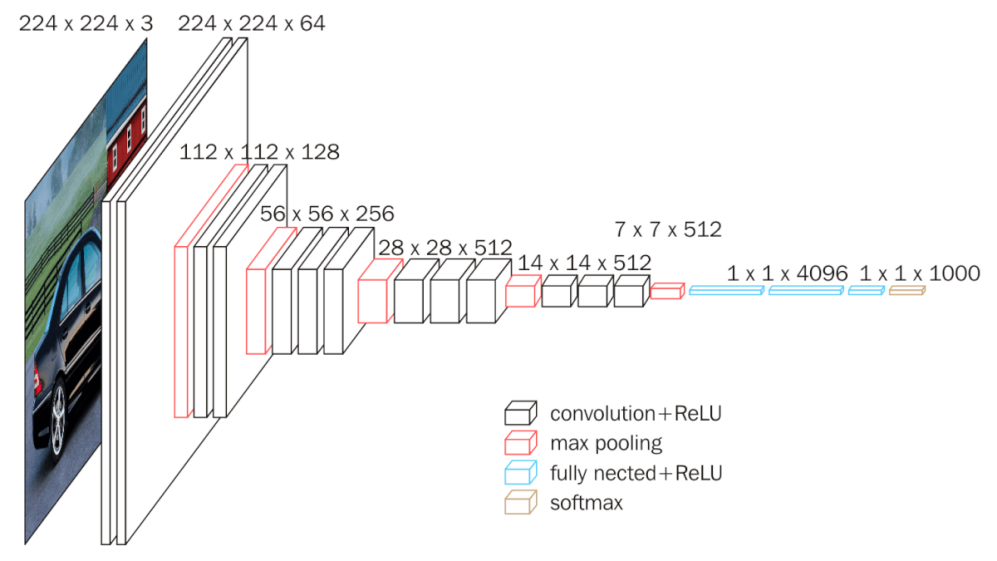
利用VGG16进行迁移学习



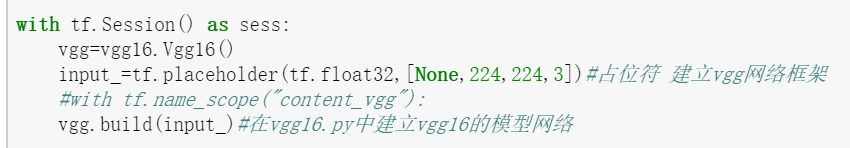
概念理解

## 什么是/为什么要迁移学习？

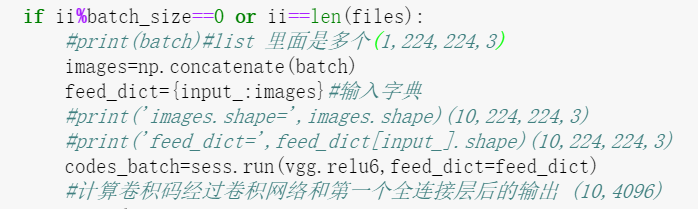
迁移学习(Transfer learning) 顾名思义就是就是把已学训练好的模型参数迁移到新的模型来帮助新模型训练。考虑到大部分数据或任务是存在相关性的，所以通过迁移学习我们可以将已经学到的模型参数（也可理解为模型学到的知识）通过某种方式来分享给新模型从而加快并优化模型的学习效率不用像大多数网络那样从零学习（starting from scratch，tabula rasa）。

由于我们的数据集有限，通常我们的数据集都是中等大小，而且我们的硬件条件有限，很难从头训练一个神经网络，并且将其的权重训练到最优。VGG16.npy是网络大牛们用最好的硬件条件，在大型的数据集下，经过无数次训练得到的权重，这些权重之所以可以被使用，是因为在图像处理中，一些基本的特征都是一致的，卷积层的权重被保留下来，可以提取图片中的一些特征，而针对不同的分类任务，改变全连接层，得到不同的分类，我们就可以实现精度比较高的图像分类识别网络。

