批归一化:通过

减少内部协变量移位

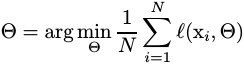
Sergey Ioffe谷歌Inc.， sioffe@google.com

**摘要**

训练深度神经网络是复杂的，因为在训练过程中，随着前一层参数的变化，每层输入的分布也会改变。这降低了训练的速度，因为需要更低的学习速率和仔细的参数初始化，并且使得训练饱和的非线性模型变得非常困难。我们将这种现象称为内部协变量移位，并通过规格化层输入来解决这个问题。我们的方法的优势在于使规范化成为模型体系结构的一部分，并对每个训练小批进行规范化。批处理规范化允许我们使用更高的学习率，并且不太注意初始化。它还充当了一个正规化程序，在某些情况下消除了对Dropout的需要。应用于最先进的图像分类模型，批归一化实现了相同的精度，而训练步骤少了14倍，并显著优于原始模型。使用一批归一化网络集成，我们改进了在ImageNet分类上发表的最佳结果:达到4.9%的前5位验证错误(和4.8%的测试错误)，超过了人工评分的准确性。

**1介绍**

深度学习极大地提高了视觉、语言和许多其他领域的技术水平。随机梯度下降(Stochastic gradient descent, SGD)已被证明是一种有效的训练深度网络的方法，SGD变量如动量(Sutskever et al.， 2013)和Adagrad (Duchi et al.， 2011)已被用来实现最先进的性能。SGD对网络参数Θ进行优化，使损耗最小化



x1...N 是训练数据集。使用SGD，训练一步一步进行，在每一步我们考虑一个小批x1...m 小批量是用来近似的梯度损失函数与参数，通过计算

FORMULA

基督教Szegedy

谷歌公司,szegedy@google.com

使用小批量的示例，而不是一次一个示例，在几个方面都是有帮助的。首先，在小批处理上的损失梯度是对训练集上的梯度的估计，训练集的质量随着批处理规模的增加而提高。其次，由于现代计算平台提供的并行性，批处理计算比单个示例的m次计算要高效得多。

虽然随机梯度简单有效，但它需要仔细调整模型超参数，特别是优化中使用的学习率，以及模型参数的初始值。由于每一层的输入都受到前面所有层的参数的影响，训练变得复杂——因此，网络参数的微小变化会随着网络的加深而放大。

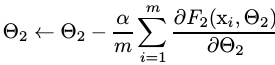
由于各层需要不断地适应新的分布，所以各层输入分布的变化是一个问题。当一个学习系统的输入分布发生变化时，被称为经历协变量转移(Shimodaira, 2000)。这通常通过域自适应来处理(Jiang, 2008)。但是，协变量迁移的概念可以扩展到整个学习系统之外，应用于学习系统的各个部分，如一个子网络或一个层。考虑网络计算

FORMULA

F1 和F2 是任意的转换，参数是Θ吗1,Θ2 都是为了减少损失ℓ。学习Θ2 可以看成输入x = F1(u,Θ1)被送入子网络

FORMULA

例如，梯度下降步骤



(对于批大小m和学习率α)与独立网络F完全等价2 因此，使训练更有效的输入分布属性——例如在训练数据和测试数据之间具有相同的分布——也适用于训练子网络。因此，x的分布在一段时间内保持固定是有利的。然后,Θ2 做

arXiv: 1502.03167 v3 [cs。LG] 2015年3月2日

1

不需要重新调整来补偿x的2分布的变化。

**对**

**减少**

**内部**

**协变量转变**

对子网络的固定输入分配也会对子网络之外的层产生积极的影响。考虑一个具有sigmoid激活函数z = g(Wu + b)的层，其中u是层输入，权值矩阵W和偏置向量b是需要学习的层参数，g(x) = 1+exp(−x)1。随着|x|的增加，g ′x趋于0。这意味着对于x = Wu+b的所有维度，除了绝对值小的维度，向下流向u的梯度将消失，模型将训练缓慢。但是，由于x受到W, b以及下面所有层的参数的影响，在训练过程中改变这些参数很可能会使x的多个维度进入非线性的饱和状态，减慢收敛速度。随着网络深度的增加，这种效应会被放大。在实践中，饱和问题和产生的消失梯度通常是通过使用校正线性单元(Nair & Hinton, 2010) ReLU(x) = max(x, 0)，仔细初始化(Bengio & Glorot, 2010;Saxe等人，2013年)和小学习率。然而，如果我们能保证非线性输入的分布在网络训练时保持更稳定，那么优化器就不太可能陷入饱和状态，训练就会加速。

在训练过程中，我们将深层网络内部节点分布的变化称为内部协变量位移。消除它意味着更快的训练。我们提出了一种新的机制，我们称之为批归一化(Batch Normalization)，它朝着减少内部协变量转移迈出了一步，并在此过程中极大地加快了深度神经网络的训练。它通过一个固定层输入的方法和方差的标准化步骤来完成这一点。批处理归一化也对通过网络的梯度流有有益的影响，因为它减少了梯度对参数规模或其初始值的依赖。这使得我们可以使用更高的学习速率，而不存在分化的风险。此外，batch normalization使模型规范化，减少了Dropout的需求(Srivastava et al.， 2014)。最后，批处理归一化通过防止网络陷入饱和模式，使使用饱和非线性成为可能。

