“华为杯”第十五届中国研究生
数学建模竞赛

题 目　　基于恐怖袭击记录数据的量化分析

摘　　要：

本文针对恐怖袭击事件数据的量化分析问题，在对原始数据处理和统计分析的基础上，分别构建量化分级模型、基于遗传模拟退火算法的 FCM 聚类模型、支持向量机分类算法模型等模型，使用 Matlab 和 python 编程，对恐怖袭击事件进行分级，并对作案者进行分类和判别。

针对问题一，要求建立基于数据分析的量化分级模型，对附件中事件的危害程度分级，列出近二十年来危害程度最高的十大恐怖袭击事件并给出题目事件的分级。首先，除了人员伤亡和经济损失两大指标以外，本文考虑到恐怖袭击特有的影响其危害程度的因素，比如地域，针对的对象等，将其定义为社会危害指标，反映对社会造成的恐慌情况；其次，引入等效伤亡折算因子方法，对分类指标进行统计分析，并以此为基础对非数值型变量进行量化处理；再次，将人员伤亡、经济损失和社会危害三大指标加总得到总体危害度，并得到相应的排序；最后根据危害度的取值和频数分布情况对其进行分级。模型结果表明，十大恐怖袭击事件分别为 9·11 事件，美国驻肯尼亚大使馆遭汽车炸弹袭击，IS 与伊拉克库尔德武装交战，索马里首都酒店袭击事件，ISIS 攻陷伊拉克第二大城市，俄罗斯别斯兰人质事件，8·14 伊拉克雅兹迪区爆炸案，7·11 孟买连环爆炸案，伊拉克芥子气火箭弹事件和乍得反政府武装攻入首都事件。题目事件分级见正文表 7。

针对问题二，首先使用基于遗传模拟退火算法的 FCM 聚类模型对 2015、2016 年无人宣称负责的恐怖事件聚类，通过不断调节聚类数，选择出合理的聚类结果；然后结合问题一求解的危害度，得出每一类作案者造成的危害度之和，选出危害度最大的前五类作案者并按照 1-5 号标号；最后，对于标记好序号的数据集，使用有监督的分类算法--支持向量机对数据集训练并测试，通过交叉验证和网格搜索选择出适当参数，使得模型在训练集上的预测准确率达到 100%，在验证集上的预测准确率达到 99.6%，然后用训练好的模型对 2017 年的 10 起事件进行分类预测，得到 10 类事件对每一类别的隶属度，事件最可能归属的类别分别为 1、3、2、5、5、5、1、1、1、1。

针对问题三，对于事件发生原因，首先对 motive 字段的文本数据绘制词云图，得出恐怖袭击动机中的主要关键词为暴乱、报复、伊斯兰教等，说明恐怖袭击事件发生的主要原因是为了煽动暴乱、报复社会、反政府和受宗教影响，进一步统计作案最多的前 10 个犯罪集团，其中 6 个都属于伊斯兰教极端宗教组织，其发动的袭击次数是前 10 类总次数的 76.07%，说明极端宗教主义对恐怖袭击有重大影响；对于时空特性，分不同的时间和国家，统计发生的恐怖袭击次数，做出时间统计图和空间分布地图，得出恐怖袭击活动随时间在减少，空间上伊拉克发生恐怖袭击的次数最多，远大于其他地区；对于蔓延特性，首先分别做出 2015、2016、2017 年恐怖袭击活动的空间地图，发现随时间推移，恐怖袭击活动在空间上呈扩散趋势，然后利用恐怖袭击密度的变化进一步分析蔓延特性；对于级别分布，分别计算不同时空下恐怖袭击的平均危险度，得出不同时空下级别分布集中在 4 级和 5 级；最后，对不同的地区，利用灰色预测 GM(1,1)模型分别预测各地区恐怖袭击密度随时间的变化，从而根据未来反恐袭击密度的增减对反恐斗争提供建议。

针对问题四，要求思考通过数学建模还可以发挥附件 1 数据的哪些作用，并给出模型和方法。为了给反恐决策提供参考的理论依据，为了量化恐怖袭击发生前防范措施的有效性，以及发生恐怖袭击前恐怖主义者的动作对恐袭事件威胁度的影响等问题，构建多模块贝叶斯网络模型，以在恐怖活动前恐怖分子是否威胁、恐吓或向大众传播某种恐怖消息和民众或决策人员对恐怖袭击的了解程度两个变量为例，量化这两个变量在不同状态下，恐怖袭击事件的威胁程度的变化情况。结果表明，恐怖分子在恐怖活动前威胁、恐吓或向大众传播某种恐怖消息会增加恐怖袭击事件的威胁程度，以及随着民众或决策人员对恐怖事件信息了解程度的加深，恐怖袭击事件的威胁度下降，说明对民众普及恐怖主义方面的知识很必要的。

本文后续对模型进行了优缺点分析，并对模型做出了横向的推广。基于模型存在的缺点，给出了相应的改进思路。

关键词：全球恐怖主义；量化分析；遗传模拟退火算法；支持向量机；贝叶斯网络；

目 录

更多数学建模资料请关注微店店铺“数学建模学习交流”
https://k.weidian.com/RHO6PSpA

§1 问题的重述

1.1 问题背景

恐怖主义行为从广义上讲，是故意滥杀滥伤、在人群中使用暴力手段制造社会恐慌的攻击行为；或者是以胁迫政府、社会组织等为实现其财政，政治，宗教或意识形态为目标的主张和行为。因此，它主要指针对和平时期目标的暴力行为或针对非战斗人员的战争。对于恐怖主义的界定，GTD 数据库给出了构成恐怖主义事件的三条标准：第一，暴力行为必须以实现政治、经济、宗教或社会目标为目的；第二，意图胁迫、恐吓或煽动更多的群众；第三，行动必须超出合法战争活动的背景范围，它以非战斗人员为目标。如果事件中使用非法武力和暴力行为，由无国家支持的非国家行为者做出，并且满足上述标准中的两条，则该事件就是恐怖主义事件。

近年来，世界范围内频频发生恐怖袭击行动，从震惊世界的美国“911 事件”到 7•11 孟买连环爆炸案，再到 11 • 13 巴黎恐怖袭击事件等等，恐怖袭击的重点也从伊拉克和叙利亚等中东国家，逐步蔓延到世界各地。恐怖袭击不仅带来巨大的人员伤亡和财产损失，还给人们留下了巨大的心理恐慌，造成社会动荡，干扰了人们日常的工作和生活，极大程度地阻碍了经济的发展。因此，根据恐怖主义袭击事件的数据记录，对数据进行深入分析，有助于研究恐怖行为特征，分析恐怖行为的动因，对预防恐怖袭击和反恐斗争提供重要的参考依据。

1.2 问题相关数据

全球恐怖主义数据库（GTD）在 1998-2017 年期间世界上发生的恐怖袭击事件的记录在题目附件 1 中给出。附件 2 提供了对应的变量说明。

全球恐怖组织数据库是一个开源的数据库，每年研究机构都会对部分历史数据进行一些勘误或者更新。该数据库中的数据具备高度系统化，包含了国际上发生的历次恐怖袭击的事件记录。对于每一次恐怖袭击事件，GTD 数据库提供了发生袭击的时间、地点、袭击目标、袭击方式、使用的武器类型、伤亡数量以及宣称负责的恐怖组织或者个人等等详细字段。

1.3 待解决的问题

问题一：构建量化分级模型，依据危害性对恐怖袭击事件分级。

对于灾难性事件的分级管理，比如交通事故、地震、气象灾害等等，通常采用主观方法进行分级，由权威组织或部门选择若干个主要指标，以人员伤亡和经济损失程度为划分依据，强制规定分级标准。然而恐怖袭击事件的危害性与上述灾难性事件不同，它不仅取决于人员伤亡和经济损失，还与发生的时机、地域、针对的对象等等诸多因素有关，因此采用上述的分级方法是不合理的。

题目要求依据附件 1 和其它有关信息，结合现代信息处理技术，借助数学建模方法建立基于数据分析的量化分级模型，将附件数据给出的事件按危害程度从高到低分为一至五级，并要求列出近二十年来危害程度最高的十大恐怖袭击事件，并给出待分级事件的分级情况。

问题二： 依据事件特征发现恐怖袭击事件制造者。

附件中有多起恐怖袭击事件尚未确定作案者。若将可能是同一个恐怖组织或个人在不同时间、不同地点多次作案的若干案件串联起来统一组织侦查，有助于提高破案效率，尽早发现新生或者隐藏的恐怖分子；

题目要求针对在 2015、2016 年度发生的、尚未有组织或个人宣称负责的恐怖袭击事件，运用数学建模方法将可能是同一个恐怖组织或个人在不同时间、不同地点多次作案的若干案件归为一类，对应的未知作案组织或个人标记不同的代号，并按该组织或个人的危害性从大到小选出其中的前 5 个，记为 1 号-5 号。再对题目列出的恐怖袭击事件，按嫌疑程度对 5 个嫌疑人进行排序。

问题三： 基于数据对未来反恐态势的分析。

对未来反恐态势的分析评估有助于提高反恐行动的针对性和效率。

题目要求依据附件数据并结合网上的相关信息，建立数学模型，研究近三年来恐怖袭击事件发生的主要原因、时空特性、蔓延特性、级别分布等规律，进而分析研判下一年全球或某些重点地区的反恐态势。要求将研究结果用图/表给出，并根据研究结果对反恐斗争提出自己的见解和建议。

问题四： 数据的进一步利用。

为了充分利用附件的数据信息，题目要求自己提出进一步的研究问题，并给出相应的模型和方法。

§2 问题的分析

1．对问题一的分析

问题一要求建立基于数据分析的量化分级模型，对附件中事件的危害程度分级，列出近二十年来危害程度最高的十大恐怖袭击事件并给出题目事件的分级。针对问题一要求，首先，除了人员伤亡和经济损失两大指标以外，还需要考虑恐怖袭击特有的影响因素，比如地域，针对的对象等，本文将其定义为社会危害指标，反映对社会造成的恐慌情况；其次，引入等效伤亡折算因子方法，对分类指标进行统计分析，用于量化处理非数值型变量；再次，将人员伤亡、经济损失和社会危害三大指标加总得到总体危害得分，并得到相应的排序；最后根据危害度的取值和频数分布情况对其进行分级。

2．对问题二的分析

问题二要求对 2015 年和 2016 年发生的，暂时无人宣称对其负责的恐怖事件归类，从而将未知作案者的恐怖事件标记不同的类别；然后根据每一类的危害性排序，选出危害性最大的前五类，标记为 1-5 号；最后根据筛选出的前五类恐怖事件，预测题目表 2 给出的 2017 年发生的 10 起恐怖事件归属于每一类的可能性。针对问题二要求，首先使用基于遗传模拟退火算法的 FCM 聚类算法对 2015、2016 年无人宣称负责的恐怖事件聚类，通过不断调节聚类数，选择合理的聚类结果；然后结合问题一求解的危害度，得出每一类作案者造成的危害度之和，并排序，得到危害度最大的前五类作案者并按照 1-5 号标号；最后，对于标记好序号的数据集，使用有监督的分类算法--支持向量机对数据集训练并测试，经过适当参数调整之后选择最优的模型对 2017 年的 10 起事件进行分类预测。

3．对问题三的分析

问题三要求分析根据近三年的数据分析恐怖袭击事件发生的主要原因、时空特性、蔓延特性和级别分布，然后根据数据特征，预测下一年恐怖袭击活动情况，从而为反恐斗争提供见解和建议。为了分析事件发生的原因，对 motive 字段的文本数据绘制词云图，反映恐怖袭击动机中的关键字，然后根据关键字反映的信息，进一步分析极端宗教主义对恐怖袭击的影响；对于时空特性，分不同的时间和不同的区域，统计发生的恐怖袭击次数，做出统计图和空间分布地图，分析恐怖袭击活动随时空改变的差异情况；对于蔓延特性，首先分别做出 2015、2016、

2017 年恐怖袭击活动的空间地图，分析随时间推移，恐怖袭击活动在空间上的蔓延，然后利用恐怖袭击密度（单位面积发生恐怖袭击的次数）的变化进一步分析蔓延特性；对于级别分布，分别统计不同时空下恐怖袭击的平均危险度，然后根据问题一求解的危险度及级别，判断恐怖袭击的级别分布；最后，对不同的区域，利用灰色预测 GM(1,1)模型分别预测各地区恐怖袭击密度随时间的变化，从而根据未来反恐袭击密度的增减对反恐斗争提供建议。

4．对问题四的分析

本问题是数据的进一步利用，要求思考通过数学建模还可以发挥附件 1 数据的哪些作用？并给出模型和方法。由问题一、问题二、问题三的分析，发现恐怖袭击事件的多特征性以及恐怖袭击事件可能造成的后果的严重性，这迫切需要对恐怖袭击事件的威胁程度进行评估，从而为反恐决策者提供决策支持，以尽可能的减少恐怖袭击所造成的影响。由于恐怖袭击事件评估信息的多样性、不确定性、模糊性及复杂性，采用多模块贝叶斯网络模型对恐怖袭击事件的威胁程度进行评估。采用 K2 算法和专家评估构建贝叶斯网络结构、进行参数学习，联结树推理算法计算后验概率，最后计算出恐怖袭击威胁度。通过上述方法，我们不仅可以判别所给特征信息是否影响恐怖袭击事件威胁度，也可以通过已有特征信息去预测恐怖袭击事件的威胁度，以便有关部门及时采取相关行动和措施，减少恐怖袭击事件造成的损失和影响。

