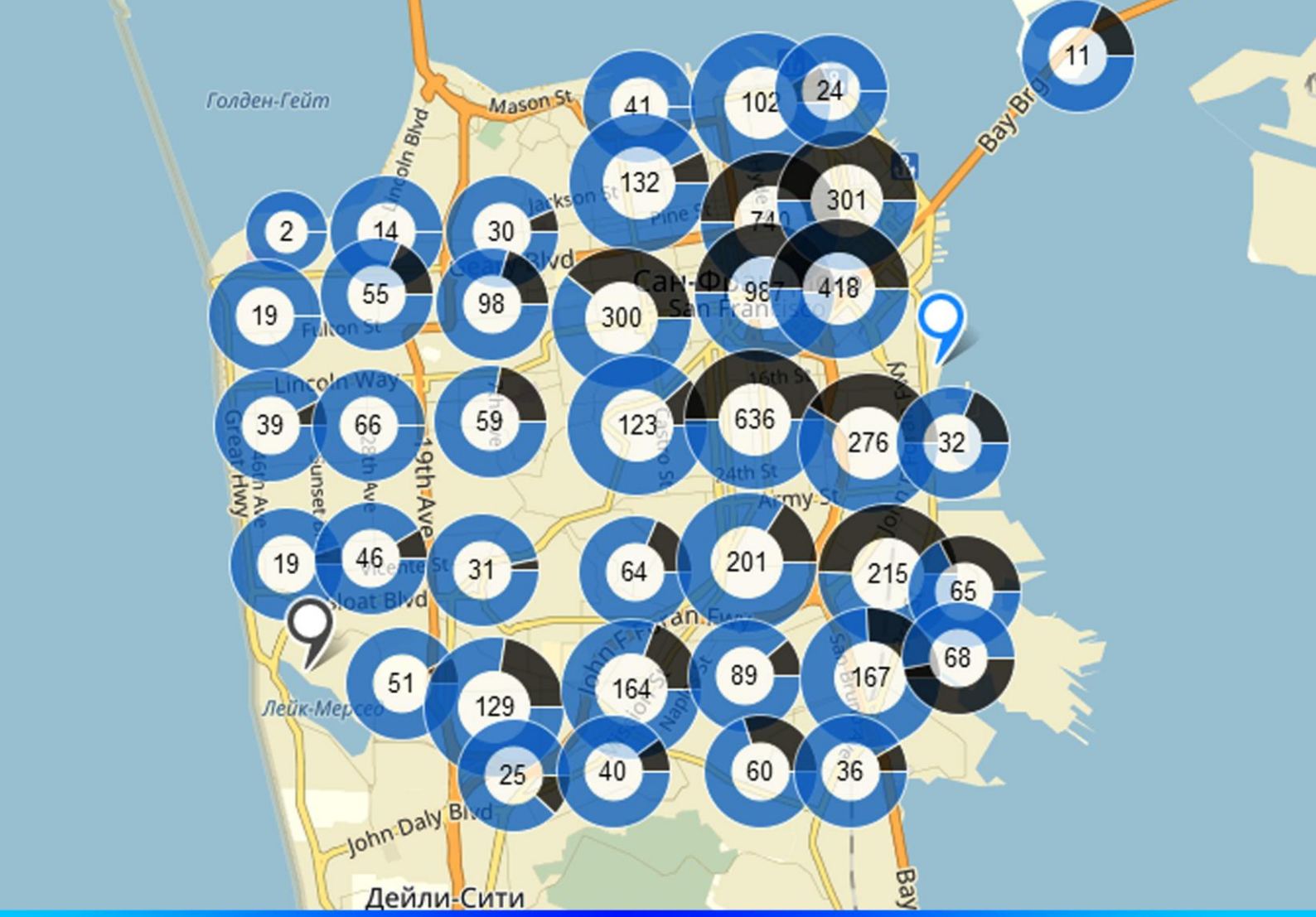


旧金山犯罪分类问题

《数据挖掘导论》项目总结

DM003

张作奇 郑沛霖

47th/2335

目录

一、解题思路

我们的解题思路如下：

1.预处理数据，对离群点做处理

2.提取特征，输出为 csv 文件

3.读取特征，尽可能尝试多的模型和参数，如神经网络、随机森林、Gradient Boosting、贝叶斯等，并在训练模型、调参的过程中修改特征，评估并增减特征

4.提交多个模型后选择最佳的几个模型进行投票式模型融合

二、虚拟机配置及环境

1.虚拟机配置

因为本地电脑显卡不适合做大规模计算，采用四台 Azure 虚拟机分别进行模型训练和调参，配置如下：

2.编程环境

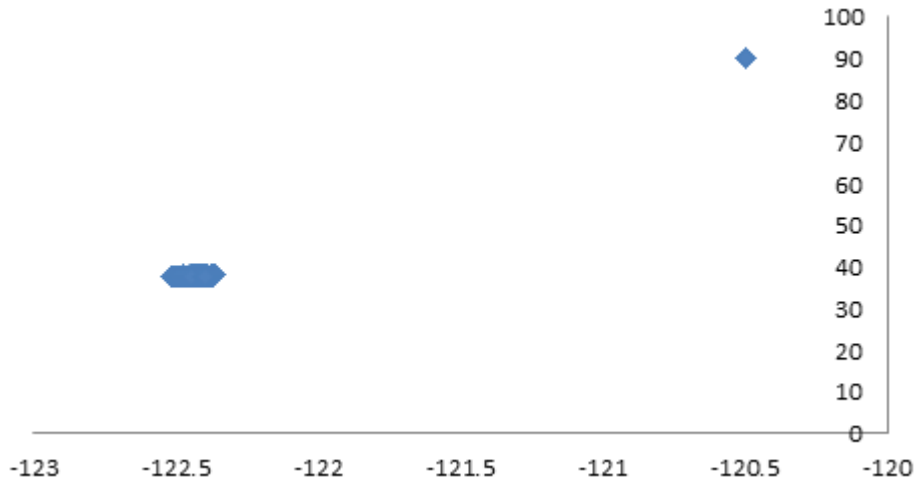
选择 Python3.4.4.2，并安装 Theano、sk-learn 等科学计算包

三、数据分析及预处理

1.数据分析

将 train.csv、 test.csv 的数据导入 Weka 与 Excel 进行预处理

查看 XY 得下图:



可见 XY 大部分集中在某一个区域 ,这与问题描述中的经纬度主要集中在旧金山相符合。

同时，在 X=120.5、Y=90 的区域存在离群点，数量大约在 60~100. 且 Address 列中有两万多个取值，数量过于庞大，不能直接转换为 01 取值。

2.处理方式

将 XY 标准化，将离群点的 XY 值转移到标准化的中心。

除了 Category 列外，train 中的与 test 无关的列则删除。

具体参见代码 feature.py

四、特征提取

1.常规读取和转化

读取时间转换为年、月、日的特征，

将 DayOfWeek 中的取值转换为取值为 0 或 1 的多维特征向量，

