机器学习与模式识别课程报告

题目：应用分类算法判断交通事故严重程度

姓　　名：王 倩 妮

班　　级：交通 2015-02 班

学　　号：2015112956

任课教师：郝 莉

2018 年 4 月 22 日

目 录

一、问题背景

随着机动化时代的到来，小汽车作为一种通勤工具，为人们的生产、生活带来巨大便利，但与此同时，机动车交通事故每年也对国家造成巨大经济损失。道路交通事故的发生有外因、内因两大部分，其中外因主要有如：天气、路况、交通设计合理性、机动车性能等，内因主要是与驾驶员的操作-反映有着密切关系。而事故发生时，内外因的状况、程度，与交通事故的严重程度也有着密切的关系。本分类问题着眼于事故数据，通过事故发生时外因数据，对事故的严重程度进行分类，有利于基于特征数据（检测器检测）的事故预警，对于减少交通事故的损失有着积极作用。

二、数据描述与可视化初探

2.1 数据描述

- 数据来源：https://www.kaggle.com/daveianhickey/2000-16-traffic-flow-england-scotland-wales/data
- 数据集名称：1.6 million UK traffic accidents
- 数据特征项

表 1

特征项	含义	数据类型	是否有空值
Accident_index	事故编号	String	False
Location_Easting_OSGR	英国本地坐标系 x 坐标	Numeric	True
Location_Northing_OSGR	英国本地坐标系 y 坐标	Numeric	True
Longitude	经度	Numeric	True
Latitude	纬度	Numeric	True
Police_Force	不详	Numeric	False
Accident_Severity	事故严重程度	Numeric	False
Number_of_Vehicles	涉事车辆数	Numeric	False
Number_of_Casualties	伤亡数	Numeric	False
Date	日期 dd/mm/yyyy	DateTime	False
Day_of_Week	星期几	Numeric	False
Time	时间 hh:mm	DateTime	False
Local_Authority_(District)	不详	Numeric	False
Local_Authority_(Highway)	不详	String	False
1st_Road_Class	不详，与交叉口相关	Numeric	False
1st_Road_Number	不详，与交叉口相关	Numeric	False

Road_Type	道路类型	String	False
Speed_limit	限速情况	Numeric	False
Junction_Detail	交叉口类型	String	True
Junction_Control	交叉口控制	String	True
2nd_Road_Class	不详，与交叉口相关	Numeric	False
2nd_Road_Number	不详，与交叉口相关	Numeric	False
Pedestrian_Crossing-Human_Control	有无行人控制	String	True
Pedestrian_Crossing-Physical_Facilities	行人控制设施类型	String	True
Light_Conditions	照明状况	String	False
Weather_Conditions	天气状况	String	True
Road_Surface_Conditions	路表面状况	String	True
Special_Conditions_at_Site	有何特殊情况	String	True
Carriageway_Hazards	道路上其他物体/事件	String	True
Urban_or_Rural_Area	城区 or 郊区	Numeric	False
Did_Police_Officer_Attend_Scene_of_Accident	警察是否到达现场	String	True
LSOA_of_Accident_Location	不详	String	True
Year	年份	Numeric	False

- 数据覆盖年份：2005 年-2014 年

2.2 数据可视化

数据可视化过程有利于帮助我们初步了解数据集的整体特征，以之为启发进行机器学习。如下图所示是按年份统计的事故数量，由条形图可见，从 2005 年至 2014 年，事故数总体呈下降趋势，但 2012 年存在上升。