§3 模型的假设与符号说明

3.1 模型假设

假设 1：题目附件中所有的数据是真实有效的；
假设 2：财产损失程度未知时，损失程度最轻，等效于没有财产损失情况；
假设 3：首要武器类型杀伤力最大，可忽略其他武器类型；
假设 4：恐怖分子以首要攻击目标为主，可忽略其他目标类型；
假设 5：恐怖分子进行恐怖活动的特征是连贯的，且不会轻易改变；
假设 6：假设恐怖袭击事件威胁程度的先验信息已知。

3.2 符号说明

序号	符号	符号说明
1	L	事件造成的直接经济损失
2	T_0	初始温度
3	T_{end}	终止温度
4	*Sizepop*	个体数目
5	*MAXGEN*	最大遗传代数
6	*Pc*	交叉概率
7	*Pm*	变异概率
8	*Jb*	目标函数值

§4 数据的处理

4.1 数据库变量字段

GTD 数据库中对袭击事件描述的有 134 个变量，主要分为⑴ GTD 的标志号

和日期；⑵事件信息；⑶事件发生的地点；⑷武器信息；⑸目标/受害者信息；⑹攻击信息；⑺凶手信息；⑻伤亡和后果；⑼附加信息和来源共九个类别。变量分为数值变量、文本变量和分类变量三类。

首先，剔除文本变量。由于文本变量对事件特征的描述与其他字段相重复，比如事件摘要字段，描述了事件的 “时间，地点，人物，内容，过程，原因”，这些变量在其它字段均有体现，因此可将其直接删除。

其次，剔除完整度很低的变量字段。在 GDT 数据库中，完整度在 20%以下的变量字段应当被剔除[1]，比如第三武器类型字段有大于 98%的空缺数据，同时第一武器和第二武器类型已经描述了主要的武器类型，因此直接删除该字段对结果的影响不大。

最后，本文在模型构建部分剔除疑似恐怖主义字段取值为 1 的数据。该字段取值为 1，说明对袭击事件是否为恐怖袭击表示怀疑，模型构建部分主要是针对恐怖袭击事件而言，包括恐怖袭击事件的量化分级和特征提取，样本量足以保证模型的准确性，剔除该部分数据可以减少噪音数据的影响。

4.2 指标的选择和处理

由 4.1 的初步筛选，进一步参考现有文献的研究结果[1-2]，选取的指标和数据预处理如表 1 所示：

表 1　指标的选择和预处理

变量类别	字段	预处理
GTD 的标志号和日期	标志号(eventid)	无
	日期(idate)	由 iyear,imonth,iday 合成
事件信息	疑似恐怖主义(doubtterr)	无
	相关事件(related)	问题一中将其合并为同一事件
事件发生的地点	地区(region)	无
	地理编码特征(specificity)	无
攻击信息	成功的攻击(success)	无
	自杀式袭击(suicide)	无
	攻击类型(attacktype1)	无
目标/受害者信息	目标/受害者类型(targtype1)	无
武器信息	武器类型(weaptype1)	无
伤亡和后果	死亡总数(nkill)	死亡总数(nkill)-凶手死亡人数（nkillter）
	受伤总数(nwound)	受伤总数(nwound)-凶手受伤人数(nwoundte)
	财产损害程度(propextent)	空值替换为该事件对应财产损失(property)字段的取值

其中财产损害程度字段针对财产损失字段取 1 的变量，对确定恐怖袭击造成财产损失的程度进行分级(1-4 级)，该字段空值部分表示袭击事件并未造成财产损失或者不知道事件是否造成财产损失，因此对空值部分取 4 处理是合理可行的。

此外，问题一对相关联事件进行合并处理。附加说明字段(addnotes)指出：当一些案例联系在一起时，信息来源会提供所有事件的一个累计死亡总数，而不是每个事件的死亡数字。因此针对伤亡字段，采用关联事件的累加结果。对于其他分类变量字段，选取代表伤害度最大的类别赋值。

§5 问题一的解答

5.1 模型的建立

5.1.1 量化分级模型

针对恐怖袭击的危害程度，建立基于人员伤亡，财产损失和社会危害三大指标体系下的量化分级模型。该模型不同于交通事故的分级界定，考虑到恐怖袭击在其发生地域和袭击目标等方面具有特有的社会危害性。

指标体系如图 1 所示：

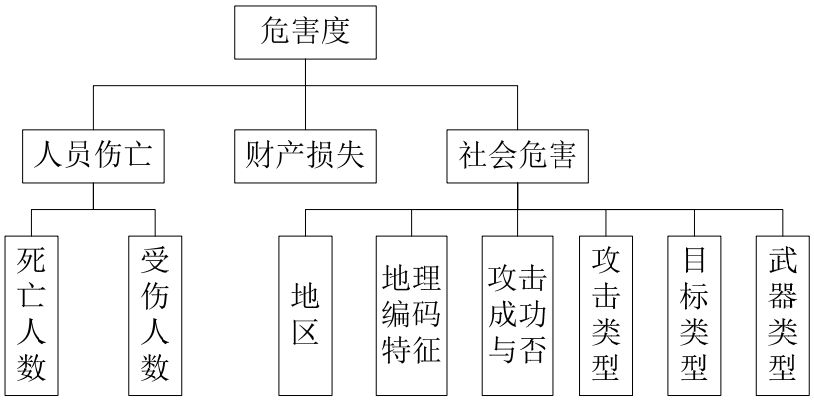

图 1　恐怖事件的量化分级模型

其中恐怖袭击发生的地区（全球范围的划分），地理编码特征（发生城市/乡/镇的等级），攻击成功与否，攻击类型，目标/受害者类型和武器类型对人们心理恐慌程度具有不同程度的影响，因此将其划分为社会危害这一指标体系下。

5.1.2 指标量化处理

指标体系中只有人员伤亡变量是数值型变量，其他变量均为分类变量。因此，需要对变量进行量化处理。采用“等效死亡”换算方法[3]，可将伤亡类指标、损失类指标、地域特征类指标和攻击类指标等等全部换算为“等效死亡”数。

为了尽量避免人为划分造成主观性影响，对伤亡类指标和损失类指标使用已有的标准进行折算；对其余的分类指标进行统计分析，得到相应的折算因子。

表 2　等效死亡折算因子标准

	类别	折算因子
死亡人数	无	1
受伤人数	无	0.3
财产损失程度	1=灾难性的（L>10 亿美元）	1000
	2=重大的（100 万美元≤L＜10 亿美元）	50
	3=较小的（L< 100 万美元）	3
	4=未知	0

*其中 L 表示事件造成的直接经济损失，单位：美元。

下面以地区和武器类型字段为例做详细说明：

地区　恐怖袭击发生在不同地区造成社会影响或者说是人们的心理的恐慌程度是不同的。这种社会影响主要体现在，发生在不同区域造成的人员伤亡和财产损失是不同的，比如发生在区域 1，即北美地区（包括加拿大，墨西哥，美国），如图 2 所示(图中人员伤亡和财产损失已根据折算因子折算为“等效死亡”人数)，

造成的平均人员伤亡和财产损失远高于其他地区，同时该地区属于发达国家，因此造成的恐慌程度远高于其他地区。

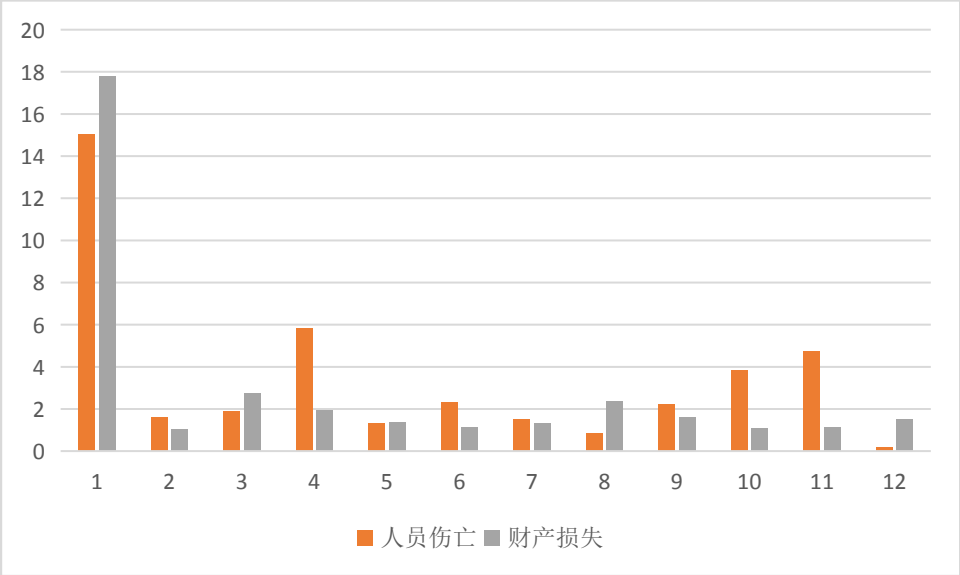

图 2 单次恐怖袭击发生在不同区域的平均伤害

表 3 地区的“等效死亡”折算因子

地区	注释	折算因子	地区	注释	折算因子
1	北美	32.8005	7	中亚	2.8198
2	中美洲和加勒比海地区	2.6414	8	西欧	3.1654
3	南美	4.6214	9	东欧	3.8348
4	东亚	7.7542	10	中东和北非	4.9116
5	东南亚	2.6824	11	撒哈拉以南的非洲	5.8771
6	南亚	3.4348	12	澳大利亚和大洋洲	1.6930

根据表 3，北美地区的等效死亡折算因子最高，说明恐怖袭击造成的社会危害（心理恐慌）最大，其次为东亚地区，澳大利亚和大洋洲的等效死亡折算因子最低，符合常识，说明了折算因子的有效性。

目标类型 恐怖袭击针对不同的目标/受害者类型造成社会影响或者说是人们的心理的恐慌程度是不同的。对应的单次恐怖袭击针对不同目标的平均伤害和“等效死亡”折算因子如图 3 和表 4 所示：

目标类型为 11,15,19，即目标为海事，宗教人物/机构，交通运输时，等效死亡的折算因子较高，目标类型为未知和记者与媒体的等效死亡的折算因子较低。直观来看，海事，交通运输遭遇恐怖袭击时由于发生在公众场合，引起的伤亡和财产损失就比较大，引发的社会恐慌更严重，等效死亡折算因子较高是完全合理的，而宗教人物/机构作为恐怖事件的目标是一个广泛发生又比较特殊的情况，宗教信仰对人们思维模式和行为模式的影响至关重要，比如伊斯兰教文化里，圣战是其重要特征，在圣战的感召下，恐怖分子视死如归，造成的危害十分巨大。

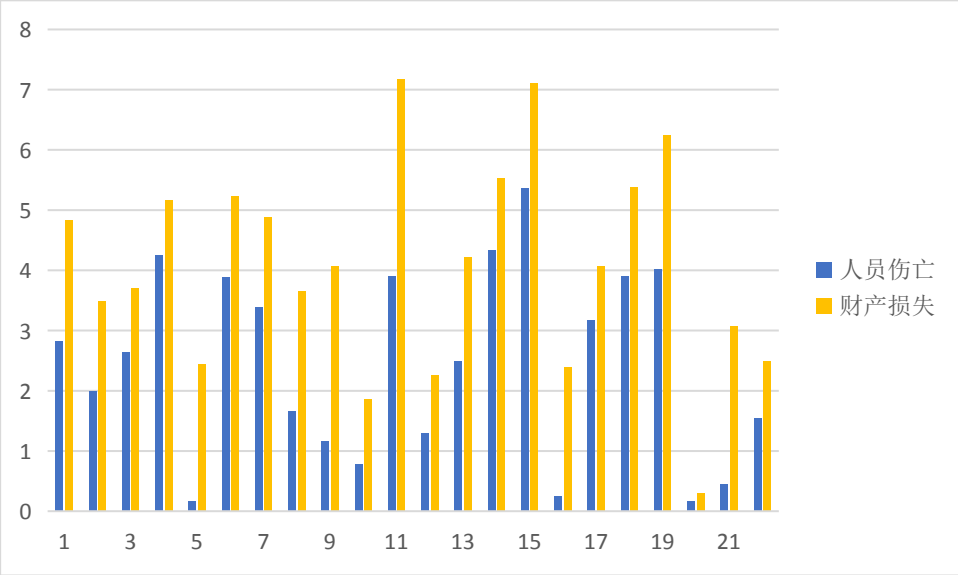

图 3 单次恐怖袭击针对不同目标的平均伤害

表 4 目标类型的“等效死亡”折算因子

类型	文本注释	折算因子	类型	文本注释	折算因子
1	商业	4.8237	12	非政府组织	2.2547
2	政府（一般意义）	3.4885	13	其他	4.2256
3	警察	3.7046	14	公民自身和私有财产	5.5223
4	军事	5.1615	15	宗教人物/机构	7.1036
5	流产有关	2.4347	16	电信	2.3958
6	机场与飞机	5.2391	17	恐怖分子/非州立民兵组织	4.0734
7	政府（外交）	4.8875	18	游客	5.3832
8	教育机构	3.6577	19	交通运输	6.2396
9	食品和水供应	4.0680	20	未知	0.2947
10	记者与媒体	1.8630	21	公用事业	3.0639
11	海事	7.1814	22	暴力政党	2.4834

