http://www.sohu.com/a/194337310\_717210

**如何一步一步提高图像分类准确率？**

2017-09-25 07:00 [对比](http://www.sohu.com/tag/59774) */*[中国移动](http://www.sohu.com/tag/48863) */*[IFA](http://www.sohu.com/tag/72590)

**本文原作者曹荣禹，本文原载于知乎专栏——机器学习教程。AI 研习社已获得转载授权。**

**一、问题描述**

当我们在处理图像识别或者图像分类或者其他机器学习任务的时候，我们总是迷茫于做出哪些改进能够**提升模型的性能**（识别率、分类准确率）。。。或者说我们在漫长而苦恼的调参过程中**到底调的是哪些参数**。。。所以，我花了一部分时间在公开数据集 CIFAR-10 [1] 上进行探索，来总结出一套方法能够**快速高效并且有目的性**地进行网络训练和参数调整。

CIFAR-10 数据集有 60000 张图片，每张图片均为分辨率为 32\*32 的彩色图片（分为 RGB3 个信道）。CIFAR-10 的分类任务是将每张图片分成青蛙、卡车、飞机等 10 个类别中的一个类别。本文主要使用基于卷积神经网络的方法（CNN）来设计模型，完成分类任务。

首先，为了能够在训练网络的同时能够检测网络的性能，我对数据集进行了训练集 / 验证集 / 测试集的划分。训练集主要用户进行模型训练，验证集主要进行参数调整，测试集主要进行模型性能的评估。因此，我将 60000 个样本的数据集分成了，45000 个样本作为训练集，5000 个样本作为验证集，10000 个样本作为测试集。接下来，我们一步步来分析，如果进行模型设计和改进。

**二、搭建最简单版本的 CNN**

对于任何的机器学习问题，我们一上来肯定是采用最简单的模型，一方面能够快速地 run 一个模型，以了解这个任务的难度，另一方面能够有一个 baseline 版本的模型，利于进行对比实验。所以，我按照以往经验和网友的推荐，设计了以下的模型。

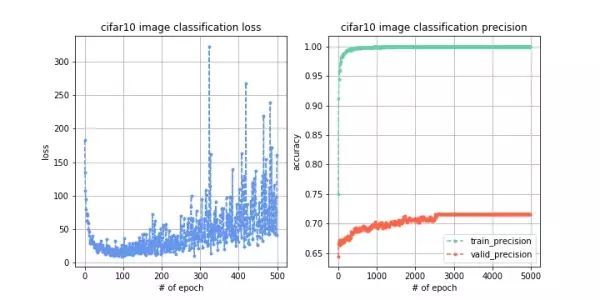
模型的输入数据是网络的输入是一个 4 维 tensor，尺寸为 (128, 32, 32, 3)，分别表示一批图片的个数 128、图片的宽的像素点个数 32、高的像素点个数 32 和信道个数 3。首先使用多个卷积神经网络层进行图像的特征提取，卷积神经网络层的计算过程如下步骤：

1. **卷积层 1**：卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 64，池化大小 2\*2，池化步长 2，池化类型为最大池化，激活函数 ReLU。
2. **卷积层 2**：卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 128，池化大小 2\*2，池化步长 2，池化类型为最大池化，激活函数 ReLU。
3. **卷积层 3**：卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 256，池化大小 2\*2，池化步长 2，池化类型为最大池化，激活函数 ReLU。
4. **全连接层**：隐藏层单元数 1024，激活函数 ReLU。
5. **分类层**：隐藏层单元数 10，激活函数 softmax。

参数初始化，所有权重矩阵使用 random\_normal(0.0, 0.001)，所有偏置向量使用 constant(0.0)。使用 cross entropy 作为目标函数，使用 Adam 梯度下降法进行参数更新，学习率设为固定值 0.001。