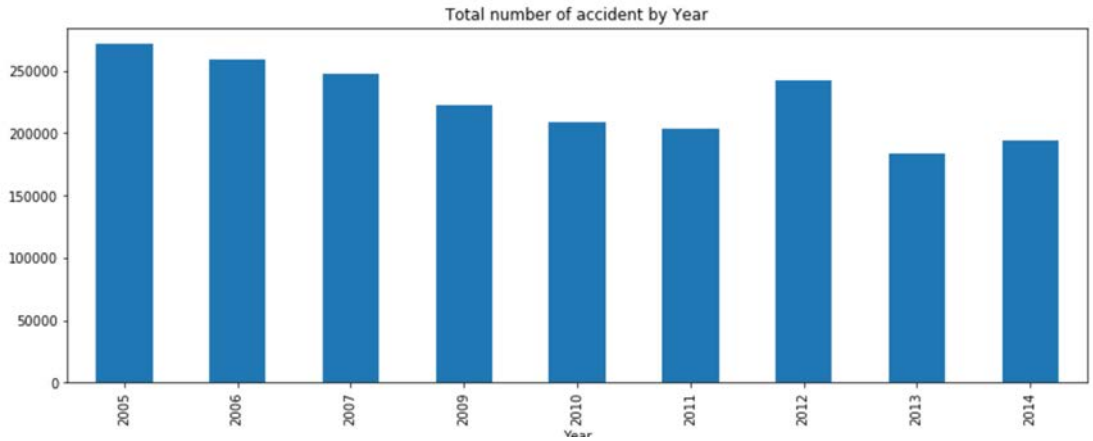

图 1 Total number of accident by year

按年份归类，以月份为横轴绘制得到的不同年份事故伤亡数变化曲线，由下图可知，一年内每月事故伤亡数变化存在波动，但总体在 1-4 月处于较低水平，此后呈现波动上升趋势至 11 月，11-12 月年末再次下降。

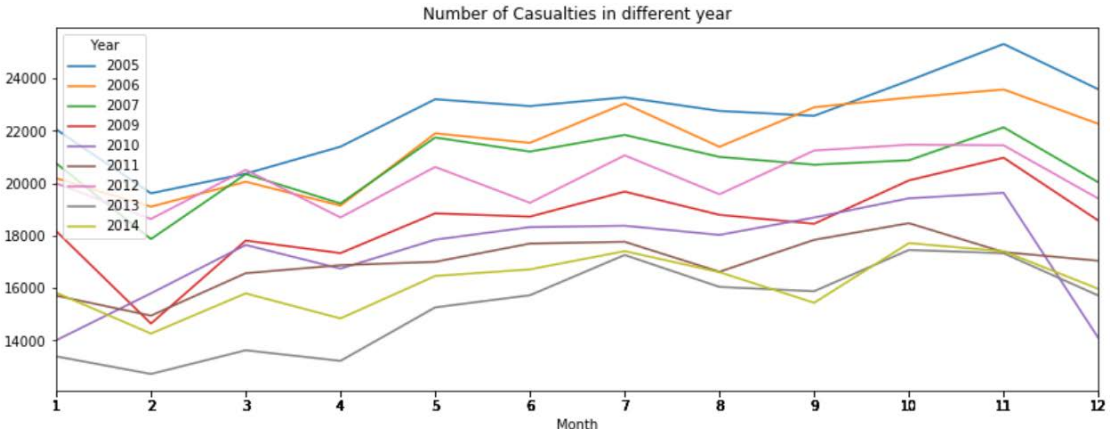

图 2 Number of Casualties in different year

每个事故具有着不同的伤亡数与涉事车辆数，通过以频率分布直方图的形式可以清楚看到统计事故中，以伤亡 1-2 人为主，事故涉事车辆以 2 辆居多。高伤亡数与高涉事车辆数的事故并不多见，但通过分析可知伤亡数从 1-93 人不等，涉事车辆数从 1-67 人不等。