其余指标字段的折算因子结果见附表 1。

5.1.3 危害程度的分级

代入各指标的折算因子，加总得到每次恐怖袭击事件的危害度。统计危害度的分布情况，根据危害度取值和频数分布情况分级，危害程度的频数分布直方图如图 4 所示，其中子图(a)显示了整体分布情况，危害度在左端的集中度较强，右端较为分散，考虑到数据特点，结合危害度极端分位点、中高分位点的取值等，对危害程度分级，比如危害度为一级时表示危害度最大，可以说这种恐怖袭击是空前的、灾难性的，引入金融数学中的在险值(VaR)的概念，将 0.1%分位点处的取值作为一级的危害度界限，以此类推，将危害度划分为五级，其中子图(b)(c)(d)(e)(f)分别表示危害程度为五级、四级、三级、二级、一级时的危害度分布情况。分级结果如表 5 所示。

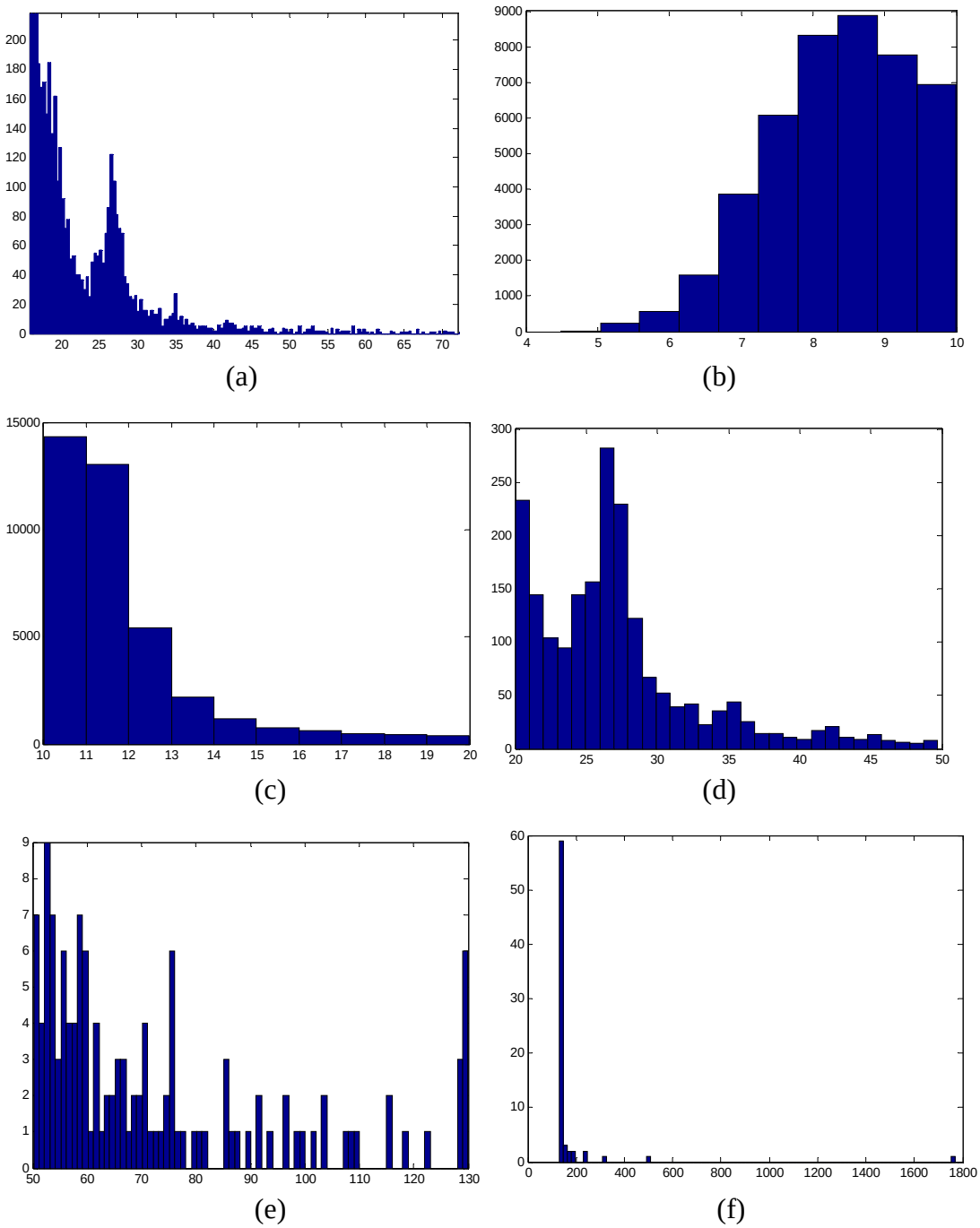

图 4　危害度的频数分布直方图

表 5　危害程度分级结果

危害度取值	分级
大于 130	一级
50-130	二级
20-50	三级
10-20	四级
0-10	五级

5.2 模型的求解结果

代入各指标的折算因子，加总得到危害度。对附件 1 的数据集处理，计算每个恐怖袭击事件的危害度得分。得分前十的事件，即危害程度最高的十大恐怖袭击事件，如表 6 所示。

表 6　近 20 年十大恐怖袭击事件

序号	标志号	事件名称	危害度
1	200109110004	911 事件	1770.5754
2	199808070002	美国驻肯尼亚大使馆遭汽车炸弹袭击	501.8333
3	201408030059	IS 与伊拉克库尔德武装交战	325.8900
4	201710140002	索马里首都酒店袭击事件	238.0787
5	201406100042	ISIS 攻陷伊拉克第二大城市	234.0884
6	200409010002	俄罗斯别斯兰人质事件	191.9522
7	200708150005	8.14 伊拉克雅兹迪区爆炸案	185.0564
8	200607120001	7·11 孟买连环爆炸案	171.5033
9	201603080001	伊拉克芥子气火箭弹事件	170.6112
10	200802010006	乍得反政府武装攻入首都	161.8911

任务 1 中典型事件的分级如表 7 所示：

表 7 典型事件危害级别

事件编号	危害级别
200108110012	一级
200511180002	二级
200901170021	一级
201402110015	三级
201405010071	四级
201411070002	四级
201412160041	二级
201508010015	五级
201705080012	四级

*其中编号为 201402110015 的事件属于疑似恐怖主义事件，不在已有的危害度评分范畴，需要重新定量计算。

§6 问题二的解答

6.1 模型的建立

6.1.1 基于遗传模拟退火算法的FCM聚类模型

模糊 C-均值聚类（FCM）通过在欧几里得空间确定数据点的几何贴进度，将数据聚集到不同的类别，FCM 算法因为其思想简单、易实现而被广泛使用。但是由于 FCM 算法需要先随机初始化模糊分类矩阵，初始化矩阵的优劣将直接影响聚类结果，可能导致局部最优解，在处理大量数据时，更加容易出现收敛到局部最优解的问题。

经过筛选后得到 2015 和 2016 年尚未有组织或个人宣称负责的恐怖袭击事件有 18961 条，数据量较大，为了避免 FCM 算法收敛到局部最优解的问题，将其

与遗传算法和模拟退火算法结合，从而对 FCM 算法做出改进，得到的混合算法可以更加有效、快速的收敛到全局最优解。

基于遗传模拟退火算法的 FCM 聚类算法的主要思想[4]为：

①设置参数初始值：

a.模糊 C-均值聚类算法：幂指数为 3，最大迭代次数为 20，目标函数的终止容限为10^{-6}，聚类数为 *cn*，尝试取不同的聚类数，比较模型效果，然后选择效果最好的聚类数进行聚类；

b.模拟退火算法：冷却系数 q=0.8，初始温度 T_0=100，终止温度 T_{end}= 1;

c.遗传算法：个体数目 *sizepop*=10，最大遗传代数 *MAXGEN*=50，交叉概率 p_c=0.7，变异概率 p_m=0.01。

②根据聚类中心，生成初始种群，计算每个样本的适应度值和对于各聚类中心的隶属度。

③对种群进行交叉、变异、选择操作，生成新个体，将新个体的适应度与旧个体的适应度比较，判断是否接受新个体。

④迭代次数加 1，循环步骤③，直至迭代次数达到最大遗传代数 *MAXGEN*，跳出步骤③，转到步骤⑤。

⑤如果当前温度 $T<T_{end}$，则返回全局最优解，算法结束；否则执行降温操作 $T_{i+1}=k*T_i$，转到步骤③。

⑥输出目标函数 *Jb* 值，*Jb* 值越小，说明个体的适应度越高。

6.1.2 支持向量机分类模型

第二题最后一问是对五类样本的有监督训练学习，根据训练好的模型对未知新样本进行分类预测，选用支持向量机多分类算法求解此问。支持向量机分类算法（SVM）是构造两个类别之间的最优超平面，使得两类之间的间隔最大，其实质是求解满足约束的最优化问题。

对于多分类问题可以采用一对多的 SVM 算法和一对一的 SVM 算法，从而将多分类问题转化成二分类问题，进而求解。其中，一对多的 SVM 算法是指对于 *n* 类数据集，构造 *n* 个两类子分类器，将每类数据与其它类数据分开。例如：第 *k* 个子分类器是将第 *k* 类数据作为正类，其余 *n*-1 类数据作为负类。一对一的 SVM 算法时任意两类数据构造一个分类器，共构造$\frac{n(n-1)}{2}$个子分类器，形成一个多分类器组合[5]。针对本文的数据特点，选用一对多的 SVM 分类算法。

SVM 算法的参数设置：

①对特征向量做归一化处理。

②选取核函数：SVM 算法的常用核函数有线性核函数、多项式核函数、高斯径向基核函数（RBF）、Sigmoid 核函数等，其中高斯径向基核函数在一般情况下性能较好，因此选用高斯径向基核函数。

③参数设置：对于 RBF 核函数，需要设置 gamma 参数和惩罚参数 C，惩罚参数 C 的默认值为 1，gamma 参数的默认值为$\frac{1}{n}$（n 为类别数）。直接使用默认参数的训练效果较差，采用交叉验证和网格搜索寻优获得最优的 C 和 gamma，同时考虑到参数 C 和 gamma 值过大会导致过拟合问题（即训练集准确率很高而预测集准确率很低），最终选取参数值为 C=6，gamma=0.2。

④选取训练样本和测试样本，用择优的参数 C 和 gamma 训练模型。

⑤用测试样本检验模型的优劣。

⑥使用训练好的模型对未知新样本分类。

6.2 模型的求解

6.2.1 数据准备

使用问题一量化处理之后的数值型数据，考虑与凶手作案特点相关的因素，并剔除数据大量缺失的指标，选取出地区（region）、地理编码特征(specificity)、成功的攻击（success）、攻击类型（attacktype1）、目标/受害者类型（targtype1）、武器类型（weapontype1）、受害者死亡人数（nkillnew）、受害者受伤人数（nwoundtenew）、财产损害程度（propextent）、团伙作案（individual）共 10 个指标。筛选出 2015-2016 年所有 claimed 值为 0（即没有人宣称对事件负责）的事件，最终得到 18961 条数据，10 个指标（数据见附件表 2.1）。

6.2.2 样本聚类

对筛选后的数据进行归一化处理，将所有事件按照 eventid 的顺序从 1 开始编号，作为事件序号列。根据基于遗传模拟退火算法的 FCM 聚类算法，利用 MATLAB 编程（程序见附程序 1），FCM 聚类算法需要首先设置聚类数 *cn*，由于样本的类别数未知，所以尝试取不同的类别数，计算目标函数 *Jb* 值，*Jb* 值越小，说明个体的适应度越高，通过 *Jb* 值判断聚类模型效果，选择效果最好的聚类数进行聚类。

首先设置聚类数为 10，得到部分聚类结果见表 8。

表 8 10 类聚类结果表

类别	每类事件总数	事件序号									
1	1031	21	169	174	269	284	287	310	406	407	415
2	1058	24	30	38	39	102	108	109	126	155	158
3	1509	3	4	6	7	28	36	46	47	68	69
4	1396	2	9	10	33	37	41	60	66	72	91
5	2303	1	32	44	45	53	55	56	74	85	87
6	1660	11	14	19	22	34	82	83	112	114	133
7	1633	12	42	49	58	59	61	77	78	103	106
8	1081	8	26	54	63	146	206	216	236	239	240
9	6109	5	15	16	18	20	23	27	29	31	35
10	1181	13	17	25	40	62	64	93	99	127	129

此时得到的 Jb= 40.2441，Jb 值较大，说明算法中个体适应度不高，聚类效果有待提高。考虑到样本数据中的分属的类别数较多，仅仅将样本聚为 10 类存在很大的不合理性，所以更改聚类数为 20 和 30，再分别运行程序，得到三种聚类数情况下的 Jb 值见表 9。

表 9 不同聚类数下 Jb 值

聚类数	10	20	30
Jb 值	40.2441	8.872	3.9193

由表 9 可以看出，随着聚类数增加，Jb 值有明显下降。在聚类数为 30 时，Jb=3.9193，结果比较合理，而继续增加聚类数，对模型效果没有明显提升，所以最终选取的聚类数为 30 类，计算得出样本聚类结果（见附件表 2.2）。