将 PdDistrict 中的取值转换为取值为 0 或 1 的多维特征向量，

将 XY 标准化并将离群点移到中心。

2.时间特征提取

昼夜：定义特征 awake，根据是否 8:00~18:00，取值为 0 或 1.

季节：定义特征 spring、summer、fall、winter，取值为 0 或 1.

判断圣诞节附近：定义特征 guonian，根据是否是 11-1 月，取值为 0 或 1.

3.地址特征提取

根据某个地址的样例数占总数的概率，以及特定地址的特定犯罪类型占地址样例数的概率，得到概率 p，通过 log(p)-log(1-p)的方式得到一个 39 维的特征向量 ,代表该地址。(此部分参考自 kaggle 论坛中的代码分享)

logodds0	logodds1	logodds2	……	logodds36	logodds37	logodds38
-6.36181	-2.34386	-7.67857	……	-2.32728	-2.63906	-4.62132
-6.36181	-2.34386	-7.67857	……	-2.32728	-2.63906	-4.62132
-6.36181	-2.34386	-7.67857	……	-2.72967	-2.9858	-4.62132
-6.36181	-2.91099	-7.67857	……	-3.4012	-3.92527	-4.62132
-6.36181	-2.34386	-7.67857	……	-2.24071	-2.9858	-4.62132
-6.36181	-1.88031	-7.67857	……	-1.75786	-2.9858	-4.62132
-6.36181	-2.34386	-7.67857	……	-2.72967	-2.9858	-4.62132

