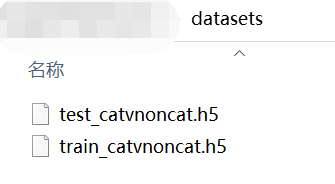
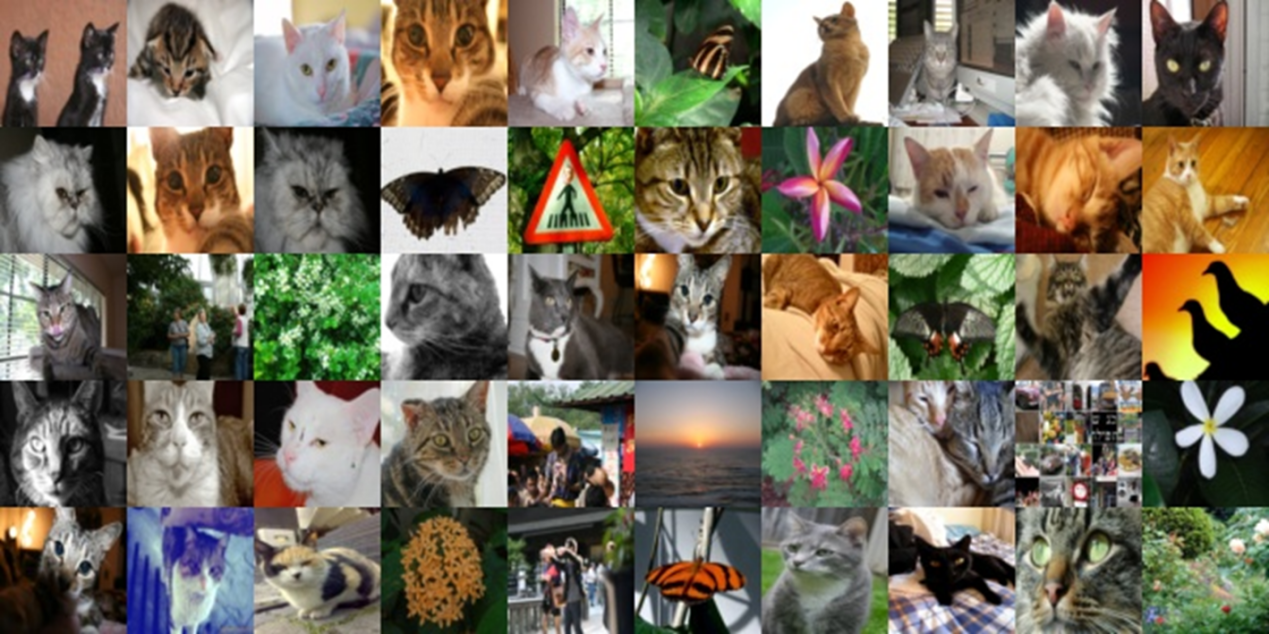
# 实验五 课内竞赛-图像分类