6.2.3 各类危害性计算

由问题一结果，可以得出每个恐怖袭击事件造成的危害度，求解每一类中所

有事件的危害度之和，作为这一类犯罪分子造成的总危害度，结果见表 10。

表 10 各类危害度

类别	危害度	类别	危害度
1	3527.8099	16	10064.4205
2	3088.3296	17	3307.1124
3	8114.6778	18	4939.9176
4	6736.7605	19	6728.6854
5	5453.5513	20	1807.1341
6	997.1202	21	7405.4061
7	4934.4838	22	2592.0380
8	2659.5287	23	8963.4065
9	5329.7677	24	4196.3334
10	496.8715	25	4939.9821
11	37223.3259	26	14386.6324
12	2186.5685	27	1178.0751
13	2626.8442	28	13299.6026
14	1112.4221	29	4097.0951
15	2624.9903	30	16704.0427

危害度最大的前五类分别是第 11、30、26、28、16 类，从原样本中筛选出这 5 类，共 8031 条样本，然后依次为 5 类样本标号 1-5（数据见附件表 2.3）。

6.2.4 新样本分类

根据支持向量机多分类算法，在附件表 2.3 中随机选取 75%的数据作为训练集，剩余数据作为验证集，利用 Python 编程（程序见附程序 2），得到训练集的预测准确度为 100%，验证集的预测准确度为 99.6%，说明 2008 条验证样本只有 8 条分类错误，分类准确度很高，模型训练效果较好。

利用训练好的模型对问题二中给定的 2017 年 10 条恐怖袭击事件分类，输出每个事件的 decision_function 结果，SVM 算法中的 decision_function 计算得出样本点归属于某一类别时到分割超平面的函数距离，距离越大，样本点归属于该类的可能性越大。计算的函数距离见表 11。

表 11 函数距离表

恐怖袭击事件	第 1 类	第 2 类	第 3 类	第 4 类	第 5 类
201701090031	4.4689	-0.2545	0.8998	2.0228	2.8630
201702210037	2.9979	0.9427	4.1946	0.9321	0.9327
201703120023	2.0117	4.0723	0.9537	-0.1167	3.0790
201705050009	2.0504	3.0422	-0.4457	0.8999	4.4532
201705050010	2.0509	3.0429	-0.4415	0.8992	4.4485
201707010028	1.9343	-0.3005	0.8009	3.2615	4.3039
201707020006	4.4652	-0.3662	1.9500	3.2188	0.7322
201708110018	4.1969	2.0218	-0.1340	3.0786	0.8368
201711010006	4.4698	-0.1916	0.7565	2.0284	2.9369
201712010003	4.5000	-0.2786	0.8337	3.0803	1.8646

将表 11 中的负向距离取绝对值，然后分别处理使每一行的距离之和为 1，得到的即为事件归属于每一类的可能性，结果见表 12。

表 12 事件归属于每一类的可能性及预测类别

恐怖袭击事件	第 1 类	第 2 类	第 3 类	第 4 类	第 5 类	预测类别
201701090031	0.4252	0.0242	0.0856	0.1925	0.2724	1
201702210037	0.2998	0.0943	0.4195	0.0932	0.0933	3
201703120023	0.1966	0.3979	0.0932	0.0114	0.3009	2
201705050009	0.1883	0.2793	0.0409	0.0826	0.4089	5
201705050010	0.1885	0.2796	0.0406	0.0826	0.4088	5
201707010028	0.1825	0.0283	0.0755	0.3077	0.4060	5
201707020006	0.4160	0.0341	0.1817	0.2999	0.0682	1
201708110018	0.4087	0.1969	0.0131	0.2998	0.0815	1
201711010006	0.4305	0.0185	0.0729	0.1954	0.2829	1
201712010003	0.4262	0.0264	0.0790	0.2918	0.1766	1

根据表 12 的结果可以得出恐怖分子关于典型事件的嫌疑度，结果见表 13。

表 13 恐怖分子关于典型事件的嫌疑度

恐怖袭击事件	1 号嫌疑人	2 号嫌疑人	3 号嫌疑人	4 号嫌疑人	5 号嫌疑人
201701090031	1	5	4	3	2
201702210037	2	3	5	1	4
201703120023	3	1	4	5	2
201705050009	3	2	5	4	1
201705050010	3	2	5	4	1
201707010028	3	5	4	2	1
201707020006	1	5	3	2	4
201708110018	1	3	5	2	4
201711010006	1	5	4	3	2
201712010003	1	4	3	2	3

6.3 灵敏度分析

在支持向量机分类算法中，调整惩罚参数 C 和参数 gamma 的值会直接影响模型训练效果，从而影响分类准确率。通过给 C 和 gamma 设置不同的值，得出训练集和测试集的准确率（结果见表 14,其中对于 C 不同的取值第一行结果为训练集准确率,第二行结果为测试集准确率），来观察 SVM 算法对参数 C 和 gamma 的敏感性。

表 14 不同参数组合下数据集准确率

C \ g	0.01	0.1	1	10	100
0.001	0.3590	0.3760	0.5680	0.3590	0.3590
	0.3540	0.3710	0.5630	0.3540	0.3540
0.01	0.5900	0.9460	0.9670	0.8210	0.5860
	0.5820	0.9390	0.9640	0.8120	0.5640
0.1	0.9580	0.9920	0.9850	0.9740	0.9280
	0.9530	0.9900	0.9830	0.9630	0.8770
1	0.9970	0.9980	1.0000	1.0000	1.0000
	0.9950	0.9940	0.9950	0.9830	0.9410
10	0.9990	1.0000	1.0000	1.0000	1.0000
	0.9980	0.9970	0.9950	0.9840	0.9430
100	1.0000	1.0000	1.0000	1.0000	1.0000
	0.9990	0.9970	0.9950	0.9840	0.9430

利用 MATLAB 编程（程序见附程序 3），对测试集准确率绘图，得出不同参数组合下，测试集准确率变化情况如图 5 所示。

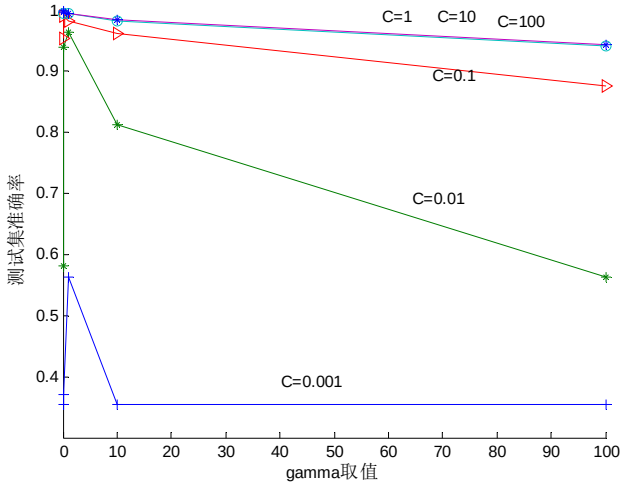

图 5 不同参数组合下，测试集准确率

由图 5 可以看出，保持 gamma 取值不变时，C 取值越大，测试集准确率越高，对于本题的数据集，C 取值大于 1 以后，准确率略有提升，效果不明显；保持 C 取值不变，随着 gamma 取值增大，测试集准确率有所波动，整体呈下降趋势。

通过灵敏度分析发现，不同的参数取值，得到模型效果差距很大，因此参数寻优是非常有必要的。

§7 问题三的解答

7.1 灰色预测GM(1，1)模型的建立

题目要求用近三年数据预测下一年全球或某些地区的反恐态势，由于数据量较少，考虑使用灰色 GM(1，1)模型。将全球分为几大区域，分别对每一区域的时间序列做预测，从而得出不同空间状态下，恐怖袭击活动的预测情况，从而为反恐提供依据。

灰色预测 GM(1，1)模型的步骤如下：

（1）数据检验与处理

计算数列的级比

$$\lambda(k)=\frac{x^{(0)}(k-1)}{x^{0}(k)},k=2,3,\cdots,n$$

若所有级比都在可容覆盖 $(e^{-\frac{2}{n+1}},e^{\frac{2}{n+1}})$ 内，则数列可以作为模型 GM(1,1)的数据进行灰色预测。否则，对数列进行变换处理，使其落入可容覆盖内。

（2）建立模型得到预测值

$$\hat{x}^{(1)}(k+1)=\left(x^{(0)}(1)-\frac{b}{a}\right)e^{-ak}+\frac{b}{a}$$

而 $\hat{x}^{(0)}(k+1)=\hat{x}^{(1)}(k+1)-\hat{x}^{(1)}(k)$

（3）检验预测值

残差检验：计算残差

$$\varepsilon(k)=\frac{x^{(0)}(k)-\hat{x}^{(0)}(k)}{x^{(0)}(k)},k=1,2,\cdots,n$$

若 $\varepsilon(k)<0.2$，则可认为达到一般要求；若 $\varepsilon(k)<0.1$，则认为达到较高要求。

级比偏差值检验：首先计算级比 $\lambda(k)$，再利用发展系数 a，求出相应的级比偏差：

$$\rho(k)=1-\left(\frac{1-0.5a}{1+0.5a}\right)\lambda(k)$$

若 $\rho(k)<0.2$，则认为达到一般要求；若 $\rho(k)<0.1$，则认为达到较高要求。

（4）预测结果

由模型 GM(1,1)所得到的结论，根据实际需要，给出相应的预测结果。

7.2 模型的求解

7.2.1 恐怖袭击发生的原因分析

为了分析恐怖袭击发生的原因，对原始数据中的变量进行分析，发现变量 motive 中包含大量袭击动机信息，由于该变量为文本型数据，考虑采用词云图分析文本中最常被提及的单词，并统计每个单词出现的频次。

筛选出 2015 至 2017 年三年，所有 motive 列非空并且确定为恐怖袭击的事件数据，得到 5295 条数据。对变量 motive 列的文本预处理，删除其中的连词、介词、冠词等无关词汇，然后对处理好的文本生成词云图，如图 6 所示。

图 6 作案动机词云图

词云图中的字体越大，说明该词汇出现的次数越多，使用词云图可以非常直观的得出文本中的关键词，同时可以输出各词汇的词频统计。截取词频数最大的前十个统计结果见表 15。

表 15 文本词频统计部分结果

关键词	Posit	Violence	Retiation	State	Accused	Islam	Electionion	Army	Military
词频数	1089	923	671	504	489	461	448	426	411

得出词频数最高的前十个关键词分别是暴乱、报复、政府（州）、控诉、伊斯兰教、选举、军方、操纵者、警察。说明恐怖袭击事件发生的主要原因是为了煽动暴乱、报复社会、反政府、受宗教影响等等。

对于关键词中出现的伊斯兰教，通过统计凶手信息判断恐怖分子是否受到宗教的影响，对 gname 列（即犯罪集团名称）去除所有未知作案者和空值，对筛选后的数据做计数统计，得到不同犯罪集团近三年来发动的袭击次数，选出次数最高的前十个作案者，结果见表 16。

表 16 犯罪集团袭击次数统计

犯罪集团	袭击次数
伊拉克和黎凡特伊斯兰国(ISIL)	3990
塔利班	3209
青年党	1531
博科圣地	1113
胡塞极端组织	892
库尔德工人党	855
新人民军	852
毛派	747
伊斯兰国西奈省	426
巴勒斯坦极端分子	370

通过查阅资料可知，排名前十的组织中除了胡塞极端组织、库尔德工人党、新人民军、毛派四个，其余六个都属于宗教极端组织，他们近三年来发动的袭击次数占 10 个组织发动的总袭击次数的 76.07%，说明宗教极端组织极易造成恐怖袭击事件，是恐怖袭击发生的重要原因之一。

7.2.2 时空特性

1.时间特性

分别统计不同年份、不同月份和不同周数情况下，恐怖袭击发生的次数，结果如图 7-9 所示。得出 2015-2017 年，恐怖袭击次数逐年下降，有递减趋势；各月袭击次数有波动，整体呈下降趋势，在 12 月份袭击次数最少，说明恐怖分子可能更偏好在上半年发动袭击；不同周数的袭击次数规律性不强，呈波动状态，周一次数最多，周五最少，可能与周一人群聚集有关，导致恐怖分子偏向选择周一袭击。

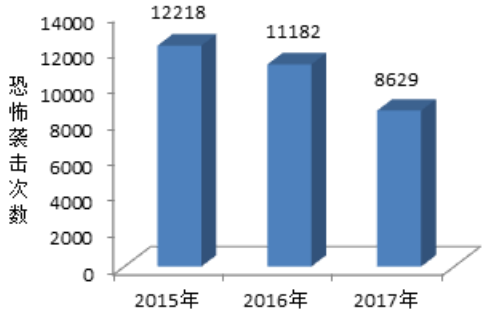

图 7 每年恐怖袭击次数

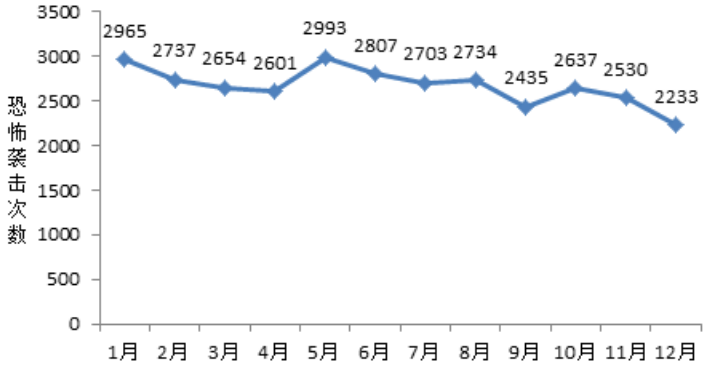

图 8 各月恐怖袭击次数

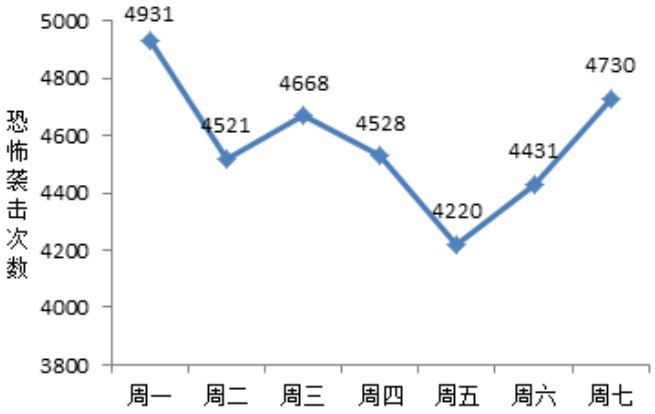

图 9 不同周数恐怖袭击次数

2.空间特性

统计不同国家近三年发生的恐怖袭击次数，利用 Python 编程（程序见附程序 4），绘制袭击次数地图，如图 10 所示。其中，区域颜色越深，说明该区域发生的袭击次数越多。

