

Project1

声明

- 开发语言不限，推荐Python。可以使用一些预处理和方便数学计算的Python库，例如 numpy。**第一部分不能使用Pytorch等深度学习框架。**
- 对于手写汉字分类的任务，目录中提供训练（train）集，**测试（test）集将在面试的时候给出。**
- 出现抄袭现象(包括祖传代码)，抄袭双方均按零分计，面试时会对代码提问。
- 请严格按照Deadline 提交，延迟一天扣10分，扣完为止。
- 更多问题可在课程群以及助教个人微信进行提问。

一、第一部分：反向传播算法（40%）

Deadline: 2023/3/30/23:59:59

任务：反向传播算法实现下列任务：

- 回归：拟合函数

$$y = \sin(x), x \in [-\pi, \pi]$$

- 分类：对12个手写汉字进行分类。

要求：

- （50分）设计实现可伸缩易调整的网络结构，例如可以灵活设置层数、神经元个数、学习率等。
- （20分）在回归任务和分类任务上精度达到规定要求，其中正弦函数数据样例请自行生成，要求在区间内随机采样，**平均误差小于0.01**。手写汉字的训练数据集从 elearning 中下载，精度分数由面试时精度排名得出。
- （30分）编写实验文档，可以包括但不限于：代码基本架构；不同网络结构、网络参数的实验比较；对反向传播算法的理解等。文档要求工整、详实、美观，并导出为 PDF 格式。

建议：

- 回归和分类的损失函数有所不同，需要注意，并考虑其内在原因。
- 一开始将各个权重和学习率调小，bias调成负数，能比较好地达到收敛。

提交：

将所有代码、文档放在 学号-姓名 文件夹下，打包上传到elearning上。

二、第二部分：卷积神经网络（40%）

Deadline: 2023/4/6/23:59:59

任务：使用卷积神经网络实现12个手写汉字分类

要求：

- （50分）了解卷积神经网络的结构，并尝试实现卷积神经网络，可以使用Pytorch等深度学习框架，但**不能直接调用模型**，并测试手写汉字分类任务。不要求可伸缩易调整，精度分数由面试时的测试集精度排名得出。
- （20分）有能力的同学可以进一步查询相关资料，使用一些方法改进网络，提高精度。精度分数由面试时精度排名得出。
- （30分）编写实验文档，可以包括但不限于：代码基本结构；设计实验改进网络并论证；对网络设计的理解等。文档要求工整、详实、美观，并导出为 PDF 格式。

Bonus：

- （10分）使用一些方法防止过拟合，并理解这些方法，可Google一下。
- （5分）尝试放弃使用Pytorch等深度学习框架，手写CNN，挑战自我：)

提交：

将所有代码、文档放在 学号-姓名 文件夹下，打包上传到elearning上。

三、第三部分：预训练模型（20%）

Deadline: 2023/4/13/23:59:59

任务：使用预训练模型实现MNIST手写数字分类

要求：

- （70分）了解计算机视觉的预训练模型及微调（Pretrain+Fine-tune），**可以直接调用模型**，比如（ResNet、VGG等）并在**MNIST数据集**上完成**手写数字**分类任务，正确率达到97.50%即可满分。
- （30分）编写实验文档，可以包括但不限于：你选择的预训练模型的基本结构；你对该预训练模型以及“预训练+微调”的理解等。文档要求工整、详实、美观，并导出为 PDF 格式。

建议：

- 可以安装torchvision框架、huggingface框架等，直接调用前人训好的预训练模型。安装方法：pip install XXX -i <https://pypi.mirrors.ustc.edu.cn/simple>
- MNIST数据集也可以直接通过torchvision框架下载

提交：

将所有代码、文档放在 学号-姓名 文件夹下，打包上传到elearning上。

Tips

- 相比于正确率，此Project更看重实验的过程，所以请详细编写实验文档。
- 面试时，会对BP、CNN的基础知识进行提问。**如果想获得Bonus，则会对Bonus部分重点提问，所以请务必理解原理。**

- 再次强调，第一部分不能使用Pytorch等深度学习框架。第三部分可能要用到的Python库：torchvision、transformers等
- 请尽快开始该Project，因为训练神经网络模型需要花费大量时间。