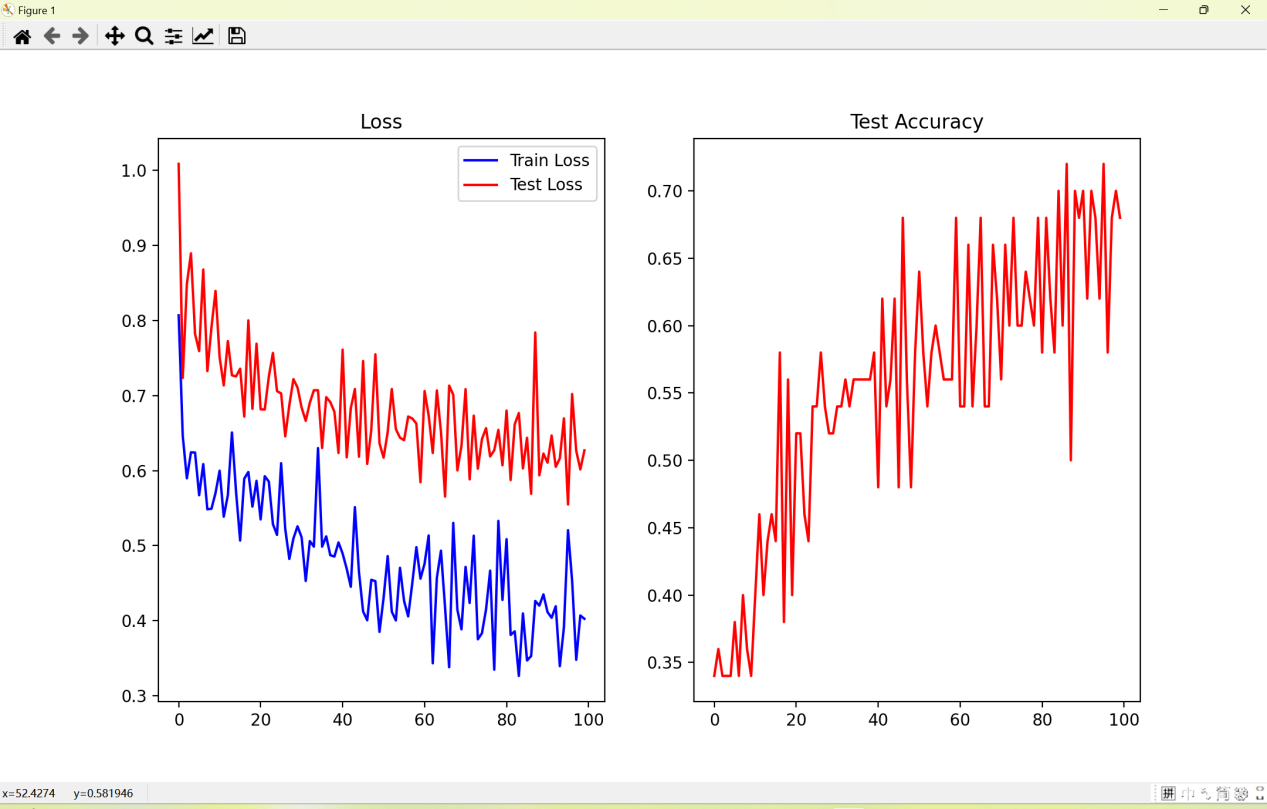
1. 实验目的
   1. 掌握使用深度神经网络解决图像分类问题的方法和流程。
   2. 掌握数据标注、增强的方法，了解并初步掌握网络优化算法和方法。
   3. 注意：本实验为课内竞赛，请对使用优化算法尽可能提升模型性能。
2. 实验内容
   1. 猫识别数据集：带数据标注的训练集和测试集以字典的形式被保存为h5格式的文件（.h5格式是使用h5py库生产数据文件类型）。其中：
      1. train\_set\_x是一个（209,64,64,3）的四维数组，包含了209张RGB格式的训练图片。其中，正样本72，负样本137
      2. train\_set\_y是一个（209，）的一维数组，依次对应于训练图片的类别标签（0 | 1）
      3. test\_set\_x是一个（50,64,64,3）的四维数组，包含了50张RGB格式的测试图片
      4. test\_set\_y是一个（50，）的一维数组，依次对应于测试图片的类别标签（0 | 1）
      5. list\_classes是一个（2，）的一维数组，保存了类别名称。



