Cifar-10 与 MNIST 训练集

Dai Jialun

July 12, 2015

1. Cifar10

CIFAR-10 数据集由 10 类总共 60000 张 32×32 的彩色图像组成，其中每一类都有 6000 张图像。在 60000 张图像中，有 50000 张的训练图像与 10000 张的测试图像。

数据集被分成 5 个训练组和 1 个测试组，每一组都有 10000 张图像。测试组中包含了从每一类中随机挑选出的 1000 张图像。训练组从所剩下的图像中随机挑选，因此在某些训练组中，某一类中的图像可能比其他类图像多。另外从整体上看，5 组训练组包含了每一类的 5000 张图像。

数据集中，每一类的图像都是独立存在，没有相互包含的情况。

用 cuda-convnet 实现的卷积神经网络，其系统在没有数据增强的情况下，测试错误为 18%；有数据增强的情况下，测试错误为 11%。

数据集的布局：

Python/Matlab 版：在这里介绍了数据集的 Python 版本。Matlab 版本的布局与其是相同的。

在下载的文档中，包含了 data_batch_1，data_batch_2，data_batch_3，data_batch_4，data_batch_5，以及 test_batch。每一个文件都是由 Python 的 “cPickle” 所产生的 pickled 对象。下面是 Python 程序，会打开一个文件并且返回一个目录。

```
def unpickle(file)
  Import cPickle
  fo=open(file,'rb')
  dict=cPickle.load(fo)
  fo.close()
  return dict
```

通过以上方式加载后，每一个组文件将会生成一个包含以下文件的文件夹：

data 一个类型为 unit8 的 10000×3072 的 numpy 矩阵。这个矩阵的每一行存储一个 32×32 的彩色图像，刚开始的 1024 个数（32×32）表示红色通道的值，接下来的 1024 个数（32×32）表示绿色

通道，最后的 1024 个数（32×32）表示蓝色通道。图像是行优先存储的，因此数组刚开始的 32 是第一行图像的红色通道值。

labels 一个列表包含 10000 个数字，范围为 0 9。在目录中，第 i 个数字表示在数组数据中，第 i 个图像的标注。

数据集包含了另外一个 batches.meta 文件。它也包含了一个 Python 目录对象，有以下内容：

label_names 一个含有 10 个目标的列表，对于上述标注数组，其给了数字标注有意义的名字。例如，label_names[0]=“airplane”

二进制版本的数据集（cifar-10-batches-bin）

二进制版本包含了 data_batch_1.bin，data_batch_2.bin，data_batch_3.bin，data_batch_4.bin，data_batch_5.bin，以及 test_batch_bin。每个文件的格式如下：

```
<1 × label> <3072 × pixel>
...
<1 × label> <3072 × pixel>
```

换句话说，刚开始的比特是第一张图像的标注，是一个范围在 0 9 之间的数字。接下来的 3072 比特是图像像素值。开始的 1024 个数（32×32）是红色通道的值，接下来的 1024 个数（32×32）是绿色通道的值，最后的 1024 个数（32×32）是蓝色通道的值。这些值是按行优先排列的，因此最初的 32 比特是图像的第一行的红色通道值。

每一个文件包含 10000 个这样 3073 个比特的图像“行”，尽管在文件中没有特别划分这些行的界限。因此每个文件都应该有 30730000 比特这么长。

另外，还有一个 batches.meta.txt 的文件。这是一个 ASCII 文件，表示每一个类别名字各自匹配范围在 0 9 之间的数字。这仅仅是一个 10 类别名字的列表，每一个一行。在第 i 行的类别名对应数字标号 i。

2. MNIST

MNIST 数据集是手写数字图像，其训练集有 60000 张图像，测试集有 10000 张图像。这只是一个更大的数据集 NIST 的一个子集。这些数字图像已经经过尺寸的正规化与在固定大小图像中的置中心处理。

这个数据集适合一些人想用学习技术以及模式识别方法来处理真实世界的数据，而且不需要花太多功夫在预处理和格式化方面的工作上。

这个数据集总共有 4 个文件：

train-images-idx3-ubyte.gz training set images (9912422 bytes)

train-labels-idx1-ubyte.gz training set labels (28881 bytes)

t10k-images-idx3-ubyte.gz test set images (1648877 bytes)

t10k-labels-idx1-ubyte.gz test set labels (4542 bytes)

在 NIST 中原始的黑白图像，为了保持长宽比，图像经过尺寸正规化处理变为 20×20 的像素框。结果图像是灰度保真图，使用了规范化算法处理。而且，图像是在 28×28 的图像中心位置。

MNIST 数据集是由 NIST 的 Special Database 3 和 Special Database 1 所组成，二者包含了手写数字的二值图像。在 NIST 中，它将 SD-3 作为训练集，SD-1 作为测试集。然而，SD-3 比 SD-1 更清洗，更容易识别。理由可能是因为 SD-3 是从 Census Bureau 的员工处收集的，而 Sd-1 是从高中生处收集的。从学习实验中得出可靠的结论要求结果与训练集的选择无关，并且测试样本的完整数据集。因此，必须通过混合 NIST 数据集来建立一个全新的数据集。‘

MNIST 训练集是由 SD-3 的 30000 张图像和 SD-1 的 300000 张图像组成。测试集是由 SD-3 的 5000 张图像和 SD-1 的 5000 张图像组成。60000 张的训练集所包含的例子从大约 250 个人中收集到的。因此，我们可以确定训练集与测试集的数字是不同的人所写的。

SD-1 是由 500 个不同的人所写的 58527 张数字图像。与 SD-3 不同，在 SD-3 中每个人所写的数字图像按序列排放，而在 SD-1 中，数据是杂乱无章的。人工分辨 SD-1 是可行的，我们用这些信息来分开所写人。我们将 SD-1 分为两部分：由前 250 位个人的数字作为型的训练集，剩下的 250 个人所写数字放在测试集中。因此，我们有了两个子集，每个自己有大约 30000 张图像。新的训练集从 SD-3 中的 #0 开始抽取足够多的图像，形成一个 60000 张的训练图像。与其相似，新的测试集是由 SD-3 中的 #35000 开始，形成一个 60000 的测试图像。只有测试集中的 10000 张测试图像（从 SD-1 中的 5000 张图像和从 SD-3 中的 5000 张图像）在这里是可用。当然，60000 张的训练图像也是可用的。

MNIST 数据库的文件格式：

数据存储在一个非常简单的用来存放向量和多维矩阵的文件格式中。在文件中所有的整数都