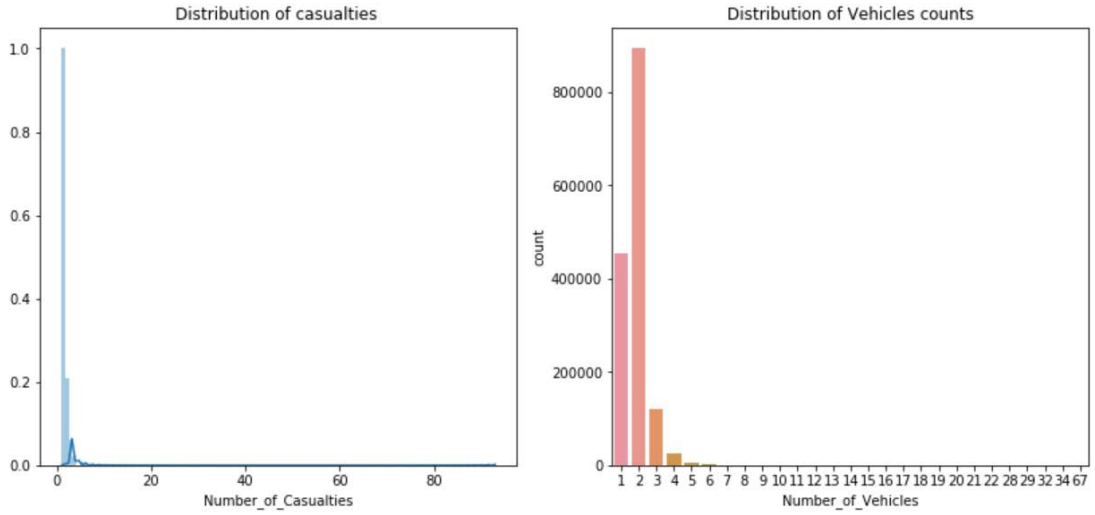

图 3 Distribution of casualties and Vehicles counts

事故的发生与时间有着密切的关系，时间对于事故的发生具有着间接影响，由下图可见，分别按照每日 24 小时、每周 7 天、每年 12 个月进行伤亡数分组统计结果。每日 24 小时、每年 12 个月的影响主要体现在不同时段的驾驶员活动不同，外部环境条件对于驾驶的影响不同；而每周 7 天的差异主要体现在驾驶员的活动、心理层面。

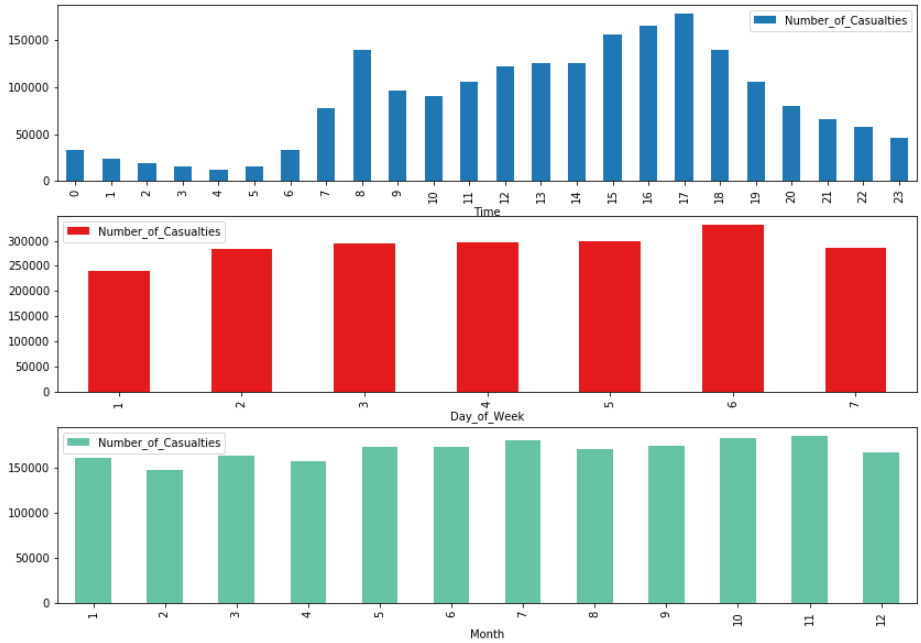

图 4 Number of Casualties in different time

三、特征选取与数据预处理

3.1 特征选取

本次的研究目的为：通过事故发生时外因数据，对事故的严重程度进行分类。又因为此数据集特征项较多，因此在进行机器学习过程前，需要选取合适的用以区分事故严重程度的外因数据。

首先，主观选择 Accident_Severity 作为分类学习的标签项。统计数据量可知，数据集中大部分数据为事故严重程度为“严重”，部分为“正常”，只有少部分数据属于“轻微”。