该网络是一个有三层卷积层的神经网络，能够快速地完成图像地特征提取。全连接层用于将图像特征整合成分类特征，分类层用于分类。cross entropy 也是最常用的目标函数之一，分类任务使用 cross entropy 作为目标函数非常适合。Adam 梯度下降法也是现在非常流行的梯度下降法的改进方法之一，学习率过大会导致模型难以找到较优解，设置过小则会降低模型训练效率，因此选择适中的 0.001。这样，我们最基础版本的 CNN 模型就已经搭建好了，接下来进行训练和测试以观察结果。

训练 5000 轮，观察到 loss 变化曲线、训练集准确率变化曲线和验证集准确率变化曲线如下图。测试集准确率为**69.36%**。



结果分析：首先我们观察训练 loss（目标函数值）变化，刚开始 loss 从 200 不断减小到接近 0，但是在 100 轮左右开始出现震荡，并且随着训练幅度越来越大，说明模型不稳定。然后观察训练集和验证集的准确率，发现训练集准确率接近于 1，验证集准确率稳定在 70% 左右，说明模型的泛化能力不强并且出现了过拟合情况。最后评估测试集，发现准确率为 69.36%，也没有达到很满意的程度，说明我们对模型需要进行很大的改进，接下来进行漫长的调参之旅吧！

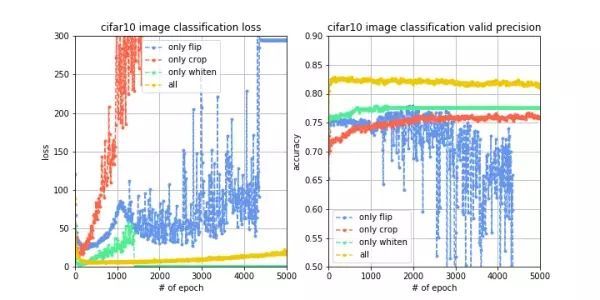
**三、数据增强有很大的作用**

使用**数据增强技术（data augmentation）**，主要是在训练数据上增加微小的扰动或者变化，一方面可以增加训练数据，从而提升模型的泛化能力，另一方面可以增加噪声数据，从而增强模型的鲁棒性。主要的数据增强方法有：翻转变换 flip、随机修剪（random crop）、色彩抖动（color jittering）、平移变换（shift）、尺度变换（scale）、对比度变换（contrast）、噪声扰动（noise）、旋转变换 / 反射变换 （rotation/reflection）等，可以参考 Keras 的官方文档 [2] 。获取一个 batch 的训练数据，进行数据增强步骤之后再送入网络进行训练。

我主要做的数据增强操作有如下方面：

1. **图像切割**：生成比图像尺寸小一些的矩形框，对图像进行随机的切割，最终以矩形框内的图像作为训练数据。
2. **图像翻转**：对图像进行左右翻转。
3. **图像白化**：对图像进行白化操作，即将图像本身归一化成 Gaussian(0,1) 分布。

为了进行对比实验，观测不同数据增强方法的性能，实验 1 只进行图像切割，实验 2 只进行图像翻转，实验 3 只进行图像白化，实验 4 同时进行这三种数据增强方法，同样训练 5000 轮，观察到 loss 变化曲线、训练集准确率变化曲线和验证集准确率变化曲线对比如下图。



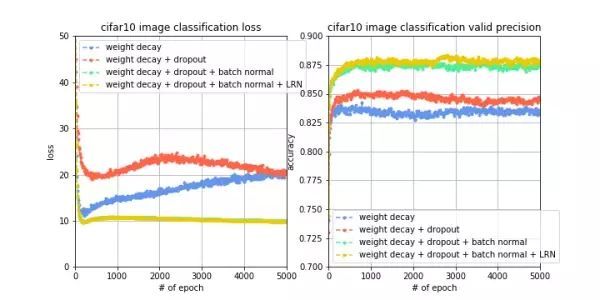
结果分析：我们观察训练曲线和验证曲线，很明显地发现图像白化的效果好，其次是图像切割，再次是图像翻转，而如果同时使用这三种数据增强技术，不仅能使训练过程的 loss 更稳定，而且能使验证集的准确率提升至**82%**左右，提升效果十分明显。而对于测试集，准确率也提升至**80.42%**。说明图像增强确实通过增加训练集数据量达到了提升模型泛化能力以及鲁棒性的效果，从准确率上看也带来了将近 10% 左右的提升，因此，数据增强确实有很大的作用。但是对于 80% 左右的识别准确率我们还是不够满意，接下来继续调参。