在4.2节中，我们对性能最好的ImageNet分类网络应用批归一化，并表明我们仅使用7%的训练步骤就可以匹配它的性能，并且可以进一步大大超过它的精度。使用一个经过批归一化训练的网络集成，我们获得了前5名的错误率，比ImageNet上最知名的分类结果更好。

我们将内部协变量偏移定义为训练过程中网络参数的变化导致的网络激活分布的变化。为了改进训练，我们试图减少内部协变量的变化。随着训练的进行，通过固定层输入x的分布，我们期望提高训练速度。它早就为人所知(LeCun et al.， 1998b;Wiesler & Ney, 2011)认为，如果输入是白化的，网络训练收敛得更快。，线性变换为零均值和单位方差，并去相关。由于每一层都观察下一层产生的输入，因此每一层的输入都能达到相同的白化效果是有利的。通过白化每一层的输入，我们将朝着实现输入的固定分布迈出一步，这将消除内部协变量移动的不良影响。

我们可以在每个训练步骤或某个间隔时间考虑白化激活，可以直接修改网络，也可以根据网络激活值改变优化算法的参数(Wiesler et al.， 2014;赖科等人，2012年;Povey等人，2014年;德斯贾丁斯& Kavukcuoglu)。然而，如果这些修改散布在优化步骤中，那么梯度下降步骤可能会试图以需要更新归一化的方式更新参数，这就减少了梯度步骤的影响。例如，考虑一个带有输入u的层，该层添加了学习偏差b，并通过减去在训练数据上计算的激活的平均值来规范化结果:xb = x−E[x]其中



E[x] = N1PNi= 1xi。如果一个梯度下降步骤忽略了E[x]对b的依赖关系，那么它

将更新b←b +∆b，其中∆b∝−∂ℓ/∂bx。然后



FORMULA

改变归一化导致没有改变的输出层，因此，也没有损失。随着训练的继续，b将无限增长，而损失保持不变。如果正规化不仅使激活集中，而且使激活规模扩大，这个问题就会变得更糟。我们在最初的实验中观察到这一点，当归一化参数在梯度下降步骤之外计算时，模型就会崩溃。

上述方法的问题是，梯度下降优化没有考虑到发生归一化的事实。为了解决这个问题，我们希望确保，对于任何参数值，网络总是生成具有所需分布的激活。这样做可以使损耗对模型参数的梯度考虑到归一化，以及它对模型参数的依赖Θ。再次让x作为一个层输入，作为a

2

向量，X是这些输入在训练数据集上的集合。然后可以将归一化写成一个变换

这不仅取决于x b= Norm(x，取决于给定的x)训练例子x，而且取决于所有的例子x——如果x是由另一层生成的，每个例子都取决于Θ。对于反向传播，我们需要计算雅可比矩阵

FORMULA

忽视后一项将导致上述爆炸。在这个框架内，白化层输入是昂贵的，因为它需要计算协方差矩阵Cov[x] = Ex∈X [xxT]−E[x]E[x]T及其平方根，以产生白化激活Cov[x]−1/2(x−E[x])，以及这些反向传播变换的导数。这促使我们寻求一种替代方案，以一种可微的方式执行输入归一化，并且不需要在每次参数更新后分析整个训练集。

以前的一些方法(例如(Lyu & Simoncelli, 2008))使用在单个训练示例上计算的统计数据，或者，在图像网络中，使用在给定位置上的不同特征地图。然而，这通过放弃激活的绝对规模改变了网络的表示能力。我们想要保存网络中的信息，通过规范化一个训练示例中的激活相对于整个训练数据的统计。

**Mini-Batch**

**3统计标准化**

由于每一层输入的完全白化是昂贵的，并不是所有地方都可微，我们做了两个必要的简化。首先，我们将每个标量特征独立归一化，使其均值为零，方差为1，而不是联合白化层输入输出中的特征。对于具有d维输入x = (x(1)的层…X (d))，我们将标准化每个维度



p

b (k) =

其中，期望和方差是在训练数据集上计算的。如(LeCun et al.， 1998b)所示，这种归一化加快了收敛速度，即使在特征不去相关的情况下。

注意，简单地规范化一个层的每个输入可能会改变该层可以表示的内容。例如，将sigmoid的输入规格化将把它们限制在非线性的线性范围内。为了解决这个问题，我们要确保插入到网络中的转换可以表示单位变换。为了实现这一点,

对于每个活化x(k)，我们引入一对参数γ(k)， β(k)，它们缩放和移动归一化值:

FORMULA

这些参数与原模型参数一起学习，恢复网络的表示能力。的确，通过设置γ(k) = Var[x(k)]和β(k) = E (x)(k)，我们可以恢复原来的激活，如果那是最理想的事情。

