

Lesson 12.深度学习基础网络手动搭建与快速实现

一、本节课程介绍

通过此前几节课的学习，我们已经基本掌握了几种基础神经网络的核心原理和实践方法，也就是线性回归、逻辑回归和softmax回归的基本原理和PyTorch中调库实现的方法。同时，我们也了解了神经网络模型的诸多基本概念和神经网络模型构成的基本结构（点线结构），而对于神经网络来说，再复杂的模型都是由这些基础模型构成的，再复杂的模型其基本原理也是和基础模型相同的。

然而，在这个过程中我们发现，尽管深度学习也属于机器学习范畴，基本的建模理念和机器学习类似，比如需要确定目标函数和损失函数、找到合适的优化算法对参数进行求解等等，但实际落地的操作层面、也就是利用PyTorch在进行建模的过程却和经典机器学习有很大的差别，或者说和最大的机器学习库Scikit-Learn中定义的机器学习建模方法有较大的差别。比如说，在数据读取过程中，PyTorch需要将数据先封装在一个Dataset的子类里面，然后再用DataLoader进行装载，然后才能带入训练，而sklearn则可以直接读取Pandas中存储的面板数据进行建模；再比如在模型调用的过程，PyTorch需要先创建一个Module的子类去定义模型基本结构，然后才能实例化这个模型进行训练，并且训练过程中优化算法也是某个类的实例化结果，在训练过程中需要实用这个类的诸多方法来将梯度清零或者更新神经元之间连接的权重，相比之下sklearn则简单的多，只需要在实例化模型的过程中定义好超参数的取值，然后实用fit方法进行训练即可。那为何PyTorch的整个实现过程看似会更加复杂？归根结底，还是和深度学习建模的特殊性有关。

类似PyTorch的这种，看似对初学者略显复杂的建模流程，实际上都是为了能够更好的满足深度学习建模的一般情况：针对非结构化数据、在超大规模的数据集上进行模型训练。我们前面提到PyTorch在读取数据的过程中需要使用Dataset和DataLoader数据进行封装和加载，其实是为了能够实现数据的迭代式存储和映射式存储，也就是通过生成数据的生成器或者保存数据的映射关系，来避免数据的重复存储，如进行小批量数据划分时，PyTorch并未真正意义上的把数据进行切分然后单独存储，而是创建了每个“小批”数据和原数据的映射关系（或者说“小批”数据的索引值），然后借助这种映射关系，在实际需要使用这些数据的时候在对其进行提取。这么做的原因，当然也是因为当进行海量数据处理时，划分多个数据集进行额外的存储显然是不合适的。当然PyTorch在设计的时候将很多数据预处理的方法也放到了数据封装和加载的流程中，这些我们将在后面谈到。而在PyTorch的建模过程中，类的频繁使用其实也是为了能够更加灵活的创建不同类型的神经网络模型。

但不管怎样，这样的建模流程还是给很多初学者造成一定的学习难度，然而熟练掌握深度学习建模流程、熟练使用基本函数和类却是后续学习的基础，因此，在正式进入到下个阶段之前，也就是正式进入到深度学习优化算法学习之前，我们需要进行一段时间的强化练习，通过对此前介绍的基础神经网络进行手动建模实现和调库实现，强化代码能力。

此外，在前几节课的学习当中我们也发现，PyTorch作为新兴的深度学习计算框架，在某些功能实现上还显得不够完善，比如此前我们看到的将“概率”结果划为类别判别的过程、准确率计算过程等等，PyTorch中都没有提供原生的函数作为支持，因此我们需要手动编写此类实用函数。外加模型训练过程本身也可以封装在函数内，因此本节我们也将手动编写PyTorch实际应用中的实用函数作为nn.functional的补充。

另外，为了在后续的优化算法部分课程中更好的观察模型不同优化算法能够起到的作用，本节课程还将介绍数据集创建函数、模型可视化工具TensorBoard安装和实用方法；同时，虽然是建模练习，但可能会涉及一定规模的运算，因此我们还将在本节还将介绍模型的GPU运行方法。

二、课程内容目录

Stage 1.准备工作

Lesson 12.1 深度学习建模实验中数据集创建函数的创建与使用

- 回归类数据集手动创建方法
- 回归类数据集创建函数
- 分类数据集手动创建方法
- 分类数据集创建函数
- 批量数据切分函数
- Python中模块的编写与保存

Lesson 12.2 PyTorch深度学习建模可视化工具TensorBoard的安装与使用

- TensorBoard的安装
- SummaryWriter类的使用方法

Stage 2.深度学习基础模型建模实验

Lesson 12.3 线性回归建模实验

- 深度学习四步建模流程
- PyTorch中可微张量的in-place operation问题解决方法
- 线性回归的手动实现
- 线性回归的快速实现

Lesson 12.4 逻辑回归建模实验

- 逻辑回归手动实现
- 逻辑回归快速实现
- 模型调试入门

Lesson 12.5 softmax回归建模实验

- softmax回归与max回归
- softmax回归手动实现
- 模型稳定性测试
- softmax回归快速实现
- PyTorch深度学习模型的GPU计算方法