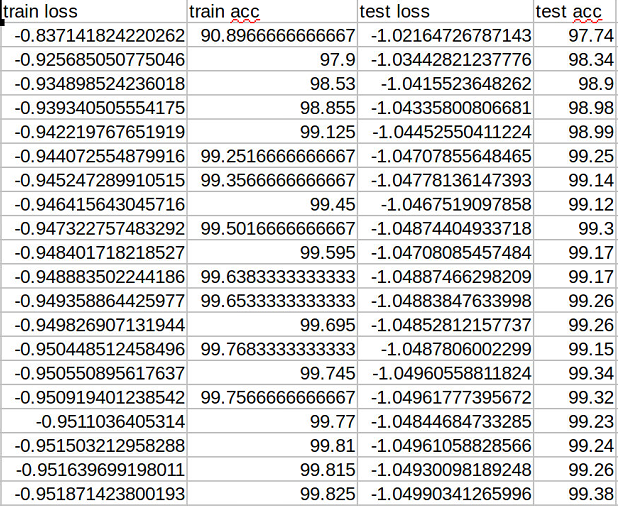
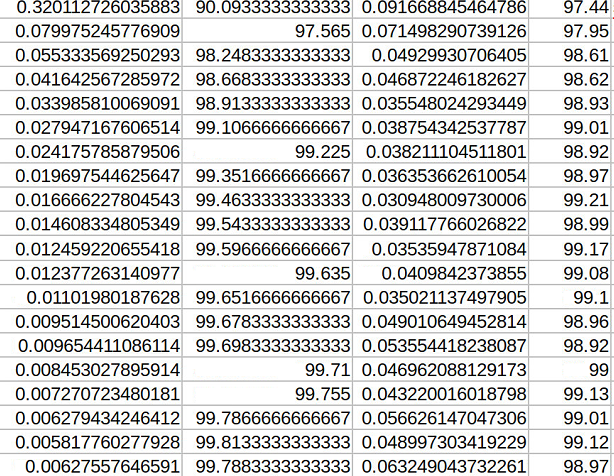
学号： 2019110019 姓名： 白雪涵

Mnist数据集是一个经典的图像分类数据集，共60000张训练集图片和10000张测试集图片。每张图为28\*28的单通道灰度图。

本次复现的模型分别是Efficientnet（参考文献：EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks，代码借鉴[1],[2]），采用GCE作为loss函数的LENET-5，后文简称GCE（采用开源代码，参考文献Improving Adversarial Robustness via Guided Complement Entropy，开源代码来自论文作者github[3]），以及GResnet,(文献：Differentiable Learning-to-Group Channels via Groupable Convolutional Neural Networks，代码参考：[4])。对比模型为resnetV\_1(参考文献：Deep Residual Learning for Image Recognition)。