在批处理设置中，每个训练步骤都是基于整个训练集的，我们会使用整个训练集来规范化激活。然而，当使用随机优化时，这是不切实际的。因此，我们进行了第二次简化:由于我们在随机梯度训练中使用了小批，每个小批产生对每个激活的均值和方差的估计。这样，用于归一化的统计数据可以充分参与梯度反向传播。请注意，使用小批是通过计算每个维度的方差而不是联合协方差实现的;在联合的情况下，将需要正则化，因为小批的大小可能小于被白化的激活数，从而导致奇异的协方差矩阵。

考虑一个大小为m的小批处理B。由于标准化是独立应用于每个激活，让我们关注一个特定的激活x(k)，为了清晰起见省略k。我们在小批量中有m个激活值，

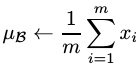
B = {x1...m}。

设标准化值是xb1...m，它们的线性变换是y1...m。我们指的是变换

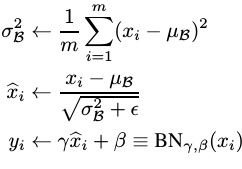
FORMULA

作为批标准化变换。我们提出了算法1中的BN变换。在算法中，ǫ是一个常量加入到小批方差中，以保证数值稳定性。

**输入:x的值在一个小批处理:B = {x1，…，m};**待学参数:γ，β输出:{yi = BNγ，β(xi)}



/ / mini-batch意味着



//小批量差异//规范化

//缩放和移位

**算法1:批处理归一化变换，应用于在一个小批处理上激活x。**

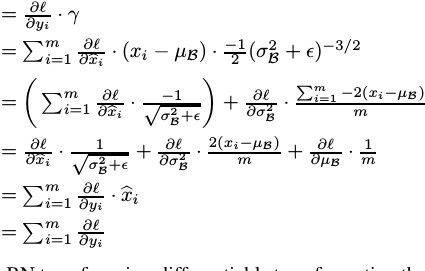
BN变换可以添加到网络中来操作任何激活。在符号y = BN中γ,β我们(x)

3.

表明参数γ和β是需要学习的，但需要注意的是，在每个训练实例中，BN变换并不是独立处理激活的。相反,BNγ,β(x)取决于训练示例和小批处理中的其他示例。缩放和移位的值y被传递到其他网络层。标准化激活xb在转化过程中是内部的，但它们的存在是至关重要的。任何xb的值的分布都是期望值为0，方差为1，只要每个小批的元素都是从相同的分布中采样，Pif我们忽略ǫ。这可以通过观察m xb = 0和1 m i=1 i m i=1 xb2i =1看出，并取expect -

实况。每个归一化激活xb(k)都可以看作是由线性变换y组成的子网络的输入(k) =γ(k)Xb (k) + β(k)，然后进行原网络的其他处理。这些子网络的输入都有固定的平均值和方差，尽管这些归一化xb(k)的联合分布在训练过程中会发生变化，但我们预计归一化输入的引入会加速子网络的训练，从而加速整个网络的训练。

在训练过程中，我们需要通过这个变换反向传播损耗ℓ的梯度，并计算关于BN变换参数的梯度。我们使用链式法则，如下(在简化前):



因此，BN变换是一种将标准化激活引入网络的可微变换。这确保了在模型进行训练时，层可以继续在显示更少内部协变量变化的输入分布上学习，从而加速训练。此外，学习到的仿射变换应用于这些归一化激活，允许BN变换表示恒等变换并保持网络容量。

**3.1用批归一化网络进行训练和推理**

为了对网络进行批规格化，我们根据Alg. 1的规定，指定一个ac- tivations的子集，并为每个ac- tivations插入BN变换。以前接收x作为输入的任何层，现在接收BN(x)。一个使用批归一化的模型可以使用批梯度下降，或随机梯度下降的小批大小m > 1，或任何它的变体，如Adagrad

(Duchi等人，2011)。依赖于小批处理的激活的规范化允许有效的训练，但在推理过程中既不是必要的，也不是理想的;我们希望输出只确定性地依赖于输入。为此，一旦网络训练完毕，我们就使用归一化



b= p

使用总体统计信息，而不是小批量统计信息。

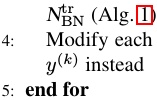
忽略ǫ，这些规范化的激活具有与训练期间相同的均值0和方差1。我们使用无偏方差估计Var[x] = m−1m ·EB(σB2]，其中期望超过训练小批量m和σB2 是样本方差。而使用移动平均线，我们可以在模型训练时跟踪它的准确性。由于均值和方差在推理过程中是固定的，所以归一化只是应用于每个激活的线性变换。它可以进一步由γ标度和β移位组成，得到一个代替BN(x)的单一线性变换。算法2总结了批量归一化网络的训练过程。

**输入:可训练参数网络N Θ;**

输出:批归一化推理网络，NinfBN 1: NBN tr←N //训练BN网络

FORMULA

3:加变换y(k) = BNγ(k),β(k) (x (k))



