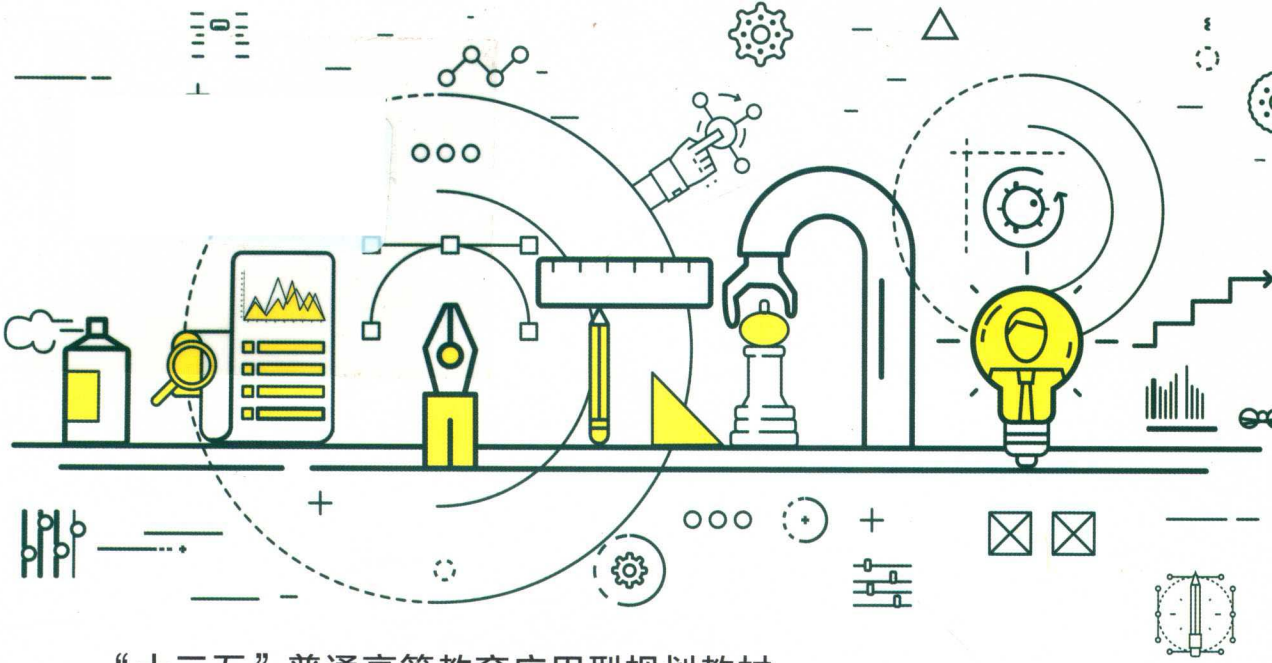


“十三五”普通高等教育应用型规划教材

College Mathematics Analysis and Its Applications

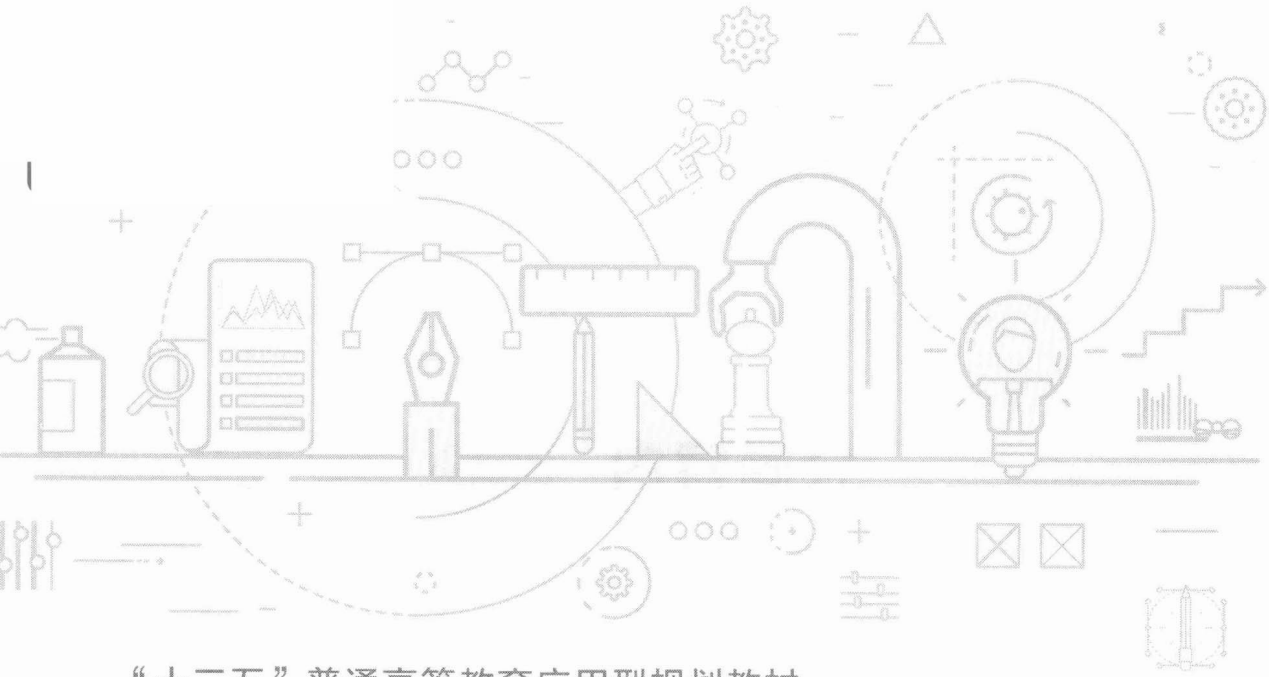
大学数学应用案例及分析

主　编　张丽梅

副主编　高胜哲　张立石　赵学达　张立峰　屈磊磊　顾剑

扫码下资源

中国人民大学出版社



“十三五”普通高等教育应用型规划教材

College Mathematics Analysis and Its Applications

大学数学应用案例及分析

主　编　张丽梅

副主编　高胜哲　张立石　赵学达　张立峰　屈磊磊　顾剑

中国人民大学出版社

· 北京 ·

图书在版编目（CIP）数据

大学数学应用案例及分析/张丽梅主编. —北京：中国人民大学出版社，2019.3
“十三五”普通高等教育应用型规划教材
ISBN 978-7-300-26705-0

Ⅰ.①大… Ⅱ.①张… Ⅲ.①高等数学-高等学校-教材 Ⅳ.①O13

中国版本图书馆 CIP 数据核字（2019）第 028555 号

“十三五”普通高等教育应用型规划教材

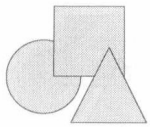
大学数学应用案例及分析

主　编　张丽梅

副主编　高胜哲　张立石　赵学达　张立峰　屈磊磊　顾剑

Daxue Shuxue Yingyong Anli ji Fenxi

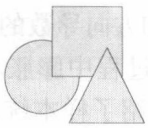
出版发行	中国人民大学出版社		
