

数据增强 (三) -Batch Augmentation

实验背景

本文为数据增强的第三篇， batch augmentation； 顾名思义， 在一个batch的数据中进行数据增强。

我们知道在深度学习的训练过程中，一般都采用mini-batch方式进行的训练；batch size的大小可能会影响训练的精度，主要如下：

- 越小的batch-size，因为数据与整个数据集分布的个体差异性，引入梯度下降的噪声越大
- 大的batch-size，梯度下降能更反映整个数据集的准确性，但是大的batch，梯度的方差变大，可能降低泛化能力；这个时候可以配合学习率的增大，学习步伐大一点，达到精度的提高。

Batch Augmentation就是在增大batch-size训练时，提供一种简单的增强方式来达到更好精度的一种数据增强策略

参考文献：Augment your batch: better training with larger batches

实验内容

增强策略

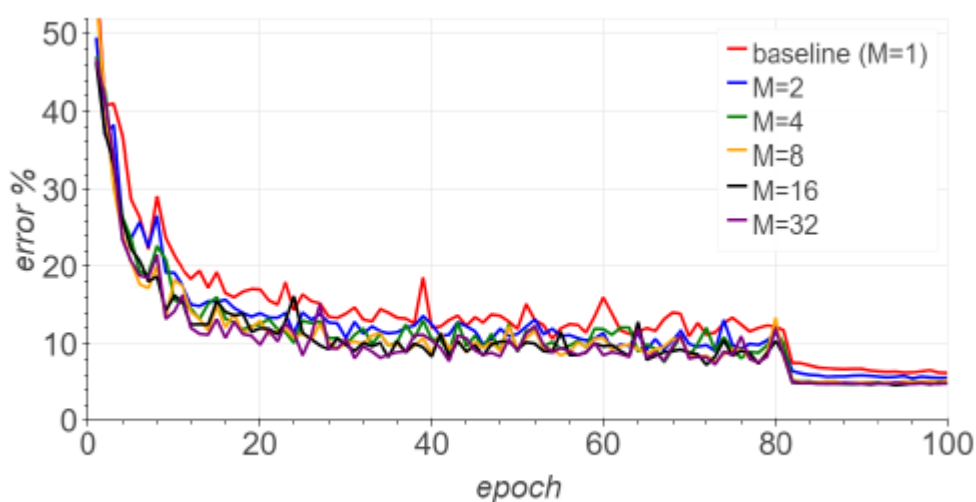
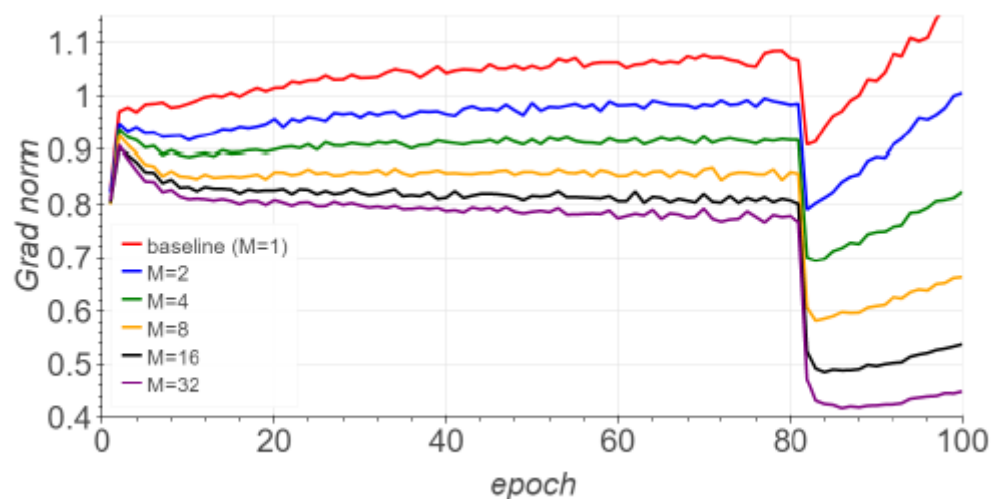
BA (batch augmentation) 的增强策略其实非常的简单；在同一个batch中，对同一个样本进行多次的augmentation (通常也包括没有进行augmentation的原始样本)，即一个batch中一个样本有不同的实例

- 1. 生成原始batch的数据， batch-size=b
- 2. 运用普通augmentation策略，分布作用原始batch的数据，生成m份不同的数据
- 3. 合并m份数据，形成新的(m+1)*b的batch数据，喂给模型进行训练

BA (batch augmentation) 好处：

- 梯度下降的方向会更准确，降低梯度下降的方差
- 梯度下降更关注某一个图像的分类，而不仅仅图像样本本身

直观上，人类也很喜欢通过对比的方式，作出总结分类；BA中同一样本多种不同的副本在同一个batch中学习，更好让模型学习到共同之处。



(a) Validation error

实现细节

按照上面的步骤，实现就很简单啦。可以结合imgaug进行

```
seq = iaa.Sequential([
    iaa.CoarseDropout(0.05, size_percent=(0.02, 0.05)),
], random_order=False)
seq2 = iaa.Sequential([
    iaa.SomeOf((0, 5), [
        iaa.OneOf([
            iaa.GaussianBlur((0, 2.0)),
            iaa.AverageBlur(k=(2, 5)),
            iaa.MedianBlur(k=(3, 7)),
        ]),
    ], random_order=False)
def batch_aug_v1(x, y, aug_list=[seq, seq2]):
    aug_x = None
    aug_y = None
    if aug_x is None:
        aug_x = x
```

```
if aug_y is None:
    aug_y = y

for aug in aug_list:
    x_tmp = aug.augment_images(x)
    aug_x = np.concatenate([x_tmp, aug_x])
    aug_y = np.concatenate([y, aug_y])
return aug_x, aug_y
```

实验结语

本实验介绍了Batch Augmentation的数据增强策略，通过在一个batch的对同一个样本建立多个不同的副本的简单策略，降低梯度下降的方差，提高训练速度和精度。实际场景中，BA某些任务中有一定的效果。增加了一种调参的方式。

希望对大家有帮助。