# 《人工智能导论》大作业

任务名称： 带 OOD 检测的 Mnist 分类器

完成组号：

小组人员： 王骋昊、刘李宇轩、马天翊、彭行健、王柯奔

完成时间： 2023/6/16

## 1．任务目标

基于 Mnist 数据集和非数字图像数据集，构建一个分类模型。该模型可以对 mnist 数据集的图像正确识别代表的数字，同时对于非数字的图像，识别为 OOD 类。要求：

- ✓ 模型是 11 个分类（0-9 代表数字的十个分类和一个 OOD 类）；
- ✓ 在 cpu 上有合理的运行时间。

## 2．具体内容

### （1）实施方案

在设计方案时，由于 mnist 数据集中只含有 0-9 这 10 种数字的标签，因此仅使用 mnist 数据集无法训练出可以识别非数字图像的模型。我们想到两种解决方案：第一种是通过训练两个模型，第一个模型用于判断输入图像是否为数字，若是则调用第二个模型具体判断其所代表的数字，否则将其识别为 OOD 类；第二种方法是将 mnist 数据集和其他非数字图像数据集拼接在一起，直接在一个模型中采用 11 个分类。最终我们选择的是第二个方案，通过 keras 神经网络库搭建神经网络，选取 fashion_mnist 数据集作为 OOD 数据集，将其图片内容和标签相应地拼接起来作为最终的训练集，

### （2）核心代码分析

**CNN 网络生成：**

**model = Sequential()** #利用序贯模型建立模型

**model.add(Conv2D(32,kernel_size=(3,3),activation='relu',input_shape=(28, 28, 1)))** #添加 Conv2D 卷积层，使用 32 过滤器，卷积核为 3*3 大小，激活函数为 relu，输入维度为 28*28

**model.add(Conv2D(64, (3, 3), activation='relu'))** #第二层卷积层使用 64 过滤器，仍为 3*3 卷积核

**model.add(MaxPooling2D(pool_size=(2, 2)))** #2D 最大池化层，池化窗口大小为 2*2

**model.add(Dropout(0.25))** #添加 Dropout 层，神经网络将输出反馈到后续层，设置保留概率为 0.25

**model.add(Flatten())** #将多维输入一维化

**model.add(Dense(128, activation='relu'))** #添加全连接层 1，输出维度为 128，激活函数设为 relu

**model.add(Dropout(0.5))** #保留概率为 0.5

**model.add(Dense(11, activation='softmax'))** #添加全连接层 2，输出维度 11，对应 11 种分类

```
Epoch 7/10
938/938 [==============================] - 69s 74ms/step - loss: 0.0210 - accuracy: 0.9934
Epoch 8/10
938/938 [==============================] - 69s 73ms/step - loss: 0.0186 - accuracy: 0.9940
Epoch 9/10
938/938 [==============================] - 69s 73ms/step - loss: 0.0172 - accuracy: 0.9947
Epoch 10/10
938/938 [==============================] - 69s 74ms/step - loss: 0.0155 - accuracy: 0.9949
```

**classify：**

**len = imgs.shape[0]** #读取输入 tensor 张量的 batch 大小

**preds = torch.empty(len, 1)** #创建返回的 n 维 tensor

**for i in range(len):**

**pre = imgs[i].reshape(1, 28, 28, 1)** #转换输入 imgs 格式

**res = self.mnist_load.predict(pre)** #利用 mnist_load 模型预测图片为各个类的概率

**preds[i] = np.argmax(res)　#设置返回值为概率最大对应的值**

**return preds**

```
1/1 [==============================] - 0s 77ms/step
预测的数字为：  6
实际数字为：  6
```

```
1/1 [==============================] - 0s 15ms/step
1/1 [==============================] - 0s 15ms/step
tensor([5., 0., 4., 1., 9., 2., 1., 3., 1., 4., 3., 5., 3., 6., 1., 7.])
```

## 3．工作总结

### （1）收获、心得

王骋昊：本次大作业完成过程中，我们在理解作业要求方面花了比较长的时间，由于之前对神经网络的了解不多，因此在理解过程中我通过网上查找相关知识初步了解模型的构建及训练过程。我主要提供了构建 11 个分类的思路并初步建立了模型，最后也设计实例对我们完成的 OodCl 类进行检验。

刘李宇轩：在本次大作业的完成过程中，我与其他组员一同努力，收获颇多。在开始阶段，主要困扰我的是对于作业要求，已有代码等的解释。我通过详细分析，明白了我们的目的主要是实现接口 oodcls.py 中加载数据集，加载模型并调整参数，classify 函数转化三维 tensor 的图像为一维 tensor 的数字的功能，以及对实现识别功能的模型进行编写与训练。

这之后，我负责模型优化，在通过查阅相关资料了解了 dropout，L2 正则化等提高识别精确率的方法之后，对 CNN 神经卷积网络进行了改良，将改良后的模型训练使用后，原来笔画痕迹位置不同导致识别结果有误，OOD 识别率差等问题好了一些。可以说我对于 CNN 的

理解与改进从懵懂无知到了一知半解的地步，也算是有进步了。

马天翊：通过这一次的人工智能大作业，我了解了带 OOD 检测的 Mnist 分类器的工作原理。在这一次大作业的完成中，最令我印象深刻的就是我们对于实现这个模型的方法探索。我们试过一些方法，比如计算相关概率等，但是最后都是因为方法不太合适而舍弃，最终我们经过讨论得出了已知的最好的方法，就是我们展示在大作业中的办法。

我觉得这一次大作业不但让我们学到了知识，更让我们有了一次探索的机会，一次通过自己的研究解决难题的机会，这对于我们以后的学习，乃至科研工作，都是有很多锻炼的。

最后，感谢老师和助教对于这一次大作业的指导！

**（2）遇到问题及解决思路**

我们遇到的主要问题是如何将非数字图像识别为 OOD 类，最初我们尝试使用两个模型分别判断 OOD 和具体数字，但这一方案和作业要求中的 11 个分类相冲突，于是我们选择将 OOD 数据集和 mnist 数据集拼接在一起，并设置 OOD 数据集图片对应的标签为 10 以此实现分类。

## 4．课程建议

课程可以再安排一些有关神经网络方面的内容理解，大作业布置时间也可以再提前一点，从零理解神经网络仍然有点吃力。