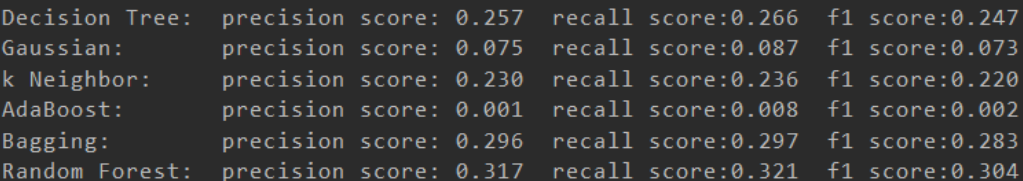
Hw4 相关说明

学号：1552635

姓名：胡嘉鑫

1. 问题阐述：

由于hw3中坐标点信息拥有时序的关系，而如果数据中的特征与时序产生某种关系的话，我们可以将这些特征拟合成特征随时间变化而变化的连续模型，这样只要生成了目标模型，就可以将预测的特征带入模型得到想要的预测结果。

 Hw3里我们使用了分类器对模型进行了分类训练， 使用较好的分类器预测出来的结果大概在0.2~0.3左右。

为了提升模型预测准确度，我们提出了一个设想：能否利用时间和特征可能存在的某种关系，去拟合出一个神经网络模型出来，使得预测结果的效果更好呢？

2. 使用库：

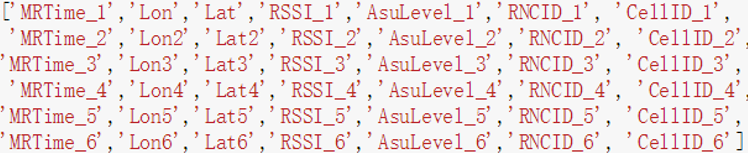
- Tensorflow

- Pandas

- Numpy

3. 操作步骤：

1） 将原有的data\_2g.csv数据文件提取出来， 结合goncan.csv生成7 \* 6 的DataFrame。之后将数据进行归一化处理，以便带入神经网络模型训练。之后，将DataFrame转化为 （928，6，7，1）的矩阵，矩阵的每个数字都是归一化后的0~1，

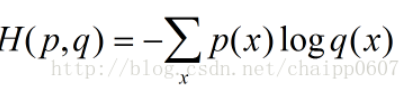


2）将data\_2g.csv 的数据利用hw3中的栅格ID转化算法进行转化，将每条数据的经纬度转化为栅格ID。其中要注意的是，由于神经网络模型最终跑出来的结果并不是如同 123 这样只有单一数字的栅格ID预测值，而是根据设定好的矩阵维数输出一个经过网络的矩阵出来；矩阵的每一行都有928个数，分别代表在相对应的index作为预测结果的可能性的值。在这里，有专门的one\_hot 概念（TensorFlow中有one\_hot转换函数），可以将单一维度的标签转化为上述矩阵形式，即one\_hot矩阵。

3）建立cnn模型。此处有几个地方需要声明：

- 由 卷积（激励） + 卷积（激励） + dropout + 全连接 层构成，其中dropout的设定是为了避免过拟合而在每次训练的过程中选取神经元的时候适当忽略一些神经元，这样每个神经元的相互独立性降低，预测的结果过拟合的现象得到缓解。

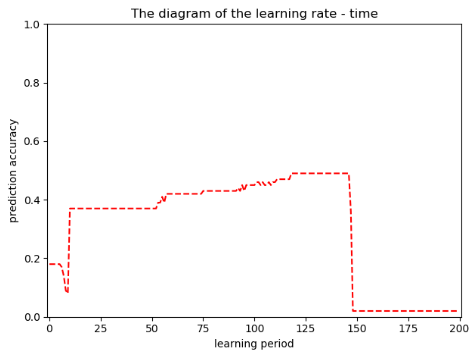
- 通过减小信息熵实现模型中神经元参数的调整。信息熵的计算公式为：



4）调整训练次数、学习率和滑窗大小，观察其对学习效率的影响。

4. 比较结果

1. 训练次数的影响：设置训练次数为200次，得到训练的准确率和训练次数之间的折线图关系：



可以看到，

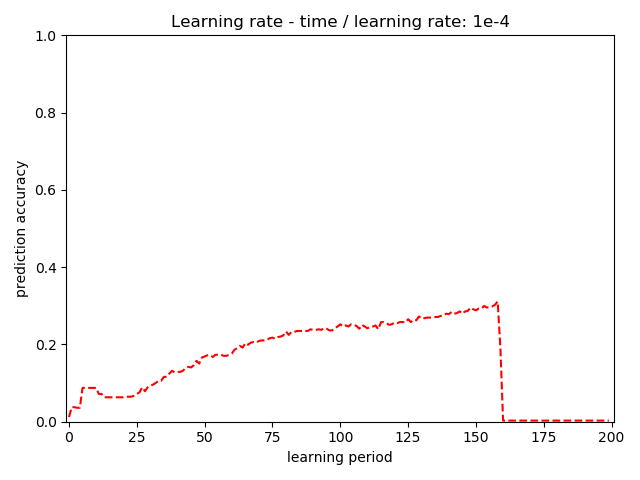
1）与分类器相比，神经网络训练出来的最好的模型预测准确率大概在40%，较之前有很大的提升