具体参见代码 feature.py

五、模型训练

1.神经网络•调参

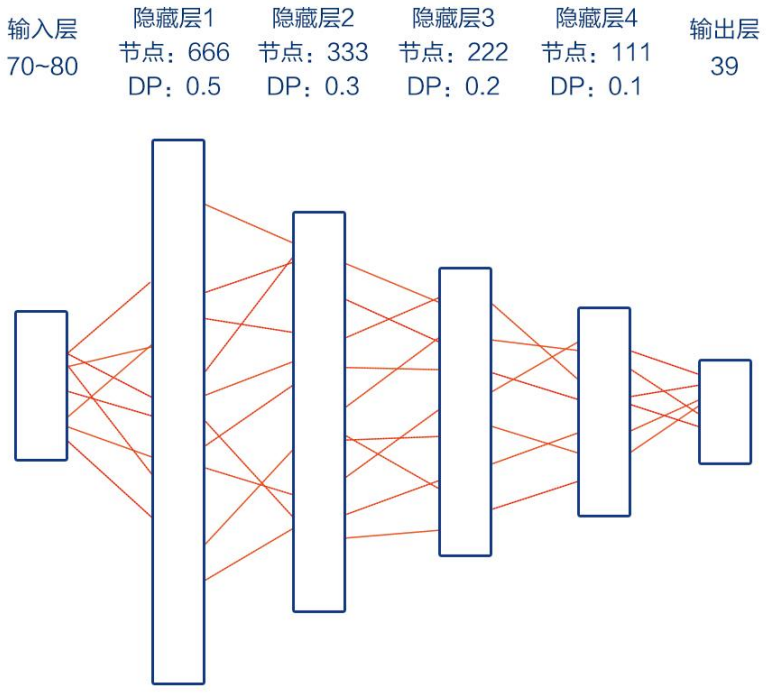
采用 theano+keras 的神经网络框架进行训练

(1) 调参尝试中的种种玄学

由于采用 keras 框架，因此比较方便地使用 adam 优化方式，基于一阶梯度来优化随机目标函数。

首先是从单层做起，多次调试之后到 500 个节点，DP 值为 0.5，已经是过拟合，尝试许多次，耗时巨大。

然后是多层的调参，尝试 2 层、3 层、4 层的结果，不断优化，得出 DP 值为逐步下降的结果较好，最优的多层节点如下图示：



(2)最终结果如下：

Kaggle_Loss	层数	每层节点	代数	DP
2.22625	4	666-333-222-111	50	0.5-0.3-0.2-0.1
2.22659	1	500	80	0.5
2.22675	1	500	80	0.5
2.2271	1	500	80	0.4
2.22761	1	500	50	0.5
2.22803	1	700	80	0.5
2.23006	1	500	80	0.5
2.23096	1	500	50	0.6
2.23169	1	500	35	0.5
2.23196	1	500	30	0.5
2.23267	1	500	55	0.3
2.23337	1	500	65	0.5
2.23351	1	1000	100	0.5
2.23602	1	500	80	0.3
2.23677	3	1000-500-100	50	0.5-0.2-0.1
2.23777	1	500	80	0.7
2.23788	1	300	20	0.5
2.23977	3	500-200-100	50	0.5-0.3-0.1
2.24046	1	500	10	0.3
2.24167	2	260-260	20	0.5-0.5

2.2436	1	45	100	0.1
2.24657	1	128	50	0.5
2.24814	1	45	30	0.1
2.25441	1	45	100	0.3
2.28252	1	45	100	0.5

2.逻辑回归•特征修改

使用 sklearn.linear_model 框架中的 LogisticRegression

耗时比神经网络少很多，但是结果不够拟合

（1）第一次跑出来的结果为

submission.zip 2.30721

（2）此时认为是特征不足，增加时间特征后结果为

logi.zip 2.27480

3.随机森林•多次算数平均

采用 sklearn 的 RandomForestClassifier 框架

依照下列提交结果可见随森林规模增加而精确度增加

rf_200.csv 3.37669

rf_100.zip	3.44244
rf30.zip	5.53609
rf_10.zip	12.26864

但是在 200 棵树以上时，出现内存不足现象，于是尝试多个 200 棵树的随机森林，并对结果取算数平均，结果如下：

200x10.csv	2.57464

可见 10 个 200 棵树的森林的算数平均值有较好效果 ,因为时间关系未往下继续测试。

4. GradientBoostingClassifier•调参

采用 sklearn 的 GradientBoostingClassifier 框架

当有 100 个分类器，默认深度为 3 时，结果如下：

GradientBoostingClassifier_100.zip	2.26498

此时增加深度到 5，结果如下：

GradientBoostingClassifier_100_5.zip	2.29785

增加弱分类器的精度却反而错误增加，应该是在模型中出现了过拟合，鉴于提升的空间不大没有进一步测试。

5.贝叶斯

在 sklearn 里的 naïve bayes 中的多个模型进行尝试

在 GaussianNB、MultinomialNB、BernoulliNB 得到的对训练集的拟合效果约在 3~7 左右，很不理想，因而没有做进一步的调整和提交。