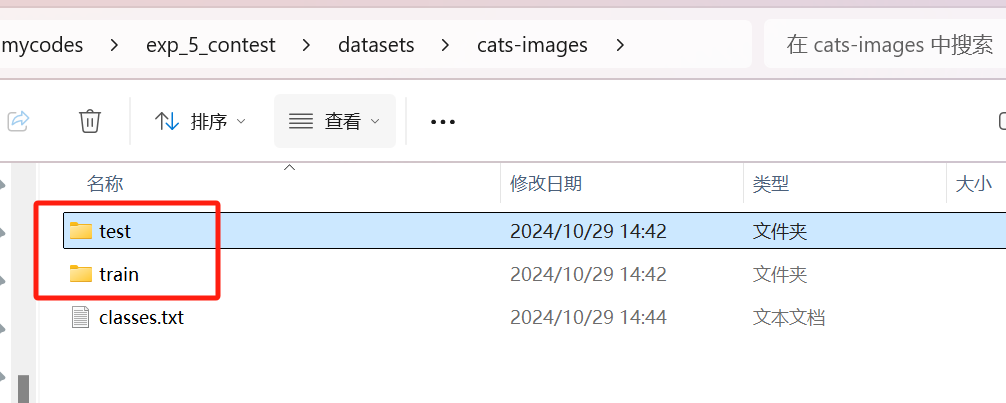


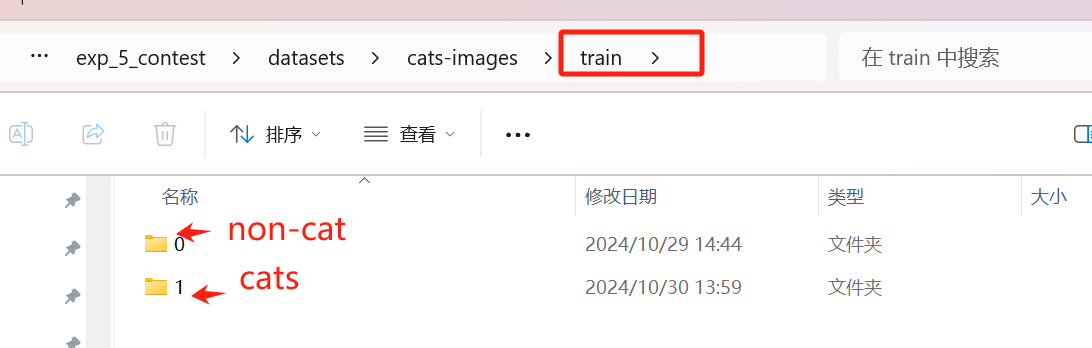


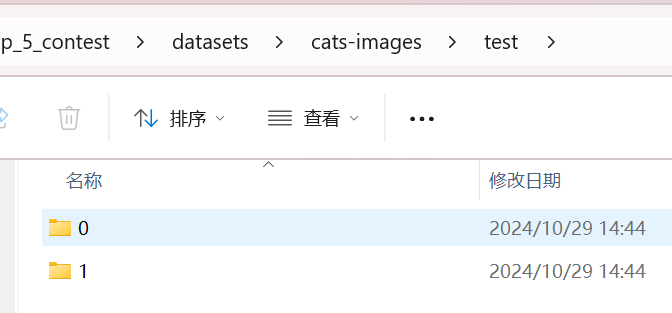
* 1. 数据导入：定义load\_data\_from\_h5()函数，导入h5数据集。代码见utils.py
  2. 定义linear\_regression\_model()函数，实现模型。代码见models.py
  3. 定义train()函数，定义训练验证过程。代码见train.py



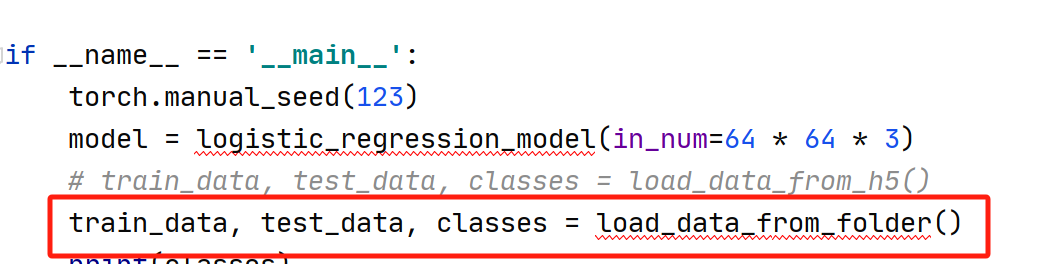
* 1. 可以定义save\_images()函数将h5数据文件另存为图像文件，并按类别放置到不同的子文件夹中。代码见utils.py



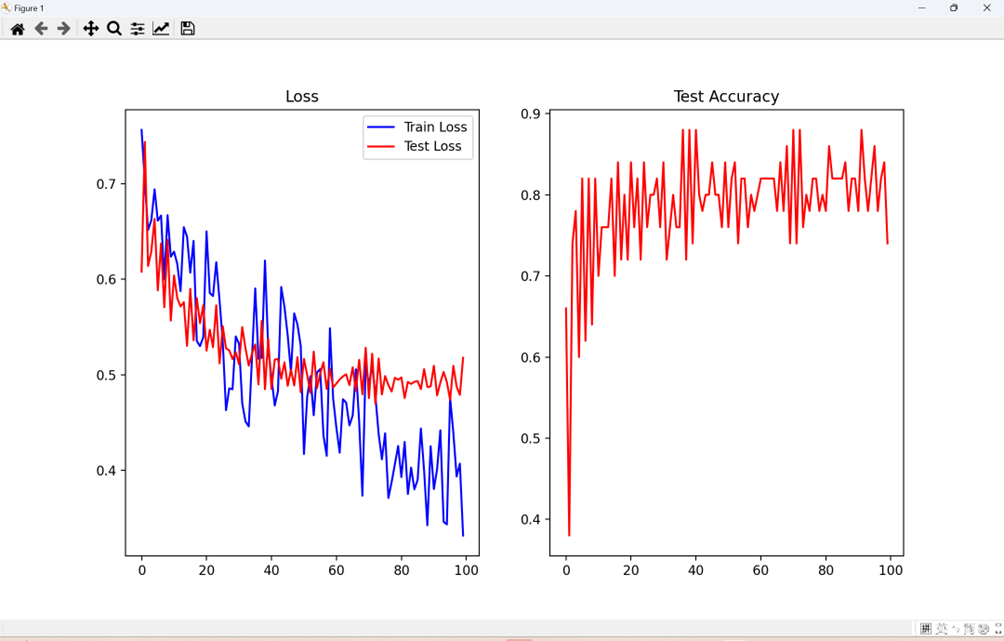




* 1. 定义load\_data\_from\_folder，从文件夹中导入图像文件。代码见utils.py
  2. 修改train.py中的执行代码



* 1. 数据增强，增加正样本的数量。例程见utils.py中data\_augument()函数



* 1. 保存和导入模型参数

通过torch.save()和torch.load()函数保存和导入模型或模型参数字典。本次竞赛要求保存整个模型。

关于模型或参数的保存和导入，可参考以下链接<https://zhuanlan.zhihu.com/p/82038049>

* 1. 导入模型和推理。例程见infer.py

1. 课后思考题

如何保存上述程序执行后获得的模型参数，以备日后推理使用？