修改NtrBN中的每一层，输入x(k) 4:取

6:训练NtrBN优化参数Θ∪{γ(k)， β(k)} k k=1



7: NBN inf←NBNtr //推理冻结//参数的BN网络

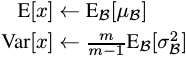
8:当k = 1时…K做

(k)

9:



10:处理多个训练小批B，每个小批m，并平均:



在NinfBN中，替换变换y = BN?γ、β(x)与y =√γ·x +β−√γE [x] Var [x] +ǫ Var [x] +ǫ

12:结束了

**算法2:训练一批归一化网络**

11:

**3.2批量归一化卷积网络**

批归一化可以应用于网络中的任何一组激活。这里，我们关注的是变换

4

包含仿射变换和单元非线性的:

FORMULA

其中W和b为模型的学习参数，g(·)为非线性，如sigmoid或ReLU。这个公式包括完全连接层和卷积层。通过规范化x = Wu+b，我们在非线性函数之前加上BN变换。我们也可以对层输入u进行归一化处理，但是由于u很可能是另一个非线性的输出，所以在训练过程中它的分布形状可能会发生变化，并且限制它的一阶矩和二阶矩不会消除协变量偏移。相比之下，Wu + b更有可能是对称的，非稀疏分布，即“更高斯”(Hyv¨arinen & Oja, 2000);正态化可能产生具有稳定分布的激活。

需要注意的是，由于我们对Wu+b进行了归一化，所以可以忽略偏差b，因为它的影响将被后续的均值减法所抵消(在Alg. 1中，偏差的作用被β所包含)。因此，z = g(Wu +b)被替换为

FORMULA

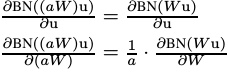
其中BN变换独立应用于x = Wu的每个维度，每个维度有单独的一对学习参数γ(k)， β(k)。

对于卷积层，我们另外希望归一化遵循卷积属性——以便同一特征映射的不同位置的不同元素以相同的方式归一化。为了实现这一点，我们在所有位置联合标准化一个小批处理中的所有激活。Alg。1,我们让B中所有值的集合特征映射在mini-batch和空间位置的元素,所以mini-batch的尺寸大小的地图p×m和特性,我们使用的有效mini-batch大小m B ' = | | = m·p q。我们学习两个参数γ(k) 和β(k) 每个特性映射，而不是每个激活。算法2进行了类似的修改，以便在推理过程中，BN变换对给定特征映射中的每个激活应用相同的线性变换。

反向传播过程中的梯度导致模型爆炸。然而，使用批处理归一化，通过一个层的反向传播不受其参数规模的影响。实际上，对于标量a，

FORMULA

我们可以证明这一点



尺度不影响层的雅可比矩阵，因此也不影响梯度的传播。权重越大，梯度越小，批归一化可以稳定参数增长。

我们进一步推测Batch Normalization可能会导致层雅可比矩阵的奇异值接近于1，已知这有利于训练(Saxe et al.， 2013)。考虑两个具有标准化输入的连续层，以及这些标准化向量之间的转换:bz = F(bx)。假设bx和bz是高斯且不相关的，且F(bx)≈Jbx是对给定模型参数的线性变换，则bx和bz都有单位协方差，I = Cov[bz] = JCov[bx]J T = jjt。因此，JJ T = I，所以J的所有奇异值都等于1，这在反向传播过程中保持了梯度大小。在现实中，变换不是线性的，并且归一化的值不能保证是高斯的或独立的，但是我们期望批归一化有助于使梯度传播更好地表现。批归一化对梯度传播的精确影响仍是一个有待进一步研究的领域。

**3.4批归一化**

**模型**

当使用批处理归一化进行训练时，一个训练示例与小批处理中的其他示例一起出现，训练网络不再为给定的训练示例产生确定性的值。在

**3.3批量归一化使得网络的泛化程度更高。**而辍学率

在我们的实验中，我们发现这种效应是有利的(Srivastava et al.， 2014)通常用于减少过度

在传统的深度网络中，过高的学习率可能会导致梯度爆炸或消失，以及陷入较差的局部最小值。批标准化有助于解决这些问题。通过将整个网络的激活规格化，它可以防止参数的小变化放大为梯度激活的更大和次优变化;例如，它可以防止训练陷入非线性的饱和状态。

成批归一化也使训练对参数尺度更具弹性。通常，大的学习速率可能会增加层参数的规模，然后放大

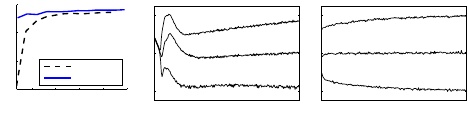
拟合，在批归一化网络中，我们发现它可以被去除或减弱强度。

**4实验**

**4.1随时间的激活**

为了验证内部协变量偏移对训练的影响，以及批归一化处理对抗它的能力，我们考虑了MNIST数据集上数字类的预测问题(LeCun et al.， 1998a)。我们使用了一个非常简单的网络，用一个28x28的二进制图像作为输入，然后

