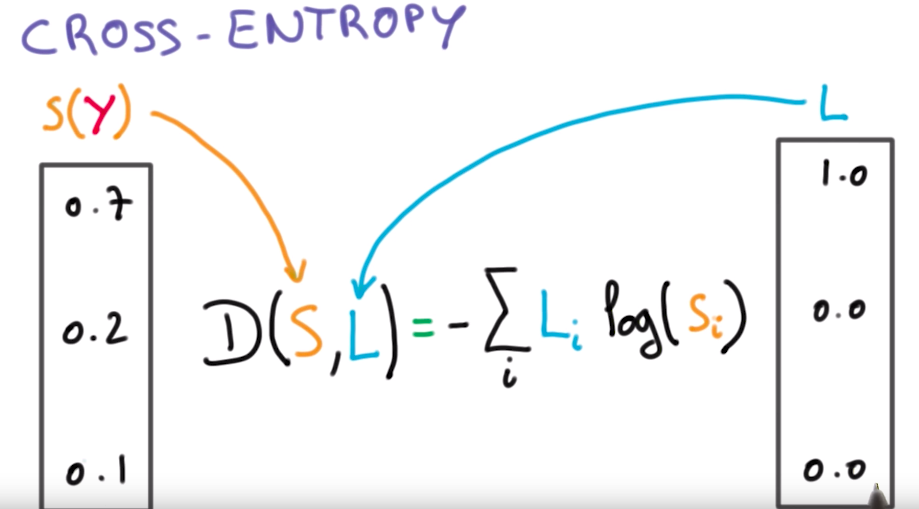
**多分类的原理（softmax和cross entropy）**

多分类问题，使用多维数组表示分类（类似onehot的表示形式）：

例如4分类：[1,0,0,0]、[0,1,0,0]、[0,0,1,0]、[0,0,0,1]

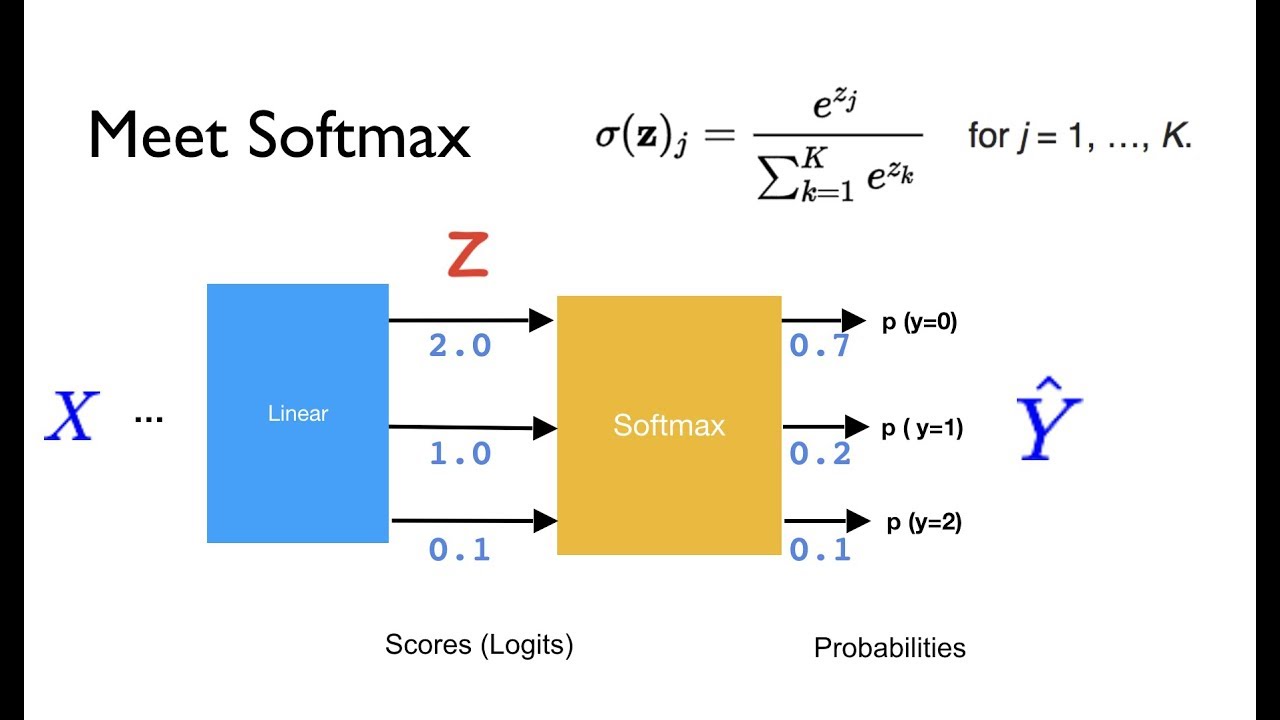
使用softmax输出为一系列的概率，如[0.3, 0.1, 0.45, 0.15]



使用cross-entropy（交叉熵）作为Loss函数。其中，S(y)表示softmax的输出预测，L表示label真实值。假如，softmax输出是[0.3, 0.1, 0.45, 0.15]，Label是4（[0,0,0,1]），根据公式，前面三个子式为零，只计算第四个元素log(0.15)。

交叉熵的形式像信息论中的信息熵：

**Softmax**



假设输入向量X(，为一个10维度的向量，经过Xw+b计算得到一个4维的输出（w为10\*4矩阵）。

得到。属于，通过softmax转换为

Normalize化的softmax

def softmax(x):

x -= np.max(x)

sum = np.sum(np.exp(x))

return np.exp(x) / sum