**BP网络二分类**

1. **实验基本原理与目的**

通过反馈神经网络将数据进行二分类处理。

1. **数据的准备及数据预处理**

该实验使用pandas来处理数据。

给定数据集的每个样本的前0~2维特征为三个输入特征，3维为输出标签，由于是二分类，结果只有0和1。

1. 首先要对输入x进行规格化  
   将数据的前三维作为输入随机矢量x的值，并且对其规格化，使得每个分量的值分布在[-1,1]区间内。
2. 然后要对y进行规格化

由于打算用softmax做输出层激活函数，所以需要将输出y变为2维特征，比如原来n个样本点的y分别为

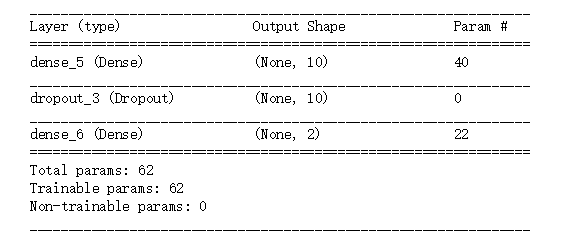
规格化后变为

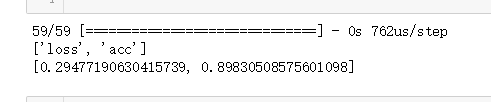
在该实验中上述过程都统一由scale函数完成，函数定义如下。

一共有291条数据，选择20%做测试集。

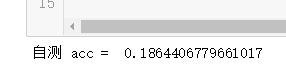
|  |
| --- |
| def scale(train, test):#对数据进行规格化，使得分布在[-1,1]内,shape 不变  # fit scaler  scaler = MinMaxScaler(feature\_range=(-1, 1))  scaler = scaler.fit(train)  # transform train  train = train.reshape(train.shape[0], train.shape[1])  train\_scaled = scaler.transform(train)  # transform test  test = test.reshape(test.shape[0], test.shape[1])  test\_scaled = scaler.transform(test)  return scaler, train\_scaled, test\_scaled |

1. **实验过程**

该实验通过keras建立了三层网络（包括Dropout层），输入层为2，第一层为10个神经元，输出层为2（二分类网络），通过库函数打印出网络结构如下  


1. **实验结果**该实验先用模型评估函数对loss和accuracy，得到结果如下图：  
   

后面在用model.predict函数将测试集输入，并与样本标签比对，得到的正确率如下



可见网络还是比较理想的。