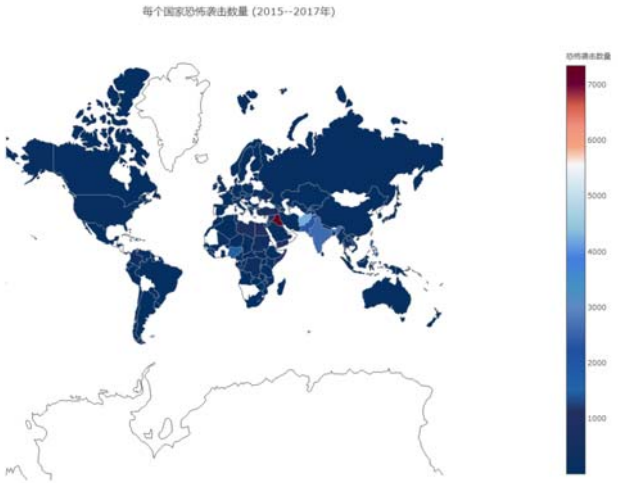

图 10 2015 至 2017 年每个国家发生的恐怖袭击次数

由图得出，伊拉克是受恐怖袭击最严重的国家，需要对恐怖袭击重点防范。

7.2.3 蔓延特性

为了分析恐怖袭击蔓延特性，分别统计 2015、2016 和 2017 年不同国家发生的恐怖袭击次数，并分年份绘制出袭击次数地图，如图 11 – 13 所示。

图 11 2015 年每个国家发生的恐怖袭击次数

图 12 2016 年每个国家发生的恐怖袭击次数

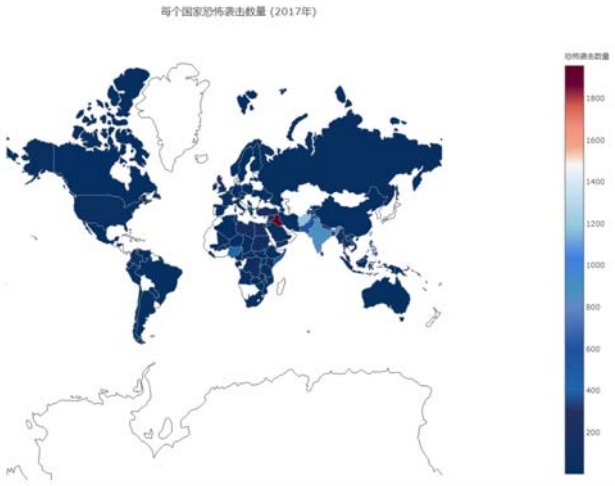

图 13 2017 年每个国家发生的恐怖袭击次数

由图 11–13 可以看出，随着时间的推移，恐怖袭击的空间范围在扩大，呈现扩散趋势。

为了进一步反映空间特性，本文定义了密度指标，指单位面积发生恐怖袭击的次数，密度越大，说明恐怖袭击的集中度越大，蔓延特性越不明显，对全球 12 个地区在 2015 年至 2017 年以半年为区间计算其密度值，结果如图 14 所示，结果表明，南亚发生恐怖袭击的密度最大，但最近三年有下降趋势，可能存在蔓延现象；澳大利亚和大洋洲发生恐怖袭击的密度最小，说明恐怖袭击在该地区发生的概率较低，大多数区域属于较为安全的地带。

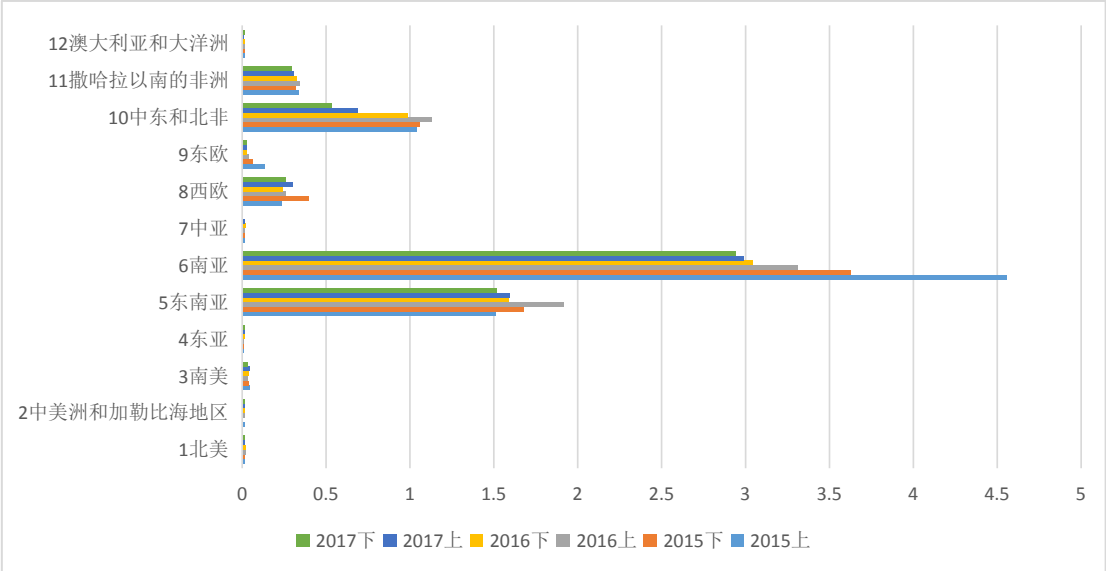

图 14　近三年各地区发生恐怖袭击的密度

7.2.4 级别分布

根据问题一求解的危险度及级别，衡量不同空间和时间的级别分布。对于空间，分别求出近三年 130 个国家受到所有恐怖袭击的平均危险度，然后根据危险度定出恐怖事件所在的级别。得出 130 个国家中，有 102 个属于级别 4，剩余 28 个属于级别 5，说明绝大部分恐怖袭击都属于不太严重的袭击，国家级别分布情况如图 15 所示。

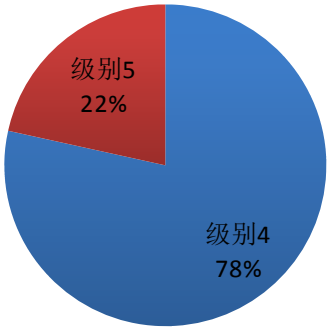

图 15 国家级别分布情况

对于时间，分别求出每年及每个月恐怖袭击事件的平均危险度，并根据危险度划分级别，得出不同时间的平均级别都是 4，与空间划分结果基本一致。

7.2.5 全球反恐态势预测

将全球划分为 12 个区域，分别以半年为一个时间区间，计算出 12 个区域近三年来共 6 个时间区间的恐怖袭击密度，结果见表 17。

表 17 2015 至 2017 年 12 大区域的恐怖袭击密度

地区	2015 上半年	2015 下半年	2016 上半年	2016 下半年	2017 上半年	2017 下半年
1	0.0058	0.0161	0.0087	0.0177	0.0161	0.0124
2	0.0036	0.0000	0.0073	0.0036	0.0036	0.0073
3	0.0423	0.0373	0.0345	0.0412	0.0451	0.0356
4	0.0104	0.0112	0.0016	0.0048	0.0032	0.0024
5	1.5103	1.6790	1.9136	1.5885	1.5967	1.5185
6	4.5596	3.6242	3.3131	3.0404	2.9899	2.9414
7	0.0075	0.0150	0.0150	0.0200	0.0125	0.0000
8	0.2360	0.3960	0.2600	0.2420	0.3020	0.2580
9	0.1337	0.0622	0.0364	0.0299	0.0270	0.0264
10	1.0372	1.0595	1.1297	0.9897	0.6868	0.5323
11	0.3387	0.3181	0.3440	0.3276	0.3086	0.2971
12	0.0036	0.0036	0.0030	0.0030	0.0012	0.0048

使用灰色预测模型分别对 12 个区域的时间序列做预测（程序见附程序 5），得到下一年各地区恐怖袭击密度的预测值及预测残差值，残差的绝对值大部分小于 0.1，模型预测性能较好。

表 18 恐怖袭击密度的预测值

地区	2018 上半年	2018 下半年	残差
1	0.090969	0.105168	0.0019
2	0.034066	0.04456	-0.0027
3	0.276804	0.318397	-0.0001
4	0.003379	0.003439	0.0021
5	1.428558	1.271027	-0.1077
6	2.515829	2.371064	0.0891
7	0.077758	0.083411	-0.0030
8	0.211372	0.191705	0.0516
9	0.032925	0.034141	0.0060
10	0.598013	0.564391	-0.1175
11	0.223119	0.25213	-0.0163
12	0.022652	0.026099	0.0006

根据原数据和预测数据，做出 12 个地区 2015 至 2018 年恐怖袭击密度图，如图 16 所示。

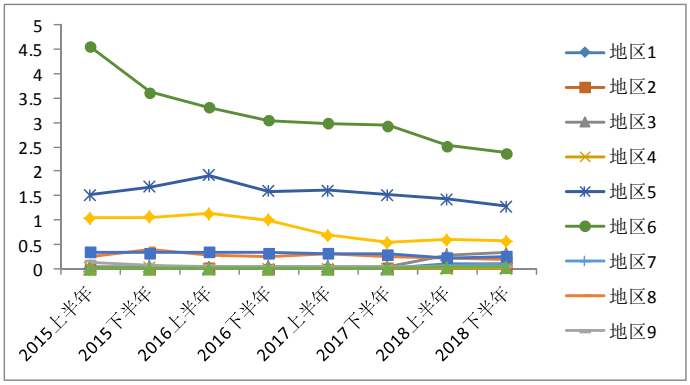

图 16 恐怖袭击密度图

可以看出各地区的恐怖袭击密度整体呈现下降或平稳的趋势，但地区 1, 2, 3, 4, 7, 10, 12 的恐怖袭击密度略有上升，相对需要反恐者提供更多关注。

§8 问题四的解答

8.1 研究问题

恐怖袭击事件的多特征性以及恐怖袭击事件造成后果的严重性，迫切需要对恐怖袭击事件的威胁程度进行评估，从而为反恐决策者提供决策支持，以期尽可能的减少恐怖袭击所造成的影响。如何量化在恐怖袭击发生前的措施是否有效，以此为反恐决策提供参考的理论依据。此外，如何量化发生恐怖袭击前恐怖主义者的动作对恐袭事件威胁度的影响，值得进一步地研究探讨。

考虑到在发生恐怖袭击事件前恐怖分子和民众的反应，以恐怖分子是否在恐怖活动前威胁、恐吓或向大众传播某种恐怖消息和民众或决策人员对恐怖袭击的了解程度两个变量为例，量化这两个变量在不同状态下，恐怖袭击事件的威胁程度的变化情况。

8.2 模型的建立

8.2.1 贝叶斯网络模型

贝叶斯网络又被称为信度网络，是*Bayes*方法的扩展，已经成为目前不确定知识表达以及推理领域最有效的理论模型之一。*Pearl*于 1988 年提出后，已经迅速成为近些年的研究热点。一个贝叶斯网络由一个有向无环图(*Directed Acyclic Graph,DAG*),代表变量节点及连接这些节点的有向边构成。节点表示随机变量，节点间连接的有向边代表了节点间的相互关系(由父节点指向其子节点)，之间的强度关系用条件概率来表示，如果一个节点没有父节点，一般用先验概率来表达信息。节点变量可以是任何问题的抽象，如:测试值，观测现象，意见征询等。贝叶斯网络模型适用于表达和分析不确定性和概率性的事件，应用于有条件地依赖多种控制因素的决策，可以从不完全、不精确或不确定的知识或信息中做出推理。*Allanach*等[6-7]曾指出贝叶斯网络是进行恐怖袭击信息整合的有效手段。

