一种全局正则化的好方法

1. 取全部可训练的变量：

all\_vars = **tf.trainable\_variables()**

**tf.trainable\_variables()**函数可以取出整个计算图中的可训练变量(只要它的属性不是trainable=False，默认都是trainable=True)。

1. 选取需要正则化的变量：

selected\_vars = [v for v in all\_vars if ('fc' in v.name.lower()) and ('bias' not in v.name.lower())]

上面的规则我是取的全连接层(fc层)的非偏移(!bias)变量。

如果你想准确一点，建议print all\_vars，看看它们的名字。

1. 计算正则值(L2正则化)

l2\_penalty\_beta = 1e-4

l2\_penalty = l2\_penalty\_beta \* tf.add\_n([tf.nn.l2\_loss(v) for v in selected\_vars])

# add\_n()：累加。所以输出是一维标量。

tf .add\_n()函数功能是累加。所以输出的l2\_penalty是一维标量。这个算法，你可以随便调整。

1. 把它加入损失函数

with tf.name\_scope('loss'):

loss = cross\_entropy\_mean + **l2\_penalty**

加入了正则项后，最优的loss会趋近于1，而不是0.