**四、从模型入手，使用一些改进方法**

接下来的步骤是从模型角度进行一些改进，这方面的改进是诞生论文的重要区域，由于某一个特定问题对某一个模型的改进千变万化，没有办法全部去尝试，因此一般会实验一些 general 的方法，比如批正则化（batch normalization）、权重衰减（weight decay）。我这里实验了 4 种改进方法，接下来依次介绍。

1. **权重衰减**（weight decay）：对于目标函数加入正则化项，限制权重参数的个数，这是一种防止过拟合的方法，这个方法其实就是机器学习中的 l2 正则化方法，只不过在神经网络中旧瓶装新酒改名为 weight decay [3]。
2. **dropout**：在每次训练的时候，让某些的特征检测器停过工作，即让神经元以一定的概率不被激活，这样可以防止过拟合，提高泛化能力 [4]。
3. **批正则化**（batch normalization）：batch normalization 对神经网络的每一层的输入数据都进行正则化处理，这样有利于让数据的分布更加均匀，不会出现所有数据都会导致神经元的激活，或者所有数据都不会导致神经元的激活，这是一种数据标准化方法，能够提升模型的拟合能力 [5]。
4. **LRN**：LRN 层模仿生物神经系统的侧抑制机制，对局部神经元的活动创建竞争机制，使得响应比较大的值相对更大，提高模型泛化能力。

为了进行对比实验，实验 1 只使用权重衰减，实验 2 使用权重衰减 + dropout，实验 3 使用权重衰减 + dropout + 批正则化，实验 4 使用权重衰减 + dropout + 批正则化 + LRN，同样都训练 5000 轮，观察到 loss 变化曲线、训练集准确率变化曲线和验证集准确率变化曲线对比如下图。



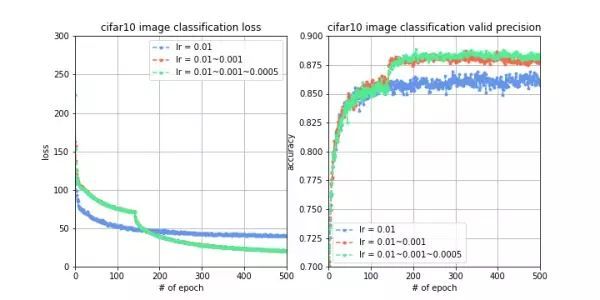
结果分析：我们观察训练曲线和验证曲线，随着每一个模型提升的方法，都会使训练集误差和验证集准确率有所提升，其中，批正则化技术和 dropout 技术带来的提升非常明显，而如果同时使用这些模型提升技术，会使验证集的准确率从 82% 左右提升至 88% 左右，提升效果十分明显。而对于测试集，准确率也提升至**85.72%**。我们再注意看左图，使用 batch normalization 之后，loss 曲线不再像之前会出现先下降后上升的情况，而是一直下降，这说明 batch normalization 技术可以加强模型训练的稳定性，并且能够很大程度地提升模型泛化能力。所以，如果能提出一种模型改进技术并且从原理上解释同时也使其适用于各种模型，那么就是非常好的创新点，也是我想挑战的方向。现在测试集准确率提升至 85% 左右，接下来我们从其他的角度进行调参。

