3

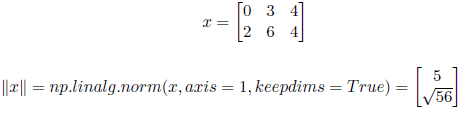
Math.exp()和numpy.exp()

只有后者计算矩阵

8

v = image.reshape((image.shape[0]\*image.shape[1]\*image.shape[2]),1))

9



Axis = 0为行，1为列

11

x\_sum = np.sum(x\_exp,axis = 1,keepdims = True)

keepdims主要用于保持矩阵的二维特性???

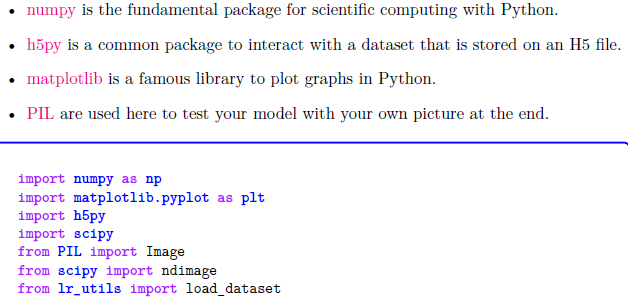
Tic =time.process\_time()

16



矩阵各元素的平方之和，常用numpy.dot(x,x)实现。np.dot()结果为实数，而np.multiply()却是矩阵

18

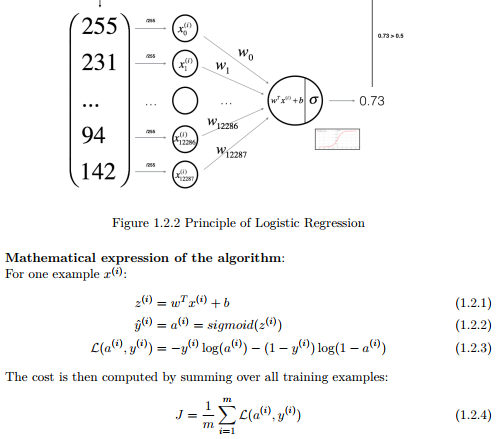
****

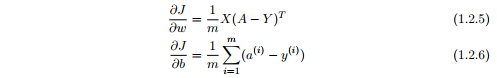
import和from的区别

import引入整个包，使用包中类时，需要加类名限定。from是从包中加载某个类，使用时不再需要加类名限定

18

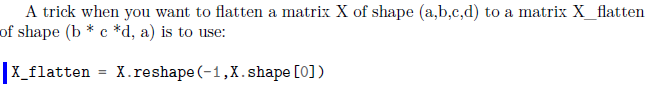
浅层神经网络计算公式，注意到矩阵乘法中w经过了转置。单个字母默认表示的都是列向量





列向量初始常用np.zeros((dim, 1))

20



24

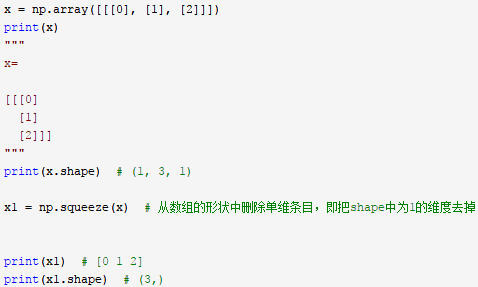
代价函数计算中，Y和A都是行向量，二者择一转置才能得到实数



np.squeeze()

删除矩阵中的维度为1的多余层。

如下[[[0] [1] [2]]]，就只是个简单的[0, 1, 2]，却又太多维度。所以用sqeeze去除。



30

画图最常用命令