本文将贝叶斯网络模型分成 4 个模块，分别为评估恐怖袭击事件威胁程度模块，财产损失程度模块，人员伤亡程度模块，不良社会影响模块。其网络结构如图 17 所示：

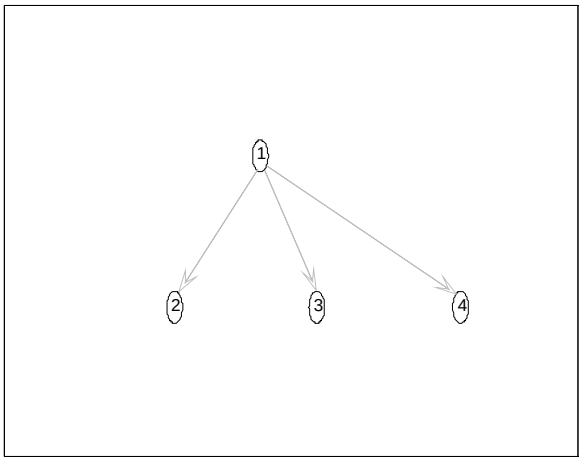

图17 恐怖袭击事件威胁程度网络图

其中各节点数字所代表的变量如表19所示：

表19 各节点信息

数字符号	变量名称
①	恐怖袭击事件威胁程度
②	财产损失程度
③	人员损失程度
④	不良社会影响程度

8.2.2 K2 算法

K2 算法是贝叶斯结构学习的经典算法之一，也是一种贪婪搜索算法。开始时每一个起始点没有父节点，然后增加结果结构打分最高时的父节点。当单独添加父节点再不能提高分数时，停止添加父节点。当我们使用固定的顺序时，我们不需要做循环检查，也不需要为每个节点单独选择父节点。其算法原理如下[8]：

假设 $X=(X_1,X_2,\cdots X_n)$， X_i 表示网络节点， X 表示节点集合，以 $\mathrm{pa}(X_i)$ 表示 X_i 父节点集合。对于任意 $1\le i<j\le n$， $X_j\notin\{X_t \mid X_t\in \mathrm{pa}(X_i),1\le t\le n\}$；

$$score(X_i,pa(X_i))=\prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{(r_i-1)!}{(N_{ij}+r_i-1)!}N_{ij}\prod_{k=1}^{r_i}N_{ijk}! \tag{7-1}$$

$$P(D\mid B_S)=\prod_{i=1}^{n}score(X_i,pa(X_i)) \tag{7-2}$$

其中 N_{ijk} 表示节点 X_i 的父节点 X_j 处于第 k 种状态的个数。 r_i 表示节点 X_i 的状态数。 N_{ij} 是节点 X_i 的父节点 X_j 所有状态数之和，其中 $N_{ij}=\sum_{k=1}^{r_i}N_{ijk}$.

8.2.3 联结树传播算法

联结树算法(*junction tree algorithm* , *clique-tree algorithm*)是 *Lauritzen* 和 *Spiegelhalter* 于 1988 年提出的[9] 。

该算法首先将贝叶斯网络转换为一个联结树(联结树是一个无向树，每个树节点是无向图的称为团的最大全连通子图)，然后通过消息传递来进行计算，消息会依次传遍联结树的每个节点,最终使联结树满足全局一致性[10]。算法过程如下：

1. 将有向图转换为无向图：

将每个有共同子节点的父节点连接起来，把所有的有向边改为无向边；

2. 将无向图三角化：

三角化问题，是指让图中不存在超过 3 个点的环。如果存在，则需要通过增加边来解决。理想的三角化问题是指，通过增加最少的边来达到三角化的目的。然而这个问题是 NPC 的。三角化问题的解决比较简单，指定一个节点顺序，然后查看与这个节点相连的几个节点，是否属于三角区域。如果不属于的话，那么判断他们之间是否存在边。如果不存在则增加边将他们连接即可。三角化的结果是否简单，取决于查看节点的顺序；

3. 将三角化的图转换为树：

每个三角都代表了一个节点。两个相临的三角具有共同的边。这条边就成为两个节点之间的中间节点。如此就组成一张联通图；

4. 寻找这张图的根，并寻找最大生成树，从而得到最终的结果。

下面就是在这颗树上进行若干次消息传递，从而完成数值的更新过程：

对于每个 *evidence variable*，把它放到包括这个变量的表里，然后把所有不满足这个 *evidence* 的 *entry* 全设为 0。接着做一个自底向上的迭代，对于每个叶子节点，给它的父节点发送一个信息，即相关的表，父节点得到信息后就将其跟自己的表相乘，依次往上迭代，直到根节点。之后再做一个自顶向下的迭代，类似的，父节点向子节点发送信息，子节点得到信息后将其与自己的表相乘，依次往下迭代，直到所有叶子节点收到信息。

8.2.4 威胁度计算

1. 根据已知特征信息 $\left[X_1, X_2, \cdots X_i\right]$ $(\mathrm{i} < \mathrm{n})$ ，分别对财产损失程度模块和人员伤亡程度模块的贝叶斯网络结构，调用联结树传播算法计算得到财产损失和人员伤亡不同程度下的后验概率。

2. 利用第一步得到的后验概率以及特征信息 $\left[X_{\mathrm{j}}, \cdots X_{\mathrm{n}}\right]$ $(\mathrm{i} < \mathrm{j} \leq \mathrm{n})$，对不良社会影响模块的贝叶斯网络结构调用联接树传播算法得到不良社会影响不同程度下的后验概率.

3. 给定恐怖袭击事件威胁程度的先验信息，将其不同程度下的概率初始化，当子节点诊断信息发生变化，更新贝叶斯网络结构，最后得到给定特证信息 $\left[X_1, X_2, \cdots X_{\mathrm{n}}\right]$ 时威胁程度的不同程度下的概率。

4. 给定恐怖袭击事件威胁程度的一个模糊效用值，乘以不同程度下的概率得到威胁度值。

8.3 模型的求解

8.3.1 数据的处理

使用问题一筛选指标之后的数据，并剔除数据大量缺失的指标，选取出地区（*region*）、地理编码特征(*specificity*)、成功的攻击（*success*）、攻击类型（*attacktype1*）、目标/受害者类型（*targtype1*）、武器类型（*weapontype1*）、受害者死亡人数（*nkill*）、受害者受伤人数（*nwound*）、财产损害程度（*propextent*）共 9 个指标。考虑到 MATLAB 软件的索引需为正整数，所以将成功的攻击（*success*）中的 0 字段替换为 2，考虑到数据的一致性，将受害者死亡人数（*nkill*）、受害者受伤人数（*nwound*）按表 20 进行量化：

表 20　变量量化

条件	处理值
$nkill = 0$ and $\text{nwound} = 0$	1
$1 \le nkill < 3$ or $1 \le \text{nwound} < 10$	2
$3 \le nkill < 10$ or $10 \le \text{nwound} < 50$	3
$10 \le nkill < 30$ or $50 \le \text{nwound} < 100$	4
$nkill \ge 30$ or $\text{nwound} \ge 100$	5

考虑到程度次序的一致性，将财产损害程度（*propextent*）按 1-4,2-3,3-2,4-1 处理，从 1-4，损害程度逐级上升。（数据见附件表 4.1）。

8.3.2 财产损失模块

选取出地区（*region*）、地理编码特征(*specificity*)、成功的攻击（*success*）、攻击类型（*attacktype1*）、目标/受害者类型（*targtype1*）、武器类型（*weapontype1*）、财产损害程度（*propextent*）作为网络节点变量，指定节点的先验顺序[2],利用K2算法构建网络拓扑结构，借助MATLAB编程(程序见附程序6)，其结果如图18所示

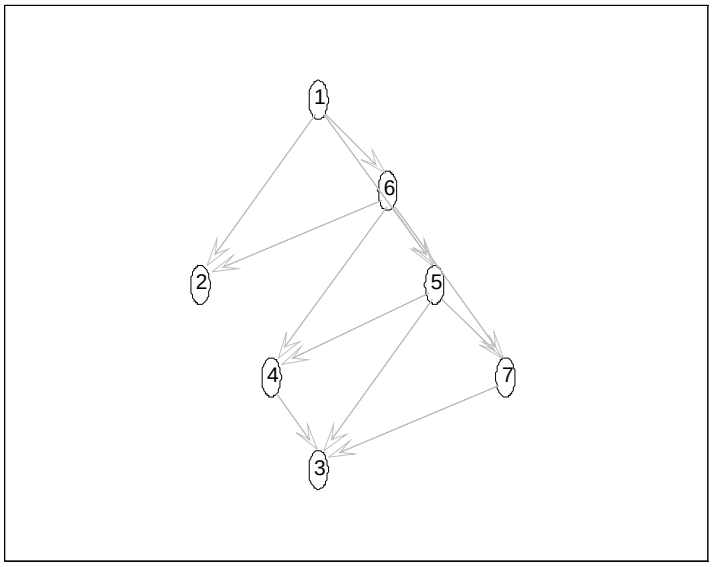

图18 财产损失模块网络结构

其中各节点数字所代表的变量如表21所示：

表21 各节点信息

数字符号	变量名称
①	地区（*region*）
②	地理编码特征(*specificity*)
③	成功的攻击（*success*）
④	攻击类型（*attacktype1*）
⑤	目标/受害者类型（*targtype1*）
⑥	武器类型（*weapontype1*）
⑦	财产损害程度（*propextent*）

构建出网络拓扑结构之后，利用联结树传播算法，结合整理之后的数据(数据见附文本4.2)，借助MATLAB编程得到给定特征下的财产损害程度的后验概率

(程序见附程序6)。以上述变量依次取1,1,1,4,11,10,为例，求得财产损害不同程度在上述特征下的概率依次为0.1584，0.3166，0.2715，0.2534．

8.3.3 人员伤亡模块

选取出地区（*region*）、地理编码特征(*specificity*)、成功的攻击（success）、攻击类型（*attacktype1*）、目标/受害者类型（*targtype1*）、武器类型（*weapontype1*）、量化后的伤亡程度作为网络节点变量，指定节点的先验顺序[2],利用K2算法构建网络拓扑结构，借助MATLAB编程(程序见附程序7)，其结果如图19所示

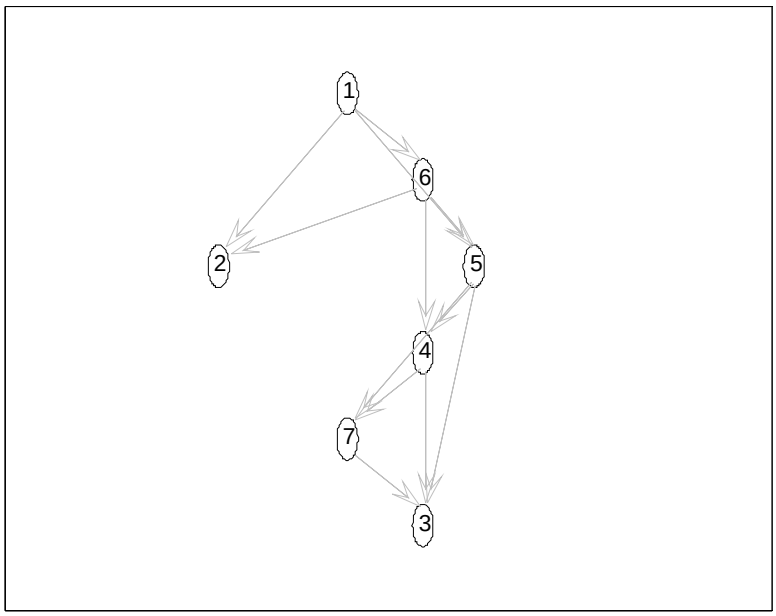

图19 人员伤亡模块网络结构

其中各节点数字所代表的变量如表22所示：

表22 各节点信息

数字符号	变量名称
①	地区（*region*）
②	地理编码特征(*specificity*)
③	成功的攻击（*success*）
④	攻击类型（*attacktype1*）
⑤	目标/受害者类型（*targtype1*）
⑥	武器类型（*weapontype1*）
⑦	量化后的人员伤亡程度

构建出网络拓扑结构之后，利用联结树传播算法，结合整理之后的数据(数据见附文本4.3），借助MATLAB编程得到给定特征下的财产损害程度的后验概率(程序见附程序7）。以上述变量依次取1,1,1,4,11,10, 为例，求得人员伤亡不同程度在上述特征下的概率依次为0.7825，0.1304，0.0870，0.0000，0.0000.

8.3.4 不良社会影响模块

该模块并没有提供样本数据，所以无法学习得到条件概率矩阵，这里应用主观赋权的方法，聘请专家来给出条件概率矩阵，本文借鉴魏静，王菊韵，于华[2]所构建网络图和条件概率矩阵，网络图如图20所示，具体的条件概率矩阵可以参见附录表2。

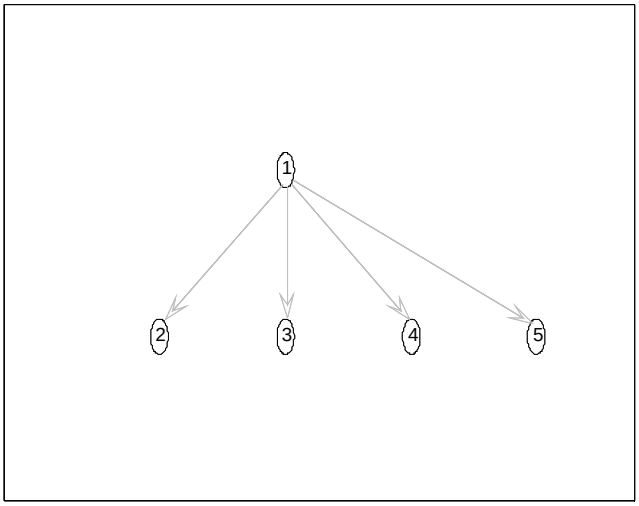

图20 不良社会影响模块网络结构

其中各节点数字所代表的变量如表23所示：

表23 各节点信息

数字符号	变量名称
①	不良社会影响程度
②	财产损失程度
③	人员伤亡程度
④	恐怖分子是否在恐怖活动前威胁、恐吓或向大众传播某种恐怖消息（*terrorist atmosphere*）
⑤	民众或决策人员对恐怖袭击的了解程度

这里财产损失程度和人员伤亡程度的类别定义为所关联的节点特征给定情况下最大后验概率所对应的类别，给定恐怖分子是否威胁、恐吓或传播消息及决策人员对恐怖袭击事件的了解程度，借助MATLAB编程得到不良社会影响在不同程度下的概率(程序见附程序8)。以上述变量依次取1,1,1,4,11,10, 2,1为例，求得不良社会影响不同程度在上述特征下的概率依次为0.2694，0.0962，0.4704，0.1639.