**五、变化的学习率，进一步提升模型性能**

在很多关于神经网络的论文中，都采用了变化学习率的技术来提升模型性能，大致的想法是这样的：

1. 首先使用较大的学习率进行训练，观察目标函数值和验证集准确率的收敛曲线。
2. 如果目标函数值下降速度和验证集准确率上升速度出现减缓时，减小学习率。
3. 循环步骤 2，直到减小学习率也不会影响目标函数下降或验证集准确率上升为止。

为了进行对比实验，实验 1 只使用 0.01 的学习率训练，实验 2 前 10000 个 batch 使用 0.01 的学习率，10000 个 batch 之后学习率降到 0.001，实验 3 前 10000 个 batch 使用 0.01 的学习率，10000~20000 个 batch 使用 0.001 的学习率，20000 个 batch 之后学习率降到 0.0005。同样都训练 5000 轮，观察到 loss 变化曲线、训练集准确率变化曲线和验证集准确率变化曲线对比如下图。



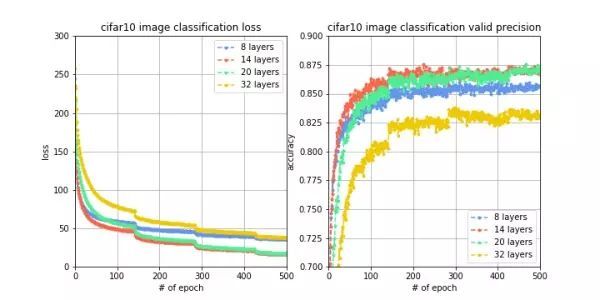
结果分析：我们观察到，当 10000 个 batch 时，学习率从 0.01 降到 0.001 时，目标函数值有明显的下降，验证集准确率有明显的提升，而当 20000 个 batch 时，学习率从 0.001 降到 0.0005 时，目标函数值没有明显的下降，但是验证集准确率有一定的提升，而对于测试集，准确率也提升至**86.24%**。这说明，学习率的变化确实能够提升模型的拟合能力，从而提升准确率。学习率在什么时候进行衰减、率减多少也需要进行多次尝试。一般在模型基本成型之后，使用这种变化的学习率的方法，以获取一定的改进，精益求精。

**六、加深网络层数，会发生什么事情？**

现在深度学习大热，所以，在计算资源足够的情况下，想要获得模型性能的提升，大家最常见打的想法就是增加网络的深度，让深度神经网络来解决问题，但是简单的网络堆叠不一定就能达到很好地效果，抱着深度学习的想法，我按照 plain-cnn 模型 [6]，我做了接下来的实验。

1. **卷积层 1**：卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 16，激活函数 ReLU，使用 batch\_normal 和 weight\_decay，接下来的 n 层，卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 16，激活函数 ReLU，使用 batch\_normal 和 weight\_decay。
2. **卷积层 2**：卷积核大小 3\*3，卷积核移动步长 2，卷积核个数 32，激活函数 ReLU，使用 batch\_normal 和 weight\_decay，接下来的 n 层，卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 32，激活函数 ReLU，使用 batch\_normal 和 weight\_decay。
3. **卷积层 3**：卷积核大小 3\*3，卷积核移动步长 2，卷积核个数 64，激活函数 ReLU，使用 batch\_normal 和 weight\_decay，接下来的 n 层，卷积核大小 3\*3，卷积核移动步长 1，卷积核个数 64，激活函数 ReLU，使用 batch\_normal 和 weight\_decay。
4. **池化层**：使用全局池化，对 64 个隐藏单元分别进行全局池化。
5. **全连接层**：隐藏层单元数 10，激活函数 softmax，使用 batch\_normal 和 weight\_decay。

为了进行对比实验，进行 4 组实验，每组的网络层数分别设置 8，14，20 和 32。同样都训练 5000 轮，观察到 loss 变化曲线、训练集准确率变化曲线和验证集准确率变化曲线对比如下图。