调用Vgg16

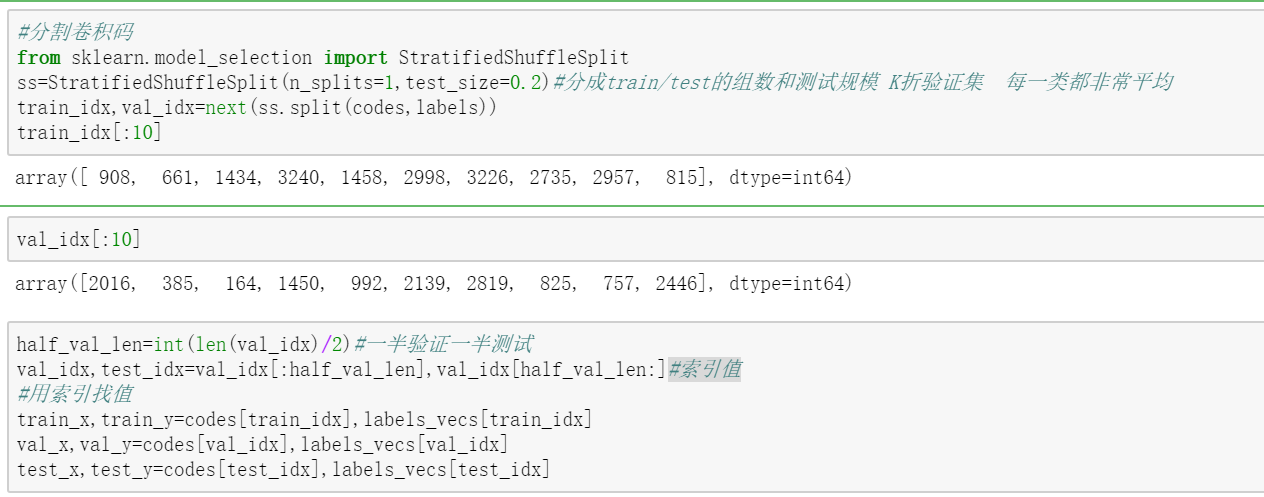


使用卷积神经网络和最后一个全连接层的输出 是卷积码 10\*4096

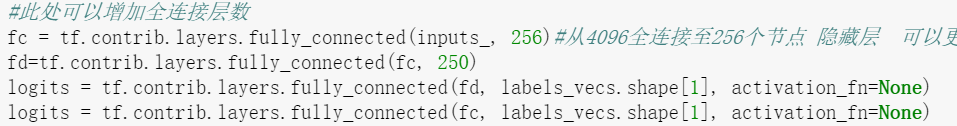


调用VGG16的前6层，去掉后面的全连接层，得到卷积码，将得到的卷积码投入自己设定的全连接层中，

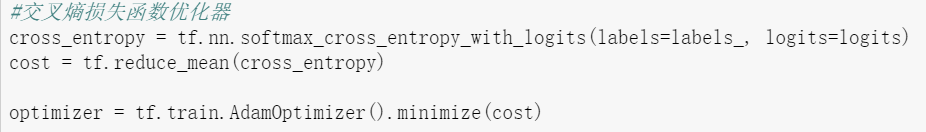
将卷积码当作正常的输入，分成训练集，测试集,验证集，



把分到的三个集投放到新训练模式中，此处的神经网络由自己进行配置，增加全连接或者增加池化层都可以，按照最后的训练效果来选择



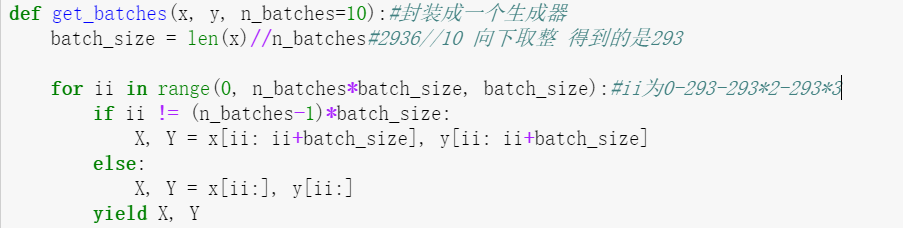
正常配置交叉熵损失函数优化器



生成器

生成器是可以迭代的，但是只可以读取它一次 ，因为它并不把所有的值放在内存中，它是实时地生成数据 ，降低内存占用量 即使是这样，跑的时候内存也爆炸了 最高98%

在下面用for进行取值时才启动，每次接着上次的位置产生数据





运行全连接层计算卷积码数据，结果