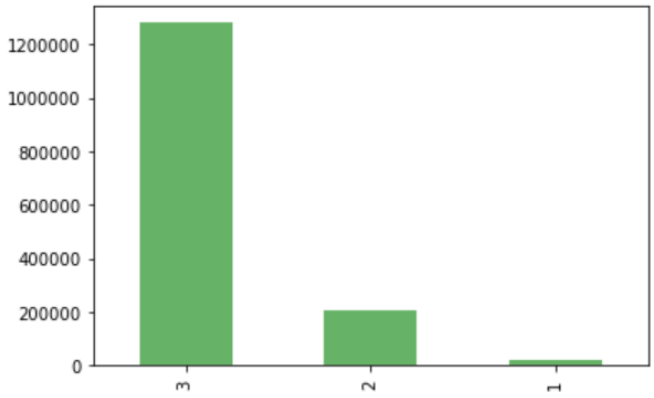

图 5 标签数量统计

然后，考虑与事故的严重程度相关性较大的因素，主观选取以下特征项作为学习的特征：

- Time：通过以上可视化的分析不难看出，事故的发生具有一定的时间差异性，

时间通过间接影响如环境条件等因素，对事故的发生产生影响。

- Light_Conditions：照明条件与事故的发生有着密切联系，照明不好的路段容易发生交通事故。
- Weather_Conditions：天气条件影响了能见度、车辆性能表现，间接影响路面条件。不良的天气条件易导致事故的发生。
- Speed_limit：速度与交通事故的发生存在一定联系。通常认为，在高速状态下，制动时间、距离增长，易发生事故。因此一些路段会进行限速控制。
- Road_Surface_Conditions：路面状况的改变与交通事故的发生有着密切的关系，当路面条件变化时，摩擦系数改变，制动距离因此产生改变。
- Special_Conditions_at_Site：此项指事故发生时是否有其他特殊情况与具体特殊情况的类型。在道路上特殊情况发生时，更易发生交通事故。
- Number_of_Vehicles：此项指事故发生时的涉事车辆数，一般涉事车辆数越多事故越严重。
- Number_of_Casualties：此项指事故造成的伤亡数，一般伤亡数越多，事故越严重。

为初步验证以上因素与事故严重程度间的关联性，以 Time 为例进行验证。

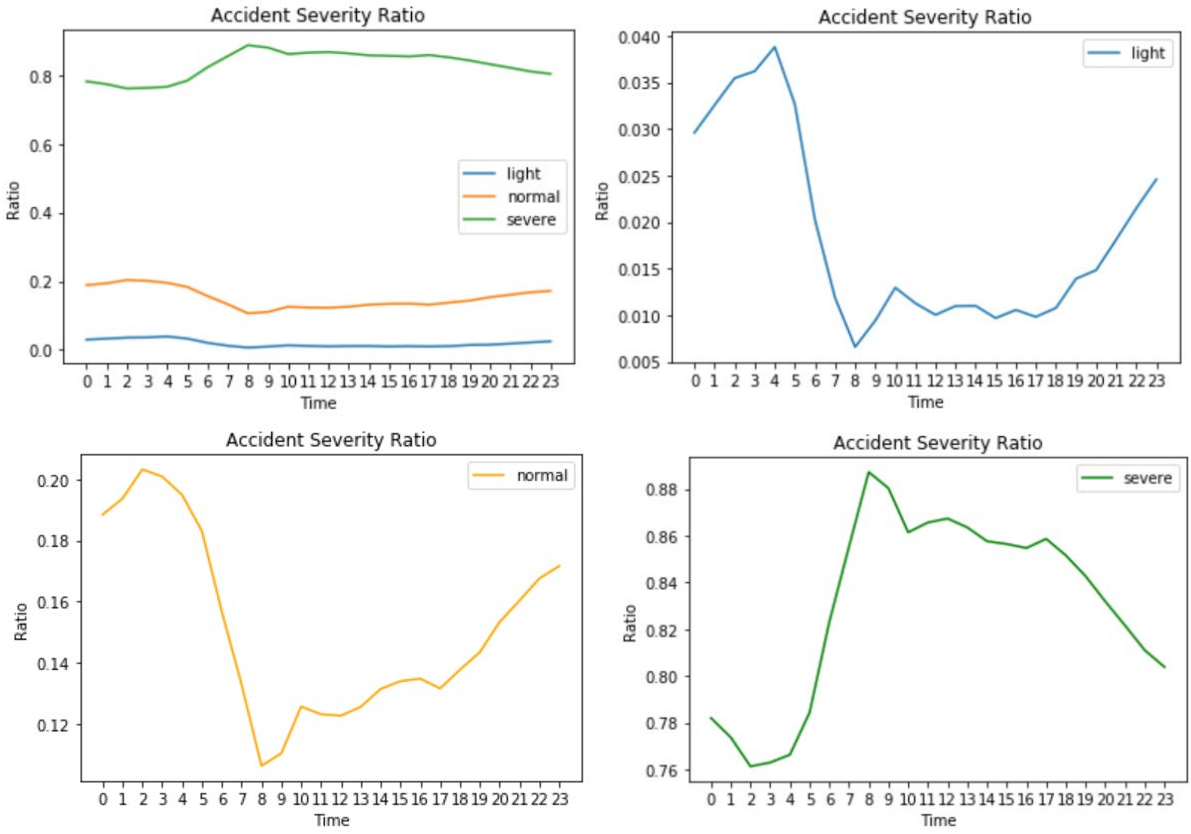

图 6 事故严重程度占比随时间变化图

以时间为因素进行分组，由于不同时段内的事故数存在差异，为避免这种差异性带来的影响，使用每一小时内的轻微、正常、严重的事故数与该小时内事故总数的比值刻画不同严重程度事故的占比状况，以此来判断事故严重程度与时间两者间的关系。