8.3.5 恐怖袭击事件威胁程度模块

网络结构已由图 17 给出。这里同样借鉴魏静，王菊韵，于华[2]所构建网络图和条件概率矩阵，具体的条件概率矩阵可以参见附录表 3，在给定具体的特征信息后，利用选取概率最大化对应类别准则，借助 MATLAB 编程可以求得不同程度下的概率(程序见附程序 9)。假设模糊效用值按程度递增依次为 0.25,0.50,0.75,1.00，以上述变量依次取 1,1,1,4,11,10,2,1 为例，求得恐怖袭击事件威胁程度在不同程度下的概率为 0.1383, 0.1197,0.7021,0.0399.威胁度为 0.6609.

如果改变倒数第二个的特征信息，使其从 2 变为 1，即恐怖分子在恐怖活动前威胁、恐吓或向大众传播某种恐怖消息，此时概率为 0.0544,0.0377,0.5523,0.3556，威胁度增加为 0.8023。说明恐怖分子在恐怖活动前威胁、恐吓或向大众传播某种恐怖消息会增加恐怖袭击事件的威胁程度，与实际情况相符。

如果改变最后一个特征信息，从 1 变为 4，即民众或决策人员对恐怖事件信息很了解，此时概率为 0.9633,0.0195,0.0163,0.0009.威胁度为 0.2637。显然威胁度是下降的，说明随着民众或决策人员对恐怖事件信息了解程度的加深，恐怖袭

击事件的威胁度下降，符合实际情况，在一定程度上说明了我们模型的合理性，因此对民众普及恐怖主义方面的知识很必要的。

§9 模型的评价与推广

9.1 模型的优点

优点一 ：本文构建的量化分级模型中，在得到每个袭击事件的危害度之后对数据进行分级时，创新性地引入在险值(VaR)的概念，根据危害度的取值和频数分布情况进行分类，分类结果更为科学；

优点二 ：采用遗传算法和模拟退火算法两种智能算法相结合的聚类方法，模拟退火算法和遗传算法互相取长补短，能有效克服传统遗传算法的早熟现象，能够更有效、快速地收敛到全局最优解，体现了智能算法的理论价值和现实意义；

优点三 ：支持向量机分类算法中，本文随机选取 75%的数据作为训练集，剩余数据作为验证集，对验证集的预测准确度进行测试，说明分类算法的准确度，对该算法的参数进行灵敏度分析，找出不同参数组合对模型效果的影响，有助于确定参数的大致范围，提升网格搜索进行参数寻优的效率；

优点四：本文进一步地利用数据库信息和前面的研究结果，更深层次地考虑到在反恐斗争中的应用问题，构建多模块贝叶斯网络模型，量化恐怖袭击事件发生前反恐措施的有效性以及恐怖主义者在进行恐怖袭击前的动作对恐怖袭击事件威胁度的影响。

9.2 模型的缺点

缺点一 ：由于世界各地举行重大活动的日期数据难以获取，本文没有考虑到恐怖袭击发生的时间因素的影响。在后文中，本文提出了相应的改进思路；

缺点二 ：由于凶手信息不全，如凶手数量存在大量缺失值，在问题二的聚类、分类中直接将此类指标删除，可能会影响模型准确度。

9.3 模型的推广

1. 基于遗传模拟退火算法的 FCM 聚类算法——遗传算法和模拟退火算法相结合提高了寻优的效率和准确度，可以将其应用到其他的聚类问题中。

2.支持向量机分类算法——可以解决高维和非线性问题，并且能够避免神经网络结构选择和局部极小点问题，可应用于人脸识别、图像识别等多个领域。

3.多模块贝叶斯网络模型——该模型不但能够集成专家的经验和知识，还能利用不同阶段得到的多个字段的信息，进行多阶段的评估，时刻掌握恐怖袭击事件的发展态势，可将其应用于恐怖袭击事件的应急决策过程中，提高智能化及决策的有效性。

§10 模型的改进

对于问题一，现有文献指出[11]，恐怖分子在袭击时间选择上，更倾向于在一些国家或政府举行重大活动期间，或重要的传统节日前后发动袭击。按照不同的国家地区，统计历年各国举行重大活动的日期，预留活动前 10 天的区间，与数据库恐怖袭击事件发生的日期匹配，在区间内取值为 1。将时机因素加入到模型的指标体系下，对社会恐慌程度有了更合理精确的度量，有助于提高模型结果的准确度。

对于问题二，采用均值漂移聚类算法等不需要预先输入聚类数的算法对样本聚类，将得到的聚类数与本文调整的 FCM 聚类数做对比，通过两种算法的比较对最优聚类数做出改进；采用多种分类算法，如随机森林算法、K-近邻（KNN）算法与本文的支持向量机算法作对比，判断哪种算法最适合用于多分类问题；采用不同的指标，如增加自杀式袭击指标，更多的反映凶手信息，比较不同指标组合下的聚类情况和分类准确度。

参考文献

[1]王锂达. 恐怖组织行为挖掘与预测[D].北京邮电大学,2017.

[2]魏静,王菊韵,于华.基于多模块贝叶斯网络的恐怖袭击威胁评估[J].中国科学院大学学报, 2015, 32(02): 264-272.

[3]周扬凡,汪彤,代宝乾.地铁既有线路运营风险量化分级方法研究[J].中国安全科学学报,2013,23(03):103-108.

[4]史峰,王辉,郁磊等. MATLAB智能算法30个案例分析[M].北京:北京航空航天大学出版社, 2011.

[5]王文剑,梁志,郭虎升.基于数据关系的SVM多分类学习算法[J].山西大学学报(自然科学版),2012,35(02):224-230.

[6]Allanach J, Tu H, Singh S, et al. Detecting, tracking, and counteracting terrorist networks via hidden Markov models[C]// Aerospace Conference, 2004. Proceedings. IEEE, 2004:3257 Vol.5.

[7]Singh S, Pattipati K R, Willett P, et al. Stochastic modeling of a terrorist event via the ASAM system.[J]. 2004, 6(6):5673-5678.

[8]Cooper G F, Herskovits E. A Bayesian method for the induction of probabilistic networks from data[J]. Machine Learning, 1992, 9(4):309-347.

[9]Lauritzen S L, Spiegelhalter D J. Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems[J]. Journal of the Royal Statistical Society, 1988, 50(2):157-224.

[10]厉海涛,金光,周经伦,周忠宝,李大庆.贝叶斯网络推理算法综述[J].系统工程与电子技术,2008(05):935-939.

[11]周文峰,陈菊娟.大型活动恐怖袭击风险及其治理路径[J].领导科学,2017(17):32-34.

附　　录

附程序 1　基于遗传模拟退火算法的FCM聚类

```
clc
clear all
close all
X=load('julei2.txt');
for i = 1 : size(X, 2)
    temp = X(:, i);
    temp = (temp-min(temp))/(max(temp)-min(temp));
    X(:, i) = temp;
end
m=size(X,2);
lb=min(X);
ub=max(X);
%% 模糊C均值聚类参数
% 设置幂指数为3，最大迭代次数为20，目标函数的终止容限为1e-6
options=[3,20,1e-6];
%类别数cn
cn=10;
%% 模拟退火算法参数
q =0.8;
T0=100;
Tend=1;
%% 定义遗传算法参数
sizepop=10;
MAXGEN=50;
NVAR=m*cn;
PRECI=10;
pc=0.7;
pm=0.01;
trace=zeros(NVAR+1,MAXGEN);
FieldD=[rep([PRECI],[1,NVAR]);rep([lb;ub],[1,cn]);rep([1;0;1;1],[1,NVAR])];
Chrom=CRTBP(sizepop, NVAR*PRECI);
V=BS2RV(Chrom, FieldD);
ObjV=ObjFun(X,cn,V,options);
T=T0;
while T>Tend
    gen=0;
    while gen<MAXGEN
        %分配适应度值
        FitnV=RANKING(ObjV);
        SelCh=SELECT('SUS', Chrom, FitnV);
        SelCh=RECOMBIN('XOVSP', SelCh,pc);
        SelCh=MUT(SelCh,pm);
```

```
            V=BS2RV(SelCh, FieldD);
            newObjV=ObjFun(X,cn,V,options);
            newChrom=SelCh;


            for i=1:sizepop
                if ObjV(i)>newObjV(i)
                    ObjV(i)=newObjV(i);
                    Chrom(i,:)=newChrom(i,:);
                else
                    p=rand;
                    if p<=exp((newObjV(i)-ObjV(i))/T)
                        ObjV(i)=newObjV(i);
                        Chrom(i,:)=newChrom(i,:);
                    end
                end
            end
            gen=gen+1; %代计数器增加
    [trace(end,gen),index]=min(ObjV);
            trace(1:NVAR,gen)=V(index,:);
            fprintf(1,'%d ',gen);
        end
        T=T*q;
        fprintf(1,'\n温度:%1.3f\n',T);
end
%计算最佳初始聚类中心的目标函数值
[newObjV,center,U]=ObjFun(X,cn,[trace(1:NVAR,end)]',options);
%查看聚类结果
Jb=newObjV
U=U{1};
center=center{1};
maxU = max(U);
index1 = find(U(1,:) == maxU);
index2 = find(U(2, :) == maxU);
index3 = find(U(3, :) == maxU);
index4 = find(U(4, :) == maxU);
index5 = find(U(5,:) == maxU);
index6 = find(U(6, :) == maxU);
index7 = find(U(7, :) == maxU);
index8 = find(U(8, :) == maxU);
index9 = find(U(9, :) == maxU);
index10 = find(U(10, :) == maxU);
```

附程序 2 支持向量机分类

```
#coding=utf-8
import numpy as np
import pandas as pd
from sklearn import svm,preprocessing,metrics
from sklearn.model_selection import train_test_split
import matplotlib as mpl
import matplotlib.pyplot as plt


def show_accuracy(a, b, tip):
    acc = a.ravel() == b.ravel()
    print('%s Accuracy:%.3f' %(tip, np.mean(acc)))

if __name__ == '__main__':

    a=[]
    b=[]
    # 加载数据
    data = pd.read_csv(r'C:\Users\cy\Desktop\svm_duofenlei.csv',
encoding='utf-8',header=None)
    yuce=pd.read_csv(r'C:\Users\cy\Desktop\svm_yuce.csv', encoding='utf-8',header=None)
    x, y = np.split(data, (10,), axis=1)
    x=preprocessing.scale(x)
    yuce=preprocessing.scale(yuce)

    print('---------------------------')

    x_train, x_test, y_train, y_test = train_test_split(x, y, random_state=1, train_size=0.75)

    # 高斯核
    clf1 = svm.SVC(C=6, kernel='rbf', gamma=0.2, decision_function_shape='ovr')
    # 线性核
    #clf = svm.SVC(C=0.5, kernel='linear', decision_function_shape='ovr')
    clf1.fit(x_train, y_train.values.ravel())

    # 中间结果的输出
    print('trianing prediction:%.3f' %(clf1.score(x_train, y_train)))
    # 预测值
    y_hat = clf1.predict(x_train)
    show_accuracy(y_hat, y_train.values, 'traing data')
    print('test data prediction:%.3f' %(clf1.score(x_test, y_test)))
    y_hat_test = clf1.predict(x_test)
    show_accuracy(y_hat_test, y_test.values, 'testing data')
    yuce_test = clf1.predict(yuce)
    # decision function
    a=clf1.decision_function(x_test)
    b=clf1.predict(x_test)
    c=clf1.decision_function(yuce)
    d=clf1.predict(yuce)


# with open(r'C:\Users\cy\Desktop\decision_function1.txt',"w",encoding='utf-8') as fw:
    # for data in a:
        # data=str(data)
        # data.encode('utf-8')
        # data=data.strip()
```

```
            # if len(data)!=0:
                # fw.write(data)
                # fw.write("\n")

# print ("end1")

# with open(r'C:\Users\cy\Desktop\predict1.txt',"w",encoding='utf-8') as fw:
    # for data in b:
            # data=str(data)
            # data.encode('utf-8')
            # data=data.strip()
            # if len(data)!=0:
                # fw.write(data)
                # fw.write("\n")

# print ("end2")

# with open(r'C:\Users\cy\Desktop\decision_function2.txt',"w",encoding='utf-8') as fw:
    # for data in c:
            # data=str(data)
            # data.encode('utf-8')
            # data=data.strip()
            # if len(data)!=0:
                # fw.write(data)
                # fw.write("\n")

# print ("end3")

# with open(r'C:\Users\cy\Desktop\predict2.txt',"w",encoding='utf-8') as fw:
    # for data in d:
            # data=str(data)
            # data.encode('utf-8')
            # data=data.strip()
            # if len(data)!=0:
                # fw.write(data)
                # fw.write("\n")

# print ("end4")
```

