# HDF5文件的创建

需要引入几个关键的库：numpy, matplotlib, h5py

创建hdf5文件关键代码：

**with** h5py.File(h5\_filename, **'w'**) **as** h:  
 ......

h.create\_dataset(**'data'**, data=imgs)  
 h.create\_dataset(**'score'**, data=scores)

省略号部分所做的工作主要是将图像数据和标签数据分别保存到imgs和scores中去，imgs和scores都为numpy.ndarray类型，使用numpy.zeros创建，对于灰度图像，imgs的shape为（n,1,w,h），其中n为图像的数量，1指图像为1个通道，w和h为图像的宽和高.由于读入的图像数据是2维的，可以将其reshape到3维：

img=img.reshape((1,)+img.shape)

如果img.shape为(100,100),那么reshape后，新图像的shape为：(1,)+(100,100)=(1,100,100)

再将img保存到一个shape为(10000,1,100,100)的imgs中去。

scores保存的是图像的标签，如果一张图像只有一个标签，共有10000张图像，那么scores的shape为(10000)

下面的代码将10000张100\*100的灰度图像和对应的标签保存为hdf5格式，同时生产一个train\_h5.txt：

**import** sys  
**import** numpy  
**from** matplotlib **import** pyplot  
**import** h5py  
  
IMAGE\_SIZE = (100, 100)  
MEAN\_VALUE = 128  
filename = **'train.txt'**setname, ext = filename.split(**'.'**)  
  
**with** open(filename, **'r'**) **as** f:  
 lines = f.readlines()  
numpy.random.shuffle(lines)  
sample\_size = len(lines)  
imgs = numpy.zeros((sample\_size, 1,) + IMAGE\_SIZE, dtype=numpy.float32)  
scores = numpy.zeros(sample\_size, dtype=numpy.float32)  
h5\_filename = **'{}.h5'**.format(setname)  
**with** h5py.File(h5\_filename, **'w'**) **as** h:  
 **for** i, line **in** enumerate(lines):  
 image\_name, score = line[:-1].split()  
 img = pyplot.imread(image\_name)[:, :, 0].astype(numpy.float32)  
 img = img.reshape((1, )+img.shape)  
 img -= MEAN\_VALUE  
 imgs[i] = img  
 scores[i] = float(score)  
 **if** (i+1) % 1000 == 0:  
 **print**(**'processed {} images!'**.format(i+1))  
 h.create\_dataset(**'data'**, data=imgs)  
 h.create\_dataset(**'score'**, data=scores)  
  
**with** open(**'{}\_h5.txt'**.format(setname), **'w'**) **as** f:  
 f.write(h5\_filename)

其中train.txt中的内容形式如下：