其余要素分析方式相似，此处不再赘述。

3.2 数据预处理

3.2.1 去除空值

提取用以进行学习的特征数据，组成新的 DataFrame。由于部分所选特征项存在空值，因此需要通过一定手段进行补全。通过统计每个特征选项特征项数目，发现空值数据占比较小，且考虑到事故之间较为孤立，不易于进行数据补全，因此此处我们将新 DataFrame 中存在空值得条目进行去除，保留特征项完整的数据。

3.3 特征值匹配

针对选取的特征值，大多是标称型的文字数据，部分分类算法无法进行运算，因此需要通过一定对应关系将标称型数据转化为文字型数据。

转化思路以直接转化与分段函数转化为主。对于类别较少的标称型数据可直接对应进行转化，对于类别较多的数值型数据，可以采用分段函数的方法进行对应转化。

表 2

序号	特征名称	特征值	匹配结果
1	Time	0-5	1
		6	2
		7-17	3
		18-23	2
2	Light_Conditions	Daylight: Street light present	1
		Darkness: Street lights present and lit	2
		Darkeness: No street lighting	5
		Darkness: Street lighting unknown	3
		Darkness: Street lights present but unlit	5
3	Weather_Conditions	Fine without high winds	1
		Fine with high winds	2
		Raining without high winds	2
		Raining with high winds	3
		Snowing without high winds	4
		Snowing with high winds	5
		Darkeness: No street lighting	5
		Unknown	2
		Other	2
4	Speed_limit	10;15;20	1
		30;40	2
		50;60;70	3
5	Road_Surface_Conditions	Dry	1

		Wet/Damp	2
		Frost/Ice	4
		Snow	3
		Flood (Over 3cm of water)	3
6	Special_Conditions_at_Site	None	0
		Roadworks	1
		01 or diesel	1
		Mud	2
		Road surface defective	2
		Auto traffic signal out	3
		Permanent sign or marking defective or obscured	3
		Auto traffic signal partly defective	2
7	Number_of_Vehicles	1	1
		2	2
		3	3
		4-5	4
		6-10	5
		>10	6
8	Number_of_Casualties	1	1
		2	2
		3	3
		4	4
		5-6	5
		7-10	6
		>10	7

四、使用分类算法进行分类

4.1 KNN 算法及实现

4.1.1 KNN 算法原理

在训练样本集中每个数据都存在标签，即我们知道样本集中每一数据与所属分类的对应关系。输人没有标签的新数据后，将新数据的每个特征与样本集中数据对应的特征进行比较，然后算法提取样本集中特征最相似数据（最近邻）的分类标签。选择 k 个最相似数据中出现次数最多的分类，作为新数据的分类。在 KNN 中，通过计算对象间距离作为各个对象之间的相似性指标，代替对象之间的匹配度计算。

对于训练样本数为 m，特征数为 n 的训练样本集，计算测试样本 x 与 m 个训练样

本的欧氏距离

$$d(x, y_i) = \sqrt{\sum_{j=1}^{n} (x_j - y_j)^2}$$

其中 i=1，2，……，m。对 $d(x, y_i)$ 进行降序排列，选择前 k 个值，出现次数最多的分类作为测试样本 x 的分类。

4.1.2 算法优缺点

- 优点：

1. 简单有效，容易理解和实现；
2. 重新训练的代价较低（类别体系的变化和训练集的变化）；
3. 计算时间和空间线性于训练集的规模；
4. 错误率渐进收敛于贝叶斯错误率，可作为贝叶斯的近似；
5. 适合处理多模分类和多标签分类问题；
6. 对于类域的交叉或重叠较多的待分类样本集较为适合；

- 缺点：

1. 是懒散学习方法，比一些积极学习的算法要慢；
2. 计算量比较大，需对样本点进行剪辑；
3. 对于样本不平衡的数据集效果不佳，可采用加权投票法改进；
4. k 值的选择对分类效果有很大影响，较小的话对噪声敏感，需估计最佳 k 值。
5. 可解释性不强，计算量大。