其中GCE的效果为：

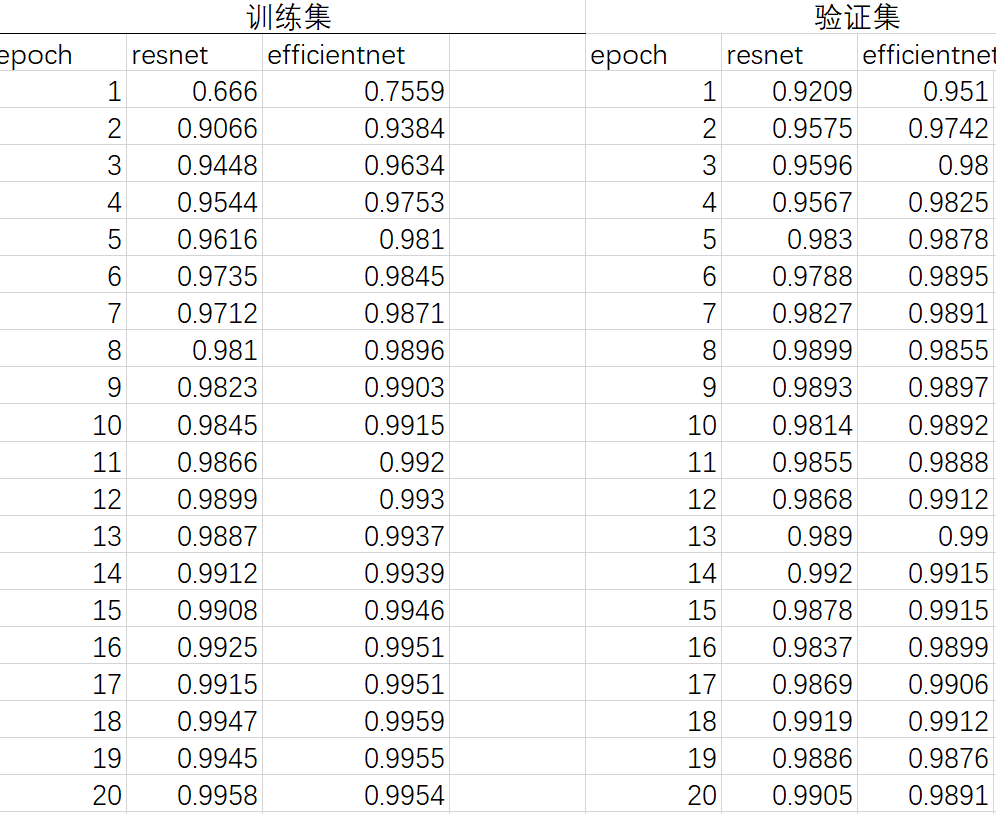
 

Alpha采用0.333，lenet-5 未采用GCE的lenet-5

除了lenet以外的其他模型，由于模型本身为三通道大数据集准备，所以mnist数据集图片较小，且为单通道，所以无法直接使用，因此采用在输入模型前按如下方法处理数据：

1. 通过3通道1\*1卷积，输出为三个通道
2. ZeroPadding，到符合大型模型输入图片大小的最小要求。

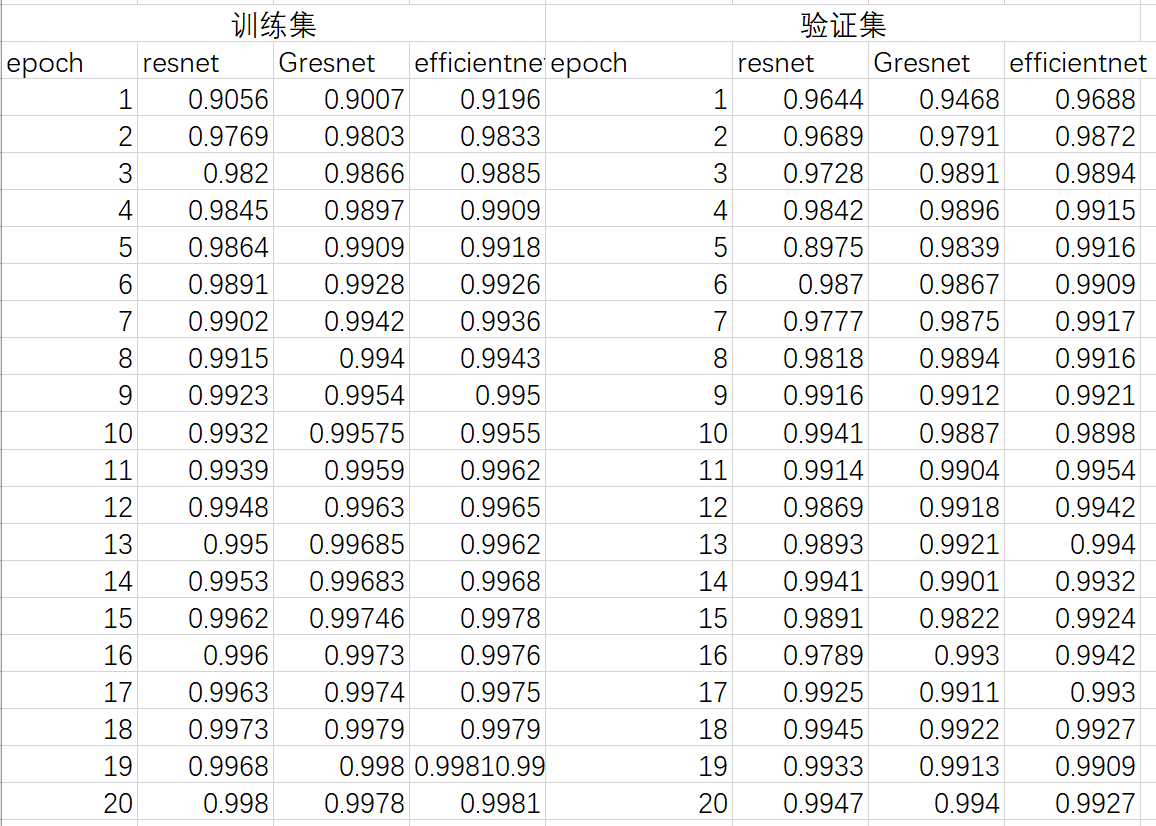
用上述方式实现了、resnet50\_v1和Efficientnet结果如下：(右图四条线分别是从下到上)