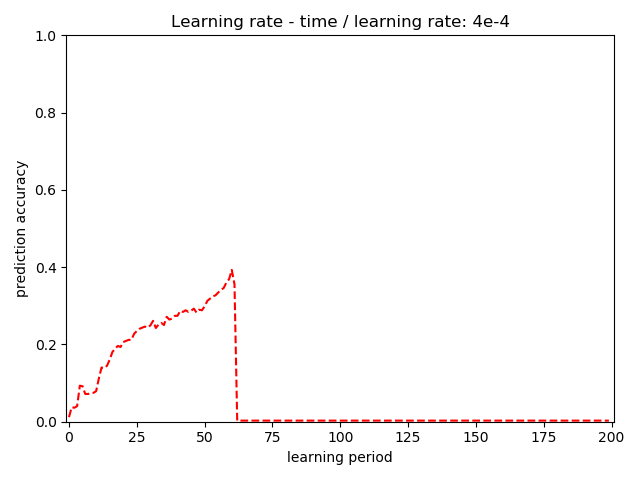
2）在训练150次的时候，训练的准确率会呈现断崖式的下跌，经过网上查证，发现可能是训练过拟合所导致的结果。提供的解决方案是：

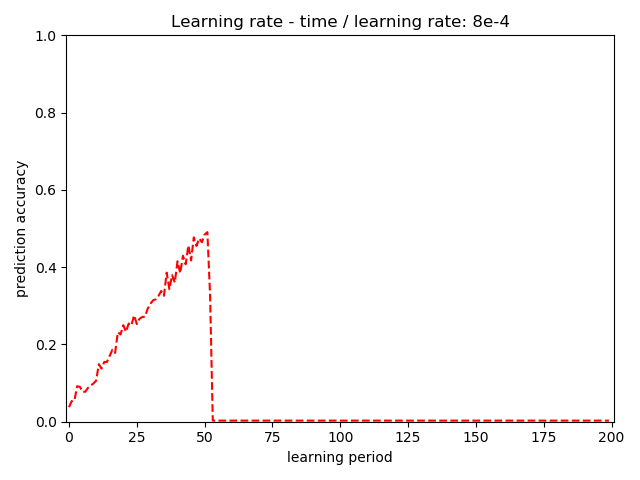
1）初期训练增大学习率，后期减小学习率

2）增加dropout减小过拟合（已采用）

2. 学习率影响：将学习率调整为1e-4 4e-4 8e-4 ，分别绘制预测准确率和训练次数的关系图：







可以根据三个图中的折线图看出：

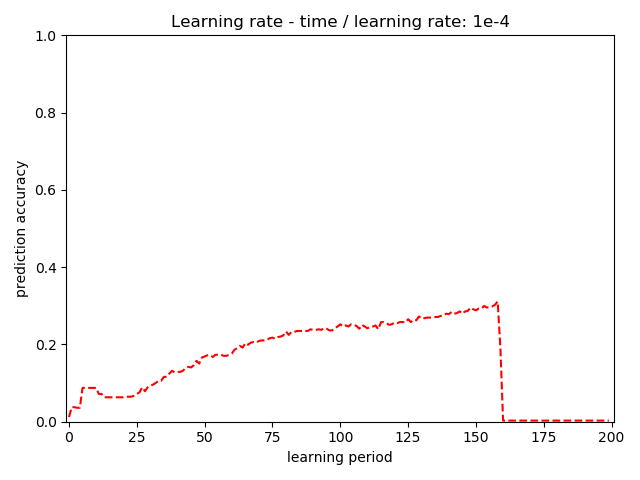
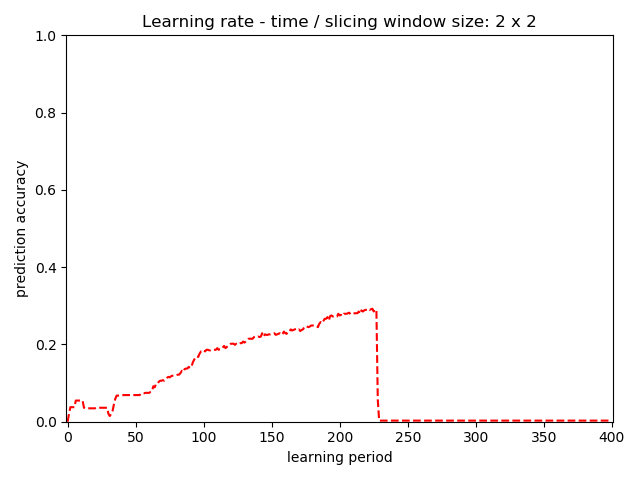
1）随着学习率的升高，模型学习的效率也升高（准确率提升速度提升）。

2）模型最优时的模型准确率大概在0.4 ，变化不大（模型最优值和训练的步长关系不大）

3）准确率到达0.4左右的时候会有突然的下降

3. 滑窗大小的影响：

取学习率为1e-4，滑窗为3 \* 3 和 2 \* 2 时的比较如下：



可见，滑窗对于学习效率的影响主要体现在学习时间上，滑窗越小学习效率越低。

综上，可以选取的较优的模型参数为：

* 训练次数：150
* 学习率：4e-4
* 滑窗：3 \* 3

4. 与hw3中分类算法的算法结果相比，神经网络在准确性上有一定的提升，但与图像识别的70% ~ 90%的准确率相比还是较差；结果表明特征点具有一定的连续性，但拟合成一维向量并不是很好的选择；时间和特征的相关性不是很强，至少在一维情况下的相关性并不理想。

后续的改进：可以更改损失函数，使得θ可以拟合成非线性的模型。