社　　址	北京中关村大街 31 号	**邮政编码**	100080
电　　话	010－62511242（总编室）	010－62511770（质管部）	
	010－82501766（邮购部）	010－62514148（门市部）	
	010－62515195（发行公司）	010－62515275（盗版举报）	
网　　址	http://www.crup.com.cn		
	http://www.ttrnet.com（人大教研网）		
经　　销	新华书店		
印　　刷	北京鑫丰华彩印有限公司		
规　　格	185 mm×260 mm　16 开本	**版　　次**	2019 年 3 月第 1 版
印　　张	10.25	**印　　次**	2019 年 3 月第 1 次印刷
字　　数	235 000	**定　　价**	25.00 元



内容简介

本书以工科类专业教材为素材，收集编写了若干建立在高等数学、线性代数、概率论和数理统计的理论与方法上的数学模型，力求为数学在工科类专业中的应用提供链接. 本书在编写上力求对问题的背景的说明简单明了，对模型的数学描述与数学表达深入浅出，同时尽量使各模型相对独立以供读者选读. 本书涉及信息计算中的几个概念，图形图像数学模型的矩阵表示方法，离散时间系统的状态空间、能控性与能观性的数学基础，线性系统中离散卷积的矩阵表示，声呐信号中主要的信号描述函数，机械设计中优化设计的数学模型，力学中扭转、弯曲应力的数学度量，不可压非黏性流体流动的基本方程，泛函的欧拉方程，加权余量法，几个集总电路基础元件约束关系的数学表示，温度场中梯度和方向导数计算问题，导热基本定律的三个方程，线性回归模型及其矩阵表示，总体主成分，正交因子模型等内容.