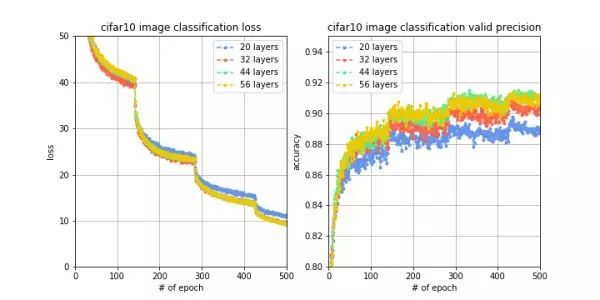


结果分析：我们惊讶的发现，加深了网络层数之后，性能反而下降了，达不到原来的验证集准确率，网络层数从 8 层增加到 14 层，准确率有所上升，但从 14 层增加到 20 层再增加到 32 层，准确率不升反降，这说明如果网络层数过大，由于梯度衰减的原因，导致网络性能下降，因此，需要使用其他方法解决梯度衰减问题，使得深度神经网络能够正常 work。

**七、终极武器，残差网络**

2015 年，Microsoft 用残差网络 [7] 拿下了当年的 ImageNet，这个残差网络就很好地解决了梯度衰减的问题，使得深度神经网络能够正常 work。由于网络层数加深，误差反传的过程中会使梯度不断地衰减，而通过跨层的直连边，可以使误差在反传的过程中减少衰减，使得深层次的网络可以成功训练，具体的过程可以参见其论文 [7]。

通过设置对比实验，观察残差网络的性能，进行 4 组实验，每组的网络层数分别设置 20，32，44 和 56。观察到 loss 变化曲线和验证集准确率变化曲线对比如下图。



结果分析：我们观察到，网络从 20 层增加到 56 层，训练 loss 在稳步降低，验证集准确率在稳步提升，并且当网络层数是 56 层时能够在验证集上达到 91.55% 的准确率。这说明，使用了残差网络的技术，可以解决梯度衰减问题，发挥深层网络的特征提取能力，使模型获得很强的拟合能力和泛化能力。当我们训练深度网络的时候，残差网络很有可能作为终极武器发挥至关重要的作用。

**八、总结**

对于 CIFAR-10 图像分类问题，我们从最简单的卷积神经网络开始，分类准确率只能达到 70% 左右，通过不断地增加提升模型性能的方法，最终将分类准确里提升到了 90% 左右，这 20% 的准确率的提升来自于对数据的改进、对模型的改进、对训练过程的改进等，具体每一项提升如下表所示。

* 改进方法 获得准确率 提升
* 基本神经网络 69.36% -
* + 数据增强 80.42% 11.06%
* + 模型改进 85.72% 16.36%
* + 变化学习率 86.24% 16.88%
* + 深度残差网络 91.55% 22.19%

其中，数据增强技术使用翻转图像、切割图像、白化图像等方法增加数据量，增加模型的拟合能力。模型改进技术包括 batch normalization、weight decay、dropout 等防止过拟合，增加模型的泛化能力。变化学习率通过在训练过程中递减学习率，使得模型能够更好的收敛，增加模型的拟合能力。加深网络层数和残差网络技术通过加深模型层数和解决梯度衰减问题，增加模型的拟合能力。这些改进方法的一步步堆叠，一步步递进，使得网络的拟合能力和泛化能力越来越强，最终获得更高的分类准确率。

本文的所有代码见我的 github，persistforever/cifar10-tensorflow

本文介绍的调参历程，希望能帮助到大家，听说过这么一句话，“读研期间学习人工智能，什么都不用学，学好调参就行了”，而 “调参” 二字却包含着无数的知识，希望大家能多分享神经网络相关的干货。

**引用**

[1]. CIFAR-10 and CIFAR-100 datasets

[2]. https://keras.io/preprocessing/image/

[3]. Krogh A, Hertz J A. A simple weight decay can improve generalization. In ICML, 1991.

[4]. Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting. In JMLR, 2014.

[5]. Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In JMLR, 2015.

[6]. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.

[7]. He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition. In CVPR, 2016.