要点：

1. 关于独热编码one-hot-encoder和标签编码label-encoder

用于离散型特征里面，一般是特征有多少维，就用多少维的vector来表示这个特征。

举例：该特征有三个离散的取值，A，B，C。用在独热编码可以表示为

A: [1, 0, 0] B: [0, 1, 0] C: [0, 0, 1] 。用于标签编码可以表示为A: 1 B: 2 C: 3。但是由于标签是有大小比较的，不适合让机器学习到这三个独立的特征取值有大小的区别，因此多采用独热编码。

1. 建立一个DNN网络

如果输入是image（2D）的话，首先要建立Flatten层将image抻平，这样就变成了一个(,m\*n)的输入。

>>>model.add(tf.keras.layers.Flatten(input\_shape=(m,n)))

然后再建立Dense层（全连接层），该层一共有N个neutron。

>>>model.add(tf.keras.layers.Dense(units=N, activation=’relu’))

注意的是：1）要在第一层定义input\_shape!

2）activation function一般多采用relu（不用sigmoid）

最后建立的Dense层需要转化独热编码的形式，也就是units应该等于特征分类总数，activation function =’softmax’即可。

1. 我的第二次实验做得稀烂

几个关于如何打印出梯度的参考

<https://zhuanlan.zhihu.com/p/64310188>

<https://stackoverflow.com/questions/51140950/how-to-obtain-the-gradients-in-keras>

<https://github.com/keras-team/keras/issues/2226>杰尼龟头像那个人说的挺好

<https://www.cnblogs.com/zhouyang209117/p/8311573.html>