本书可作为工科类学生数学模型选修课教材，也可供有关专业工程技术人员参考.



前　言

随着科学的发展和技术的进步，目前国家设定的工科类应用型本科专业课程中涉及的数学问题越来越有深度. 为了在数学及其在工科专业课程中的应用之间搭建一座“桥梁”，将更多大学数学知识（以高等数学、线性代数、概率论与数理统计等为基础）在专业课程中的应用背景与算法原理分析透彻，我们收集了若干计算机类、电子类、机械类、力学类、热学能源类等相关专业中的基础数学模型，指出了其中的数学理论或算法原理，给出了其应用实例或解决问题的方法步骤，为更深入地理解这些问题所涉及的专业知识以及开阔数学应用的眼界均起到了一定的启发与激励作用.

全书共 10 章，第 1 章主要介绍了信息计算过程中信息熵、伪随机数概念；随机计算定积分算法，PageRank 网页排名算法，向量夹角用于新闻分类，奇异值分解的方法和应用场景等内容. 第 2、3 章主要介绍了矩阵表示数字图像的几何变换的常用方法；三维图形的几何变换方法；图像分析的数学模型基础——梯度计算及锐化，图像的增强，几何校正与特征分析；对静态图像的分割法——阈值法，背景差分、图像差分以及光流分割，人脸特征的简单分析方法. 第 4 章介绍了离散时间系统的状态空间，离散时间系统的能控性与能观性的数学基础；线性定常系统的时域响应与稳定性；线性系统中有重要作用的卷积定义；离散卷积的矩阵表示；主动声呐信号中主要的信号描述函数——时间函数、频谱函数以及模糊函数；信号的多普勒频移以及信号模糊函数的作用. 第 5 章介绍了机械平移系统、旋转体运动系统、电气系统的数学模型基础，机械设计中的优化设计问题，机械设计中的数学模型；机械零件与系统的可靠性设计、疲劳强度可靠性设计方法的数学模型. 第 6 章介绍了平面图形的几何性质，包括静矩和形心、惯性矩和惯性积、平行移轴公式、转轴公式和主惯性轴；轴向拉伸和压缩中的基础数学模型；扭转中的数学模型，包括薄壁圆筒的扭转的扭转角、等直圆杆扭转时的应变能等；弯曲应力中的数学模型，包括弯矩、剪力与分布荷载集度；梁弯曲变形时的数学度量，包括梁的位移——挠度及转角、梁的挠曲线近似微分方程及其积分. 第 7 章介绍了不可压非黏性流体流动的基本方程、速度势方程

和流函数方程；变分法、泛函的增量与泛函的变分以及欧拉方程；加权余量法的基本思想，以及不同权重设定的方法——配置法、子区域法、最小二乘法和矩法；静水中的扰动水压力分析. 第 8 章介绍了集总电路基础元件约束关系的数学原理与数学表示——电流、电容，*RC* 电路的零输入响应，静态电阻和动态电阻，幅值与有效值，电阻元件、电感元件、电容元件，理想变压器，回转器；互感元件、未知回路的电流与电压，基尔霍夫定律的数学表现形式，节点电压电流的矩阵形式，基本回路的 KVL、KCL 方程，电阻、电导的参数方程等内容. 第 9 章讨论了温度场中梯度和方向导数的计算问题，包括温度场、温度梯度、导热基本定律以及导热微分方程；可逆过程中膨胀功的计算，热力学微分方程式；几个涉及微分方程的传热学例子. 第 10 章介绍了样本统计中的几个数学模型，包括各类测量误差及其数据处理，观测数据的数字特征，线性回归模型及其矩阵表示，总体主成分，正交因子模型，参数的最大似然估计与牛顿迭代解法.