5



1

2

0.9

0

0.8

没有BN

用BN 0.7 10K 20K 30K 40K 50K−2

2

0

−2

(一)

没有BN (b)

与BN (c)

图1:(a)经过批归一化和未经过批归一化训练的MNIST网络的测试精度与训练步数之比。*批处理归一化可以帮助网络更快地训练，获得更高的准确率。*(b, c)在训练过程中，输入分布演变为典型的s型曲线，如{15,50,85}个百分位数所示。*批归一化使分布更加稳定，减少了内部协变量偏移。*

3个完全连接的隐藏层，每个层有100个激活。每个隐层计算y = g(Wu+b)具有sigmoid非线性，并将权值W初始化为小的随机高斯值。最后一个隐藏层后面是一个完全连接的层，包含10个激活(每个类一个)和交叉熵损失。我们对该网络进行了50000个步骤的训练，每个小批有60个示例。我们为网络的每一个隐藏层添加了批标准化，如3.1节。我们感兴趣的是基线和批标准化网络之间的比较，而不是在MNIST上达到最先进的性能状态(所描述的体系结构没有做到这一点)。

图1(a)显示了随着训练的进行，两个网络对保留测试数据的正确预测的比例。批归一化网络具有较高的测试精度。为了探究其原因，我们研究了原始网络N和批归一化网络N的sigmoid输入trBN (Alg. 2)在训练过程中。在图1(b,c)中，我们展示了每个网络最后一层的一个典型激活，它的分布是如何演变的。原始网络中的分布随着时间的推移而发生显著变化，包括其均值和方差，这使得后续层次的训练变得复杂。相比之下，随着训练的进行，成批归一化网络的分布更加稳定，有助于训练的进行。

**4.2 ImageNet分类**

我们将批标准化应用于Inception网络的一个新变体(Szegedy等，2014)，并对ImageNet分类任务进行训练(Russakovsky等，2014)。该网络有大量的卷积和池化层，有一个softmax层来预测1000种可能性中的图像类别。卷积层使用ReLU作为非线性。与(Szegedy et al.， 2014)中描述的网络的主要区别是，5 × 5卷积层被2个连续的3 × 3卷积层和128个滤波器所取代。该网络包含13.6·106个参数，除顶层softmax层外，没有完全连通的层。更多的

详情见附录。在本文的其余部分中，我们将这个模型称为Inception。该模型使用带动量的随机梯度下降(Sutskever等人，2013)的版本进行训练，使用32个小批量。培训使用了大规模的分布式体系结构(类似于(Dean et al.， 2012))。所有的网络都是通过计算验证精度@1来评估训练过程的，即在一个固定的集合上，每幅图像使用单一裁剪，从1000种可能性中预测正确标签的概率。

在我们的实验中，我们评估了用批标准化初始化的几个修改。在所有情况下，Batch Normalization都以卷积的方式应用于每个非线性的输入，如第3.2节所述，同时保持体系结构的其余部分不变。

**4.2.1加速BN网络**

简单地将批标准化添加到网络中并不能充分利用我们的方法。为此，我们进一步改变了网络及其训练参数，如下:

*提高学习速率。*在批归一化模型中，我们已经能够从更高的学习速率中实现训练加速，而没有任何副作用(第3.3节)。

*删除辍学。*如第3.4节所述，批标准化实现了一些与Dropout相同的目标。从Modified BN-Inception中移除Dropout可以加速训练，而不会增加过拟合。

*减少L*2 *体重正规化。*而在《盗梦空间》中，L2 模型参数上的损失控制过拟合，在Modified BN-Inception中，这种损失的权重减少了5个因子。我们发现这提高了对保留的验证数据的准确性。

*加速学习速率衰减。*在训练《盗梦空间》时，学习率呈指数衰减。因为我们的网络训练速度比《盗梦空间》快，所以我们将学习速度降低了6倍。

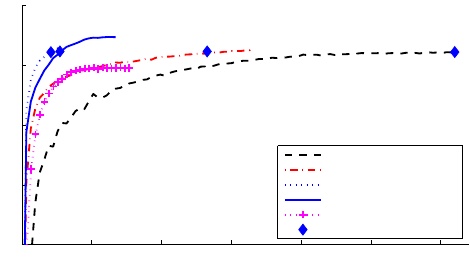
*当初始化和其他网络(Srivastava等人，2014)从中受益时，我们发现使用批处理标准化是不必要的。*

*更彻底地洗牌训练例子。*我们在shardshuffling中启用了训练数据，这可以防止相同的例子总是同时出现在一个小批处理中。这导致验证准确性提高了约1%，这与batchnormationas regularizer(第3.4节)的观点一致:当每次对一个例子的影响不同时，我们方法中固有的随机化应该是最有益的