附程序 3 灵敏度分析

```
gamma=[0.01 0.1 1    10    100];
a1=[0.354    0.371    0.563    0.354    0.354];
a2=[0.582    0.939    0.964    0.812    0.564];
a3=[0.953    0.99     0.983    0.963    0.877];
a4=[0.995    0.994    0.995    0.983    0.941];
a5=[0.998    0.997    0.995    0.984    0.943];
a6=[0.999    0.997    0.995    0.984    0.943];
hold on
plot(gamma,a1,'-+',gamma,a2,'-*',gamma,a3,'->',gamma,a4,'-o',gamma,a5,'-',gamma,a6,'b*')
%plot(gamma,a4,'-o',gamma,a5,'->',gamma,a6,'-p')
gtext('C=0.001')
gtext('C=0.01')
gtext('C=0.1')
```

```
gtext('C=1')
gtext('C=10')
gtext('C=100')
```

附程序 4 PYTHON绘图

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
import plotly.plotly as py
py.sign_in('Aorigami', 'H64SFtVv5Kbd2rBGdIcK')
import pandas as pd


df = pd.read_csv(r'C:\Users\cy\Desktop\Country_KXCS.csv', encoding='utf-8')
data =[dict(
        type = 'choropleth',
        locations = df['CODE'],
        z = df['KXCS'],
        text = df['COUNTRY'],
        colorscale = [[0,"rgb(5,48,97)"],
            [0.15265666689933938,"rgb(35,48,97)"],
            [0.20531333379867875,"rgb(33,102,172)"],
            [0.30531333379867875,"rgb(33,80,162)"],
            [0.4079700006980182,"rgb(97,137,195)"],
            [0.4579700006980182,"rgb(67,147,195)"],
            [0.5279700006980182,"rgb(67,127,225)"],
            [0.6006266675973575,"rgb(146,197,222)"],
            [0.7032833344966969,"rgb(209,229,240)"],
            [0.7577361120805807,"rgb(247,247,247)"],
            [0.8021888896644646,"rgb(244,165,130)"],
            [0.85066416672483484,"rgb(244,145,130)"],
            [0.9010944448322323,"rgb(214,96,77)"],
            [0.9555472224161161,"rgb(103,0,56)"],
            [1,"rgb(103,0,31)"]],
        autocolorscale = False,
        # reversecale = True,
        marker = dict(
            line = dict(
                color = 'rgb(180,180,180)',
                width = 0.5,
            )),
        colorbar = dict(
            title = '恐怖袭击数量'),
)]
```

```
layout = dict(
    title = '每个国家恐怖袭击数量  (2015--2017年)',
    geo = dict(
        showframe=False,
        showcoastlines =True,
        projection = dict(
            type = 'mercator'
        )
    )
)

fig =dict(data = data,layout=layout)
py.plot(fig,validata=False,filename='world_map')
```

附程序 5 GM(1,1)模型

```
%x0=[0.0057784380.016097078   0.008667657   0.01774806    0.016097078   0.012382368];
A=load('yuce.txt');
A1=A(1,:);
A2=A(2,:);
A3=A(3,:);
A4=A(4,:);
A5=A(5,:);
A6=A(6,:);
A7=A(7,:);
A8=A(8,:);
A9=A(9,:);
A10=A(10,:);
A11=A(11,:);
A12=A(12,:);
x0=A6;
n=length(x0);
lamda=x0(1:n-1)./x0(2:n);
range=minmax(lamda);
x1=cumsum(x0);
for i=2:n
    z(i)=0.5*(x1(i)+x1(i-1));
end
B=[-z(2:n)',ones(n-1,1)];
Y=x0(2:n)';
u=B\Y;
x=dsolve('Dx+a*x=b','x(0)=x0');
x=subs(x,{'a','b','x0'},{u(1),u(2),x1(1)});
```

```
yuce1=subs(x,'t',[0:n-1]);    %yuce1=subs(x,'t',[0:n+2]);
digits(6),y=vpa(x);
yuce=[x0(1),diff(yuce1)];
epsilon=x0-yuce;
delta=abs(epsilon./x0);
rho=1-(1-0.5*u(1))/(1+0.5*u(1))*lamda;
h1=mean(delta);
```

附程序 6 财产损失模块

```
n=7;
ns=[12,5,2,9,22,13,4];
names={'x1','x2','x3','x4','x5','x6','x7'};
x1=1;x2=2;x3=3;x4=4;x5=5;x6=6;x7=7;
order=[1 6 2 5 4 7 3];
result_matrix=zeros(ns(x7),ns(x7));
max_fan_in=2;
data=load('C:\Users\Administrator\Desktop\text1.txt');
[num_n,num_m]=size(data');
data_train=zeros(num_n,num_m);
dag=zeros(n,n);
dag=learn_struct_K2(data',ns,order,'max_fan_in',max_fan_in);
bnet=mk_bnet(dag,ns);
draw_graph(dag);
%下面是参数学习
priors=1;
seed=0;
rand('state',seed);
for i=1:n

bnet.CPD{i}=tabular_CPD(bnet,i,'CPT','unif','prior_type','dirichlet','dirichlet_type','BDeu','dirichle
t_weight',priors);
        % 采用全局联合树推理算法，提高运算速度
end
bnet2=bayes_update_params(bnet,data');
CPT3=cell(1,n);
for i=1:n
       s=struct(bnet2.CPD{i});
       CPT3{i}=s.CPT;
end
%给定证据，求最大后验概率
engine = jtree_inf_engine(bnet2);
evidence = cell(1,n);
  evidence{x1} = 1;
  evidence{x2} = 1;
  evidence{x3} = 1;
  evidence{x4} = 4;
  evidence{x5} = 11;
  evidence{x6} = 10;

  [engine, loglik] = enter_evidence(engine, evidence);
    marg = marginal_nodes(engine,x7);
M=marg.T
    m1=find(M==max(M))
```

附程序 7 人员伤亡模块

```
n=7;
ns=[12,5,2,9,22,13,5];
names={'x1','x2','x3','x4','x5','x6','x7'};
x1=1;x2=2;x3=3;x4=4;x5=5;x6=6;x7=7;
order=[1 6 2 5 4 7 3];
result_matrix=zeros(ns(x8),ns(x8));
max_fan_in=2;
data=load('C:\Users\Administrator\Desktop\text2.txt');
[num_n,num_m]=size(data');
data_train=zeros(num_n,num_m);
dag=zeros(n,n);
dag=learn_struct_K2(data',ns,order,'max_fan_in',max_fan_in);
bnet=mk_bnet(dag,ns);
draw_graph(dag);
%下面是参数学习
priors=1;
seed=0;
rand('state',seed);
for i=1:n

bnet.CPD{i}=tabular_CPD(bnet,i,'CPT','unif','prior_type','dirichlet','dirichlet_type','BDeu','dirichle
t_weight',priors);
        % 采用全局联合树推理算法，提高运算速度
end
bnet2=bayes_update_params(bnet,data');
CPT3=cell(1,n);
for i=1:n
       s=struct(bnet2.CPD{i});
       CPT3{i}=s.CPT;
end
%给定证据，求最大后验概率
engine = jtree_inf_engine(bnet2);
evidence = cell(1,n);
  evidence{x1} = 1;
  evidence{x2} = 1;
  evidence{x3} = 1;
  evidence{x4} = 4;
  evidence{x5} = 11;
  evidence{x6} = 10;

  [engine, loglik] = enter_evidence(engine, evidence);
    marg = marginal_nodes(engine,x8);
M=marg.T

m2=find(M==max(M))
```

附程序 8 不良社会影响模块

```
N = 5;
dag = zeros(N,N);
x1=1; x2=2; x3=3;x4=4;x5=5;
dag(x1,[x2 x3 x4 x5]) = 1;
draw_graph(dag);
discrete_nodes = 1:N;
node_sizes = [4 4 5 2 4];
bnet = mk_bnet(dag, node_sizes, 'discrete', discrete_nodes);
```

```
bnet.CPD{x1} = tabular_CPD(bnet, x1, [0.25 0.25 0.25 0.25]);
bnet.CPD{x2} = tabular_CPD(bnet, x2, [0.55 0.10 0.05 0.01 0.40 0.60 0.15 0.04 0.07 0.25 0.55
0.25 0.03 0.05 0.25 0.70]);
bnet.CPD{x3} = tabular_CPD(bnet, x3, [0.60 0.03 0.02 0.01 0.30 0.45 0.03 0.02 0.06 0.40 0.50
0.05 0.03 0.10 0.40 0.22 0.01 0.02 0.05 0.70]);
bnet.CPD{x4} = tabular_CPD(bnet, x4, [0.10 0.40 0.60 0.90 0.90 0.60 0.40 0.10]);
bnet.CPD{x5} = tabular_CPD(bnet, x5, [0.01 0.03 0.15 0.92 0.02 0.07 0.80 0.05 0.07 0.70 0.03
0.02 0.90 0.20 0.02 0.01]);
engine = jtree_inf_engine(bnet);
evidence = cell(1,N);

  evidence{x2} = m1;
  evidence{x3} = m2;
  evidence{x4} = 2;
    evidence{x5} = 1;

  [engine, loglik] = enter_evidence(engine, evidence);
    marg = marginal_nodes(engine,x1);
    M=marg.T
    m3=find(M==max(M))
```

附程序 9 恐怖袭击事件威胁程度模块

```
N = 4;
dag = zeros(N,N);
x1=1; x2=2; x3=3;x4=4;
dag(x1,[x2 x3 x4]) = 1;
draw_graph(dag);
discrete_nodes = 1:N;
node_sizes = [4 4 5 4];
bnet = mk_bnet(dag, node_sizes, 'discrete', discrete_nodes);
bnet.CPD{x1} = tabular_CPD(bnet, x1, [0.25 0.25 0.25 0.25]);
bnet.CPD{x2} = tabular_CPD(bnet, x2, [0.50 0.10 0.05 0.02 0.44 0.60 0.15 0.08 0.05 0.25 0.55
0.25 0.01 0.05 0.25 0.65]);
bnet.CPD{x3} = tabular_CPD(bnet, x3, [0.65 0.03 0.02 0.01 0.25 0.45 0.03 0.02 0.06 0.40 0.50
0.03 0.03 0.10 0.40 0.29 0.01 0.02 0.05 0.65]);
bnet.CPD{x4} = tabular_CPD(bnet, x4, [0.80 0.07 0.04 0.01 0.15 0.75 0.06 0.09 0.04 0.15 0.60
0.15 0.01 0.03 0.30 0.85]);
engine = jtree_inf_engine(bnet);
evidence = cell(1,N);

  evidence{x2} = m1;
  evidence{x3} = m2;
  evidence{x4} = m3;
    [engine, loglik] = enter_evidence(engine, evidence);
    marg = marginal_nodes(engine,x1);
    M=marg.T
```

附表 1 指标的“等效死亡”折算因子

地理编码特征	折算因子
1	4.6867
2	3.9404
3	3.1070
4	3.0702

5	4.2733

成功攻击与否	折算因子
0	0.6164
1	4.9504

攻击类型	折算因子
1	2.4446
2	3.7986
3	4.8850
4	57.4704
5	13.4136
6	2.8075
7	3.5169
8	4.2644
9	3.3820

武器类型	折算因子
1	1.2375
2	30.8775
3	0.3909
5	2.5719
6	7.2133
7	6.2500
8	3.9342
9	2.3558
10	370.2957
11	5.0095
12	2.3232
13	1.7912

附表 2 恐怖袭击事件威胁程度模块的贝叶斯网络条件概率矩阵

Threat level	$P(P\|T)$				$P(C\|T)$					$P(A\|T)$			
	1，	2，	3，	4	1	2	3	4	5	1	2	3	4
1	0.50	0.44	0.05	0.01	0.65	0.25	0.06	0.03	0.01	0.80	0.15	0.04	0.01
2	0.10	0.60	0.25	0.05	0.03	0.45	0.40	0.10	0.02	0.07	0.75	0.15	0.03
3	0.05	0.15	0.55	0.25	0.02	0.03	0.50	0.40	0.05	0.04	0.06	0.60	0.30
4	0.02	0.08	0.25	0.65	0.01	0.02	0.03	0.29	0.65	0.01	0.09	0.15	0.85

附表 3 不良社会影响模块的贝叶斯网络条件概率矩阵

Threat level	$P(P \mid A)$				$P(C \mid A)$					$P(TA \mid A)$		$P(DK \mid A)$			
	1	2	3	4	1	2	3	4	5	1	2	1	2	3	4
1	0.55	0.40	0.07	0.03	0.60	0.30	0.06	0.03	0.01	0.10	0.90	0.80	0.15	0.04	0.01
2	0.10	0.60	0.25	0.05	0.03	0.45	0.40	0.10	0.02	0.40	0.60	0.07	0.75	0.15	0.03
3	0.05	0.15	0.55	0.25	0.02	0.03	0.50	0.40	0.05	0.60	0.40	0.04	0.06	0.60	0.30
4	0.01	0.04	0.25	0.70	0.01	0.02	0.05	0.22	0.70	0.90	0.10	0.01	0.09	0.15	0.85