模型的训练详见 train.py

六、模型融合

1.选择几个较独立的较好模型

在所有的结果中，以下几个模型效果较好：

增加特征的逻辑回归模型、

隐藏层 4 个，节点分别为 666、333、222、111、dp 值为 0.5、0.3、0.2、0.1 的神经网络

隐藏层 1、节点 500 个、dp 值 0.5、80 代的神经网络

在 kaggle 中的得分依次为：2.27480、2.22625、2.22675

2.对结果进行投票后融合

在三个模型的结果中，对每一行取距离最短的两个，作算数平均。

尝试过几何平均、加权平均，效果均不理想。

不选择三个最佳得分的模型是因为单层神经网络间过于接近，选择较独立的模型有助于提升在投票中排除错误解的概率。

采用 numpy，详见 sub.py

七、遇到困难及解决办法

1. **困难：本地无法较快的做大规模运算调参**

   解决：借用朋友的四台 Azure 四核+28G 内存虚拟机解决，但是仍没有专业显卡快速。

2. **困难：某些科学计算 python 包在 linux 下的 python3 不支持**

   解决：使用 windows 下的 python3.4

3. **困难：模型出现过拟合**

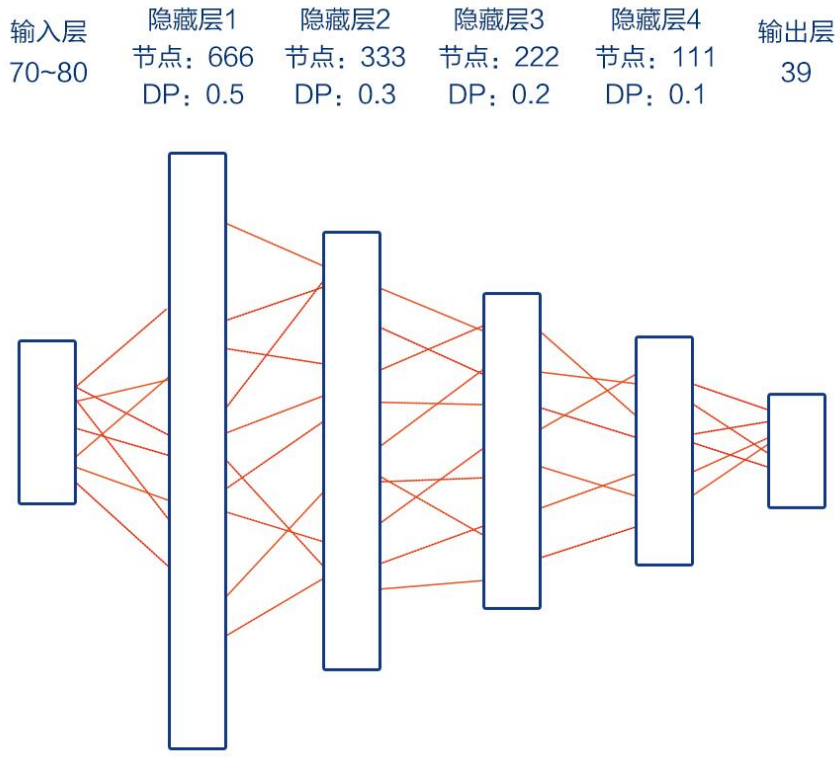
   解决：神经网络中调整 DP 值、增加到多层，初始化 init 调整其他模型中采用增减部分特征，减少单次精度多次训练取融合结果等方式解决。

4. **困难：神经网络单个模型调参到极限，耗时久仍没有突破**

   解决：加上对课程得分的考虑，尝试其他模型，作投票融合。

八、结果分析及总结

1.在主攻的模型神经网络中，如果按照一般的经验公式确定隐藏层节点数是（输入输出的几何平均+1~10 的整数），但是在这个样本中得出的结果却是隐藏层节点在 500~1000 左右 最好，如下图：



究其原因，**可能是在地址特征的提取中，为了降低内存消耗，把两万多维的地址提取成 39 维的特征向量，因此某种意义上输入层的维度应该远大于目前的值，因此隐藏层节点的数量是可以接受的。**

2.如果想要在得分上进一步提升，那么在模型调参上应该已经没有多大的进步空间，**可考虑变换各个特征的权重进行模型的训练**，也可能是特征的提取仍不足。

3.在本次课程项目中，**我们小组毫无数据挖掘比赛的经验，从零开始，能达到【2.22185，47/2335】的结果已经比较满意，而且我们小组尝试了多种模型，实践了多种特征提取和数据处理的方法**，已经基本达到了预期，收获颇丰。

九、参考资料

1.kaggle 论坛：https://www.kaggle.com/c/sf-crime/forums

2.sklearn 文档：http://scikit-learn.org/stable/

3.keras 文档：http://keras.io/

4.numpy 文档：http://www.numpy.org/

5.pandas 文档：http://pandas.pydata.org/