4.1.3 算法实现

由于本次数据集较大，而 KNN 算法时间复杂度较高，调用课本代码由于使用 for 循环数量较多，造成运行缓慢。因此调用 scikit-learn 机器学习库实现 KNN 算法。

Scikit-learn 的通用使用步骤为：

① 引入模块
② 定义分类器
③ 利用分类器、训练数据进行 fit
④ 加入验证数据输出 predict
⑤ 比较验证数据集标签与 predict 结果，计算准确率

以 KNN 算法为例，其使用方法如下所示：

```
from sklearn.neighbors import KNeighborsClassifier
neigh = KNeighborsClassifier(n_neighbors=5)
neigh.fit(data_xnew,data_ynew)
preuseknn=neigh.predict(Data_xnew)
```

4.2 朴素贝叶斯算法及实现

4.2.1 朴素贝叶斯算法原理

朴素贝叶斯是一种简单但是非常强大的线性分类器。它在垃圾邮件分类，疾病诊断中都取得了很大的成功。朴素贝叶斯算法的核心是朴素贝叶斯公式：

$$\mathrm{P(B|A)} = \frac{P(a|b)P(B)}{P(A)}$$

对于机器学习中的朴素贝叶斯算法，可将贝叶斯公式转化为下式：

$$\mathrm{P}\big(\text{类别}\big|\text{特征}\big) = \frac{P\big(\text{特征}|\text{类别}\big)P(\text{类别})}{P(\text{特征})}$$

$$\mathrm{P}\big(\text{后验}\big) \propto \mathrm{P}\big(\text{先验}\big) * \mathrm{P}(\text{似然})$$

因此，其核心思想：选择具有最高后验概率作为确定类别的指标。

它只所以称为“朴素”，是因为它假设特征之间是相互独立的，但是在现实生活中，这种假设基本上是不成立的。那么即使是在假设不成立的条件下，它依然表现的很好，尤其是在小规模样本的情况下。但是，如果每个特征之间有很强的关联性和非线性的分类问题会导致朴素贝叶斯模型有很差的分类效果。

4.2.2 算法优缺点

- 优点：

1. 数学基础坚实，分类效率稳定，容易解释；
2. 所需估计的参数很少，对缺失数据不太敏感；
3. 无需复杂的迭代求解框架，适用于规模巨大的数据集。

- 缺点：

1. 属性之间的独立性假设往往不成立（可考虑用聚类算法先将相关性较大的属性进行聚类）；
2. 需要知道先验概率，分类决策存在错误率。

4.2.3 算法实现

高斯朴素贝叶斯分类器是朴素贝叶斯分类器的一种，此外还有多项式分布（主要适用于文本分类）、伯努利分布。此处调用 scikit-learn 中高斯朴素贝叶斯分类器：

```
from sklearn.naive_bayes import GaussianNB
clf = GaussianNB()
clf.fit(data_xnew,data_ynew)
predictusenb = clf.predict(Data_xnew)
```

五、测试方法与结果

5.1 测试方法

首先，对数据划分训练集与验证集，比例为 70%训练数据，30%验证数据，选取方式为随机选取。

由于效果不理想，进行了以下几种尝试：

- 方法一：经“三”部分处理过后的数据，与原计划的标签项、特征项。
- 方法二：经“三”部分处理后的数据，Accident_Severity 的 1、2、3 等级比例大致相同数据，数据随机选取，与原计划的标签项、特征项。
- 方法三：经“三”部分处理过后的数据，选取“Number_of _Casualties”为标签，改变其 map 的方式，其余为特征项。
- 方法四：经“三”部分处理过后的数据，选取“Number_of _Casualties”为标签，改变其 map 的方式，其余为特征项，选取 Number_of _Casualties 各等级比例大致相同数据。

由于几种测试方式效果均不理想，因此，希望可以与大家共同探讨结果不理想的原因。在测试结果中，也会进行一定的分析。

5.2 测试结果

通过测试发现应用 KNN 方法与应用朴素贝叶斯方法，准确率差距很小，差距在 3%以内。

5.2.1 方法一

- 准确率结果：

表 3

类型	准确率
总体（朴素贝叶斯）	0.8301030074955604
Accident_Severity=3	0.9650133143775841
Accident_Severity=2	0.021206679834799784
Accident_Severity=1	0.09622563251762754