观察。

*减少光度失真。*因为批处理归一化网络训练更快，观察每个训练例子的次数更少，我们让训练器聚焦于更“真实”的图像，通过更少地扭曲它们。

6



0.7

0.6

0.5

《盗梦空间》

BN−基线

BN−x5

BN−x30

BN−x5−Sigmoid匹配Inception的步骤

0.4

5米

10米

15米

20米

25米

30米

图2:Inception及其成批归一化变体的单一作物验证准确性与训练步骤的数量。

**4.2.2单独的网络分类**

我们对以下网络进行了评估，所有网络都使用LSVRC2012训练数据进行了训练，并对验证数据进行了测试:

Inception:第4.2节开头描述的网络，初始学习率为0.0015。

BN-Baseline:与每个非线性前的批标准化初始相同。

BN-x5:初始与批标准化和第4.2.1节中的修改。初始学习率提高了5倍，达到0.0075。同样的学习率的增加与最初的初始，使模型参数达到了机器无穷大。

BN-x30:类似于BN-x5，但初始学习率为0.045(是Inception的30倍)。

BN-x5- sigmoid:类似于BN-x5，但具有sigmoid非线性g(t) = 1+exp(−x)1而不是ReLU。我们还在,

他试图用“s”形来训练最初的《盗梦空间》，但模型仍然保持着与概率相等的准确性。

在图2中，我们展示了网络的验证准确性，作为训练步骤数的函数。经过31·106步的训练，初始准确率达到72.2%。如图3所示，对于每个网络，达到相同的72.2%准确率所需要的训练步数，以及网络达到的最大验证精度和达到的步数。

通过仅使用批处理归一化(BN-Baseline)，我们在不到一半的训练步骤中匹配初始的准确性。通过第4.2.1节的修改，我们显著提高了网络的训练速度。BN-x5需要比Inception少14倍的步骤才能达到72.2%的精度。有趣的是，进一步提高学习率(BN-x30)会导致模型最初的训练速度稍慢，但允许它最终达到更高的精度。6·10后达到74.8%6 步骤，即比《盗梦空间》所需的步骤少5倍，达到72.2%。

我们还验证了内部协变量移动的减少允许深度网络与批归一化

|  |  |  |
| --- | --- | --- |
| 模型 | 步骤到72.2% | 马克斯准确性 |
| 《盗梦空间》  BN-Baseline  BN-x5  BN-x30  BN-x5-Sigmoid | 31.0·106  13.3·106  2.1·106  2.7·106 | 72.2%  72.7%  73.0%  74.8%  69.8% |

图3:对于Inception和批归一化的变量，达到Inception的最大正确率(72.2%)所需的训练步骤数，以及网络实现的最大正确率。

当使用sigmoid作为非线性时进行训练，尽管训练这样的网络有众所周知的困难。的确，BN-x5-Sigmoid的准确率为69.8%。没有批量归一化，初始与sigmoid从未达到优于1/1000的精度。

**4.2.3系综分类**

目前在ImageNet大规模视觉识别竞赛中报道的最佳结果是传统模型的深度图像集成(Wu et al.， 2015)和集成模型(He et al.， 2015)。后者报告的前5位错误为4.94%，由ILSVRC服务器评估。这里我们报告了前5个验证错误为4.9%，测试错误为4.82%(根据ILSVRC服务器)。根据(Russakovsky et al.， 2014)，这在之前的最佳结果上有所改进，并且超过了人类评分者的估计准确性。

在我们的集成中，我们使用了6个网络。每一个都基于BN-x30，通过以下方法进行了修改:增加了卷积层中的初始权值;使用Dropout (Dropout概率为5%或10%，而最初的Inception是40%);使用非卷积，每激活批标准化与最后隐藏层的模型。每个网络在大约6·106次训练后达到了最大的准确率。集成预测是基于组成网络预测的类概率的算术平均。集成和多丛推理的细节类似(Szegedy et al.， 2014)。

我们在图4中演示了批标准化使我们能够在ImageNet分类挑战基准上以健康的余量设置新的技术水平。

**5的结论**

我们提出了一种新的机制来显著加速深度网络的训练。它是基于协变量转移的前提，这是已知的复杂的机器学习系统训练，也ap

7

决议

作物

模型

模型

(错误

五大错误

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GoogLeNet合奏 | 224 | 144 |  | 7 | - | 6.67% |
| 深低分辨率图像 | 256 | - |  | 1 | - | 7.96% |
| 深度图像分辨率 | 512 | - |  | 1 | 24.88 | 7.42% |
| 深度图像合奏 | 变量 | - |  | - | - | 5.98% |
| BN-Inception单一的农作物 | 224 | 1 |  | 1 | 25.2% | 7.82% |
| BN-Inception multicrop | 224 | 144 |  | 1 | 21.99% | 5.82% |
| BN-Inception合奏 | 224 | 144 |  | 6 | 20.1% | **\* 4.9%** |

图4:在提供的包含50000张图片的验证集上，与以前的技术状态进行批标准化初始比较。*\*BN-Inception集合在ImageNet测试集的100000张图片上达到了4.82%的前5位错误。*

