）一组超参数，重新训练模型，并在测试集上测试（注意，这理应是你的实验中**唯一**一次在测试集上的测试），并记录测试的结果（MSE）。

实验提交

本次实验截止日期为 **4 月 2 日 23:59:59**，需提交代码源文件及实验报告到邮箱：proton00@mail.ustc.edu.cn，具体要求如下：

- 全部文件打包在一个压缩包内，压缩包命名为 **【学号】 - 姓名 - exp1.zip**

2. 代码仅包含 .py 文件，请勿包含实验中间结果（例如中间保存的数据集等），如果有多个文件，放在 src/ 文件夹内。
3. 代码中提供一个可以直接运行的并输出结果的 `main.py`，结果包括训练集损失、验证集损失随 epoch 改变的曲线（保存下来）和测试集的 MSE。
4. 代码中提供一个描述所有需依赖包的 `requirements.txt`，手动列入代码中用到的所有非标准库及版本或者使用 `pip freeze > requirements.txt` 命令生成。
5. 实验报告要求 pdf 格式，要求包含姓名、学号。内容包括简要的**实验过程**和**关键代码**展示，对超参数的**实验分析**，最优超参数下的训练集、验证集**损失曲线**以及测试集上的**实验结果**。

参考资料

[Learning PyTorch with Examples — PyTorch Tutorials 2.0.0+cu117 documentation](#)

对 Pytorch 不熟悉的同学，可以看一下 Pytorch 官方的一个样例。这个样例使用三次函数拟合 $\sin(x)$ 函数，我们的实验与这个有点类似。

[GitHub - hehaha68/USTC_2022Spring_Introduction-to-Deep-Learning: 中科大2022春《深度学习导论》课程资源](#)

这是往年深度学习导论课程实验代码，同学们可以参考一下。（这里面没有实验一的代码）