- 原因分析：

若关注整体准确率，则本问题整体准确率较高，但具体分析每个分类的具体准确率时发现，3 类准确率很高，2、1 类准确率很低。分析原因，这主要与不同类别的数据量有关。本问题中，3 类数据明显偏多，且与 1 类相差两个数量级。因此提出方法二，筛选部分数据用于训练。

5.2.2 方法二

- 准确率结果：

考虑数量最少类型数据的条数，随机选取 3 种类型数据各 19000 条数据用于训练。

表 4

类型	准确率
总体（朴素贝叶斯）	0.4456140350877193
总体（KNN）	0.43871345029239767
Accident_Severity=3	0.7385146804835924
Accident_Severity=2	0.16820276497695852
Accident_Severity=1	0.4225476358503881

- 原因分析

选取原方案特征、标签，选取规模相同的数据，测试结果表明准确率很低。由于观察数据发现，严重程度‘3’的数据最多，但由伤亡数和涉事车辆数看，大部分事故应该均处于一般等级或较低等级。考虑于此，可能出于 Accident_Severity 的定义问题，决定在方法三中更换‘伤亡者人数’作为标签，其余数据作为特征。

在本步骤中，通过尝试调整 3 种类型数据量（如换作：30000,40000,19000）对准确率结果均有较大影响。

5.2.3 方法三

- 准确率结果

将“伤亡数”按照=1；=2、3；>3 化为三类。应用清洗后的全部数据，可得准确率情况如下表所示。

表 5

类型	准确率
总体（朴素贝叶斯）	0.7263147580702629
Number_of_Casualties>3	0.12410995197880444
Number_of_Casualties=2、3	0.12555868952050336
Number_of_Casualties=1	0.9080186079965661

- 原因分析

此方法依旧造成数据量差距悬殊的问题，因此方法四中采用与方法二类似思路进行尝试。

5.2.4 方法四

- 准确率结果

考虑数量最少类型数据的条数，随机选取 3 种类型数据各 40000 条数据用于训练。

表 6

类型	准确率
总体（朴素贝叶斯）	0.459113902940686
总体（KNN）	0.43740619267330033
Number_of_Casualties>3	0.42218172782241364
Number_of_Casualties=2、3	0.2638738212467662
Number_of_Casualties=1	0.6927528938097635

5.3 准确率影响因素

虽然本次分类算法没有取得令人满意的结果，但总结以上经验可知，机器学习的准确率与很多因素有关。

- 准确率与数据集数据质与量

由以上测试，我们可以明显看出使用数据量的差异对于结果有着巨大影响。同时数据本身的质的情况也会对于结果存在影响。

- 准确率与特征选择

真实数据中，采集项目较多，但应用机器学习算法并非特征越多越好，选取合适的特征对于结果的提升有着重要作用，本次虽然在特征选取上进行了一定程度的解释，但还需更有力的证明与科学的选择方法。

- 准确率与特征匹配

交通事故数据中很多数据是以文本形式存储的，针对这些数据，在应用算法之前，按照主观认定进行匹配，可能对结果产生了影响。如何标定特征值也需要进行更加深入的探讨。

- 准确率与应用算法

本次主要利用 KNN 与朴素贝叶斯两种分类算法对于交通事故数据进行学习，二者的结果差异不大，但运行速度上有较大差异。此外，算法自身参数对于准确率也存在影响，如 KNN 中 k 值得选取，经过多次尝试，本问题中 k 取 9 效果较好。

六、总结与心得体会

本次利用 Kaggle 平台英国 2005-2014 交通事故数据集进行分类算法的练习，感触颇深。真实数据往往统计项目较多且易存在异常数据，对于特征的选取与数据的清洗对于问题的解决至关重要。近年来，有很多学者基于机器学习算法对于交通安全问题进行研究，对于及时做好预警，减少事故的发生有着积极作用。通过练习，我的编程能力、数据分析能力进一步得到提高，加深了对于分类算法的理解，未来也希望学习更多的机器学习方法，结合数据对于交通问题的原因与解决策略进行更加深入的探讨。本次练习

并未利用地理信息的空间数据（如 GPS 信息），此后如果能结合 GIS 手段对于交通事故数据进行分析，可以分析出不同外部环境下的事故易发路段、热点区域，为交通管理提出更有针对性的建议。