层对子网络和层，并从内部激活的网络删除它可能有助于训练。我们提出的方法的力量来自于激活的规范化，并将这种规范化纳入网络体系结构本身。这确保了规范化被任何用于训练网络的优化方法适当地处理。为了实现在深度网络训练中常用的随机优化方法，我们对每个小批量进行归一化，并通过归一化参数反向传播梯度。批处理标准化只增加两个额外的参数每次激活，这样做保持了网络的表示能力。我们提出了一种用批归一化网络构造、训练和执行推理的算法。由此产生的网络可以用饱和的非线性进行训练，对增加的训练率有更大的容忍度，而且通常不需要Dropout来进行正则化。

仅仅将批归一化添加到一个最先进的图像分类模型中，就可以大大提高训练速度。通过进一步提高学习率，去除Dropout，并应用批归一化的其他修改，我们只需要一小部分训练步骤就达到了先前的技术水平，然后在单网络图像分类方面击败了现有的技术水平。此外，通过结合使用批处理归一化训练的多个模型，我们的性能明显优于ImageNet上最知名的系统。

有趣的是，我们的方法与(G¨ulc ehre & Bengio, 2013)的标准化层有相似之处，尽管这两种方法源于非常不同的目标，执行不同的任务。Batch Normalization的目标是在整个训练过程中实现激活值的稳定分布，在我们的实验中，我们在非线性之前应用它，因为在非线性中，匹配的第一和第二矩更可能导致稳定分布。相反，(G¨ulc本吉奥，2013)将标准化层应用于非线性的输出，导致稀疏激活。在我们的大规模图像分类实验中，我们没有观察到非线性输入是稀疏的，无论是有或没有成批归一化。其他值得注意的差异,

entiating批规范化的特征包括规模和学习转变,使BN变换代表身份(标准化层不需要这个,因为随之而来的线性变换,从概念上讲,吸收必要的规模和转移),卷积处理层,确定性推理,并不取决于mini-batch,和批处理,网络中每个卷积层正常化。

在这项工作中，我们没有探索批处理标准化潜在支持的全部可能性。我们未来的工作包括将我们的方法应用于递归神经网络(Pascanu et al.， 2013)，其中内部协变量移动和消失或爆炸的梯度可能特别严重，这使我们能够更彻底地测试归一化改进梯度传播的假设(第3.3节)。我们计划调查是否批标准化有助于领域适应气候变化,在传统意义上,即标准化执行的网络是否会使它更容易推广到新的数据分布,也许只有一个人口的重新计算均值和方差(Alg。2)。最后,我们相信的进一步理论分析算法将允许更多的改进和应用。

**参考文献**

本吉奥，约书亚，格洛洛，泽维尔。理解训练深度前馈神经网络的困难。在2010年5月的AISTATS学报，第9卷，第249 - 256页。

Dean, Jeffrey, Corrado, Greg S, Monga, Rajat, Chen, Kai, Devin, Matthieu, Le, Quoc V.， Mao, Mark Z.， Ranzato, Marc 'Aurelio, Senior, Andrew, Tucker, Paul, Yang, Ke, and Ng, Andrew Y.大规模分布的深层网络。在少量的酒,2012。

纪尧姆·德斯雅丁斯和科莱·卡乌库奥格鲁。自然神经网络。(未出版)。

Duchi, John, Hazan, Elad, Singer, Yoram。自适应次梯度方法的在线学习和随机

8

优化。*j·马赫。学习。Res 12:2121 - 2159, 2011年7月。*ISSN 1532 - 4435。

所有的婚礼举办权都要在举办权范围之内。先验信息对优化的重要性。*2013年,abs / 1301.4083。*

何凯，张晓，任胜，孙杰。深入研究整流器:超越人类水平的图像网分类。*ArXiv电子出版物，2015年2月。*

独立成分分析:算法与应用。*神经。*2000年5月。

江,静。统计分类器领域适应性研究综述，2008。

基于梯度学习的文档识别应用。*中国生物医学工程学报，22 (4):461 - 468,2008a。*

关键词:高效支撑;在奥尔，g和K。《神经网络:交易的诀窍》。施普林格,1998 b。

基于分裂归一化的非线性图像表示。《计算机视觉与模式识别》，第1-8页。IEEE计算机协会，2008年6月23-28日。doi: 10.1109 / CVPR。2008.4587821。

奈尔，维诺德和辛顿，杰弗里e矫正线性单位改善限制玻尔兹曼机。在ICML，第807-814页。Omnipress, 2010年。

帕斯卡努，拉兹万，米克洛夫，托马斯，本吉奥，约书亚。关于训练循环神经网络的困难。第三十届机器学习国际会议论文集，ICML 2013, Atlanta, GA, USA, 16- 21 June 2013, pp. 1310-1318, 2013。