可以发现虽然验证集合准确率都在95%以上，甚至可以到达98%-99%，但是这和没采用改进方法的lenet-5差不多，略低于采用GCE的lenet-5，一方面这是由于大的模型中许多超参数没有经过仔细调参，效果较差，另一方面这是由于层数过多，图片太小导致经过多次卷积后，图片宽和高都变得非常小，导致后面的层卷积效果不佳。这对越复杂，泛化性越强的网络影响越大。

为了解决这个问题，采用先对原始图片进行双线性插值的方法，扩大图片大小，然后输入较大模型中。

下面分别是GResnet，Resnet和Efficientnet在经过插值为200\*200的图像之后在经过1\*1卷积核和原模型之后的准确度。



其中Gresnet只有插值之后的图片的运行结果。和之前的情况相比，显然Resnet和efficientnet都有着小幅度的提高，而且更大的图片带来了更快的收敛速度。显然在插值之后的图片只经过一轮训练就基本收敛到90%以上，而小的图片需要大概三到四轮。这说明更大的图片可以提供更多的信息。而且在使用回大图之后，Resnet，Gresnet和efficientnet的验证集准确度都达到或者略微超过采用GCE的lenet-5模型。而这三个模型之间对比发现三个模型效果基本一致，efficientnet的最佳准确率最高，所以稍微好一点。Gresnet相比较于传统的Resnet并没有提高，甚至最佳准确率还稍微低一点。三个模型的收敛速度上，efficientnet最快，Gresnet最慢。可以发现在预测准确率和收敛速度上efficientnet有提高，而Gresnet则没有提高，这可能是因为Gresnet的主要优点是可以综合考虑不同channel之间的信息交互，而mnist是单通道数据，输入模型的三通道是通过卷积生成的，因此影响了Gresnet的效果。不过Gresnet和efficientnet的预测准确度随着训练产生的波动较小，而相比较而言，resnet随着训练继续进行，测试集上的预测准确度产生的波动比较大。

三种模型的参数量分别为：resnet-23555088，Gresnet-66854873，efficientnet-7177792，由此可见，efficientnet用最少的参数实现了最佳效果。除此以外efficientnet论文中提出了多种超参数设置的同类型不同网络，考虑到其他参数版本需要计算量较大，所以采用了基础版本。因此理论上，efficientnet可是通过优化参数设置和网络结构，增加参数量和输入图片大小，获得效果更好的网络。

综上所述，对小图片采用插值生成大图片，可以有效提高大模型的效果。在mnist上，efficientnet具有最少参数，最快收敛速度和最佳预测准确率，resnet其次，而Gresnet由于无法发挥结合不同channnel信息的优点，导致效果相对较差。

GCE-lenet-5模型看起来较为简单，相对其他几个模型效果较差，但这是由于GCE的提出是为了解决对抗攻击的问题。该模型在目前威胁最大，最难解决的PGD攻击下的表现为：

基本模型在PGD攻击下准确率： 0.28%

GCE在PGD攻击下准确率： 5.42%

说明原来的模型基本没有对对抗攻击的任何抵抗能力，而选用GCE可以给模型带来一定的抵抗能力，而且无论在正常预测的准确率和收敛速度上都没有任何负面的影响，因此GCE可以作为一种通用的简单易实现的loss函数使用。