波维，丹尼尔，张小慧，和库丹普尔，圣吉夫。基于自然梯度和参数平均的深度神经网络并行训练。*2014年,abs / 1410.7455。*

莱科，塔帕尼，瓦尔波拉，哈里，和勒昆，扬。通过感知机的线性变换，深度学习变得更加容易。人工智能与统计学(AISTATS)，第924-932页，2012。

Russakovsky, Olga, Deng, Jia, Su, Hao, Krause, Jonathan, Satheesh, Sanjeev, Ma, Sean, Huang, Zhiheng, Karpa- thy, Andrej, Khosla, Aditya, Bernstein, Michael, Berg, Alexander C.和Fei-Fei, Li。ImageNet大型视觉识别挑战，2014。

Saxe, Andrew M.， McClelland, James L.和Ganguli, Surya。深度线性神经网络非线性动力学学习的精确解。*2013年,abs / 1312.6120。*

Shimodaira Hidetoshi。利用对数似然函数加权改进协变量移位下的预测推理。*统计规划与推断，90(2):227-244,2000年10月*

斯里瓦斯塔瓦，尼蒂什，欣顿，杰弗里，克里热夫斯基，亚历克斯，苏茨基弗，伊利亚，萨拉胡蒂诺夫，鲁斯兰。Dropout:防止神经网络过拟合的一种简单方法。*j·马赫。学习。第15(1):1929-1958,2014年1月。*

Sutskever, Ilya, Martens, James, Dahl, George E，和Hinton, Geoffrey E.关于深度学习中初始化和动量的重要性。在ICML(3)，卷28的JMLRProceedings, pp.1139-1147。JMLR.org, 2013。

Szegedy, Christian, Liu, Wei, Jia, Yangqing, Sermanet, Pierre, Reed, Scott, angelov, Dragomir, Erhan, Dumitru, Vanhoucke, Vincent, and Rabinovich, Andrew。深入研究卷积。*2014年,abs / 1409.4842。*

威斯勒，西蒙，内，赫尔曼。对数线性训练的收敛性分析。(In Shawe-Taylor, J.， Zemel, R.S.， Bartlett, P.， Pereira, F.C.N.，和Weinberger, K.Q.(编著)，Advances In Neural Information Processing Systems24, pp. 657-665, Granada, Spain, December 2011.)

Wiesler, Simon, Richard, Alexander, Schl¨uter, Ralf, Ney, Hermann。用于大规模深度学习的均值归一化随机梯度。在IEEE国际声学、语音和信号处理会议上，pp. 180-184, Florence, Italy, 2014年5月。

吴任燕沈恩山易党青青孙刚。深度图像:放大图像识别，2015。

**附录**

**所使用的初始模型的变体**

图5记录了与googleet体系结构相比所执行的更改。对此表的解释请咨询(Szegedy et al.， 2014)。与googleet模型相比，值得注意的架构变化包括:

•将5×5卷积层替换为两个连续的3×3卷积层。这将使网络的最大深度增加9

9

体重层。同时增加了25%的参数数，增加了约30%的计算量。

•28×28初始模块数量从2个增加到3个。

•在模块内部，有时使用平均池，有时使用最大池。这在与表的池层对应的条目中表示。

•在任何两个初始模块之间没有跨板的池化层，但是在模块3c, 4e的过滤器串联之前，stride-2卷积/池化层被使用。

我们的模型在第一卷积层采用深度乘子8的可分离卷积。这降低了计算成本，同时增加了训练时的内存消耗。

10

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **卷积类型\*** | **补丁大小/步幅7×7/2** | **产出规模112×112×64** | **深度1** | # 1×1 | # 3×3减少 | # 3×3 | **# 3×3减少两倍** | **# 3×3的两倍** | **池+项目** |
| 马克斯池卷积 | 3×3/2 3×3/1 | 56 56××64 56 56××192 | 0 1 |  | 64 | 192 |  |  |  |
| 马克斯池 | 3×3/2 | 28××28日192年 | 0 |  |  |  |  |  |  |
| Inception (3a) Inception (3b) |  | 256年××28日28日28日××28日320年 | 3个3 | 64 64 | 64 64 | 64 96 | 64 64 | 96 96 | 平均+ 32平均+ 64 |
| Inception (3c) Inception (4a) Inception (4b) | 步2 | 576年××28日28日14×14×576 14×14×576 | 3 3 3 | 0 224 192 | 128 64 96 | 160 96 128 | 64 96 96 | 96 128 128 | Max + through avg + 128 |
| Inception (4c) Inception (4d) Inception (4e) | 步2 | 14×14×576 14×14×576 14×14×1024 | 3 3 3 | 160 96 0 | 128 128 128 | 160 192 192 | 128 160 192 | 160 192 256 | 平均+ 128平均+ 128最大+通过 |
| 初始(5a)初始(5b)平均池 | 7×7/1 | 1024年7×7×7×7×1024 1×1×1024 | 3 3 0 | 352 352 | 192 192 | 320 320 | 160 192 | 224 224 | 平均+ 128 Max + 128 |

图5:初始架构

11