本书编写顺序如下：**张丽梅**（第 1、2、3 章），**高胜哲**（第 4 章第一节、第二节、第三节以及第 5 章第二节、第三节、第四节、第五节、第六节），**张立石**（第 9 章第三节，第 10 章），**赵学达**（第 4 章第四节、第五节，第 5 章第一节，第 6 章第一节，第 7 章第五节、第六节，第 9 章第一节、第二节），**张立峰**（第 7 章第一节、第二节、第三节、第四节），**屈磊磊**（第 6 章第二节、第三节、第四节、第五节，第 8 章第一节），**顾剑**（第 8 章第二节、第三节、第四节、第五节、第六节）.

本书可作为应用型工科专业本科数学应用类选修课程的教材，也是相关领域研究人员的数学类参考书.

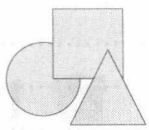
本书的素材来自书后所列参考文献，对所有参考书目的作者表示深深的敬意与感谢！

感谢中国人民大学出版社编辑人员的辛苦工作！

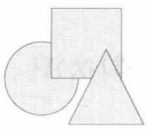
本书获得了辽宁省教育教学研究项目的支持和大连海洋大学教育教学项目的支持，在此表示感谢！

限于作者的水平和经验，书中肯定存在不少缺点和错误，殷切地希望读者批评指正.

2018 年 7 月



目　录



第 1 章　信息计算中的若干数学模型

本章讨论了信息计算过程中信息熵、伪随机数概念，分析了定积分的随机计算方法、PageRank 网页排名算法、向量夹角用于新闻分类以及奇异值分解的方法和应用场景等内容. 本章主要参考了文献 [1]、[2]、[3].

第一节　信息熵、伪随机数、定积分的随机计算

一、信息熵

人们常说信息量多、信息量少，这个“多”或“少”如何来度量呢？1948 年，香农 (Shannon) 提出了“信息熵”的概念，简称“熵”. 下面用猜宝游戏进行解释.

假设 1～32 号盒子中只有一个盒子装有一件宝物，人们每猜一次付费 1 元，那么需要付费多少才能找到宝物呢？找宝人每次以盒子总数的一半进行猜测.

找宝人提问：

1）宝物在 1～16 号盒子中吗？答：是，找宝人付费 1 元；

2）宝物在 1～8 号盒子中吗？答：不是，找宝人付费 1 元；

3）宝物在 9～12 号盒子中吗？答：是，找宝人付费 1 元；

4）宝物在 9～10 号盒子中吗？答：不是，找宝人付费 1 元；

5）宝物在 11 号盒子中吗？答：不是，找宝人付费 1 元；

找宝人断定：宝物在 12 号盒子中. 在这个过程中，回答人如实作答，不论答“是”或“不是”，都可以按这种方法找到宝物所在的盒子，共需付费 5 元钱.

当然，香农不是用钱，而是用“比特”这个概念来度量信息量. 一个比特是一位二进制数，计算机中一个字节是 8 比特. 信息量的比特数和所有可能情况的 log 有关，该 log

以 2 为底，$\log 32=5$，$\log 64=6$. 具体地

$$H=-(p_1\log p_1+p_2\log p_2+\cdots+p_{32}\log p_{32}),$$

其中 p_1，p_2，…，p_{32} 分别表示 32 个盒子装有宝物的概率. H 叫作信息熵(entropy)，单位是比特.

从概率的观点看，宝物在 1～32 号盒子的概率是等可能的，即

$$p_i=\frac{1}{32}\ (i=1,2,\cdots,32),\ \log p_i=\log\frac{1}{32}=-5\quad (i=1,2,\cdots,32),$$

故 $H=-32\times\frac{1}{32}\times\log\frac{1}{32}=5$（数学上可以证明 $H\leqslant 5$），即这个问题的信息熵是 5 比特. 试想，如果有 64 个盒子，猜宝的信息熵就是 6 比特，因为要多猜一次.

现将这个问题加以扩展，假设世界杯比赛有 32 支球队参赛，“宝物”指的是冠军. 根据历史经验，有一些球队弱，夺冠的概率很小；有一些球队强，夺冠的概率大. 参考这些信息得出 p_i（第 i 支球队夺冠的概率）的值不等，也就是能够确定的信息多了，这时 H 将小于 5. 也就是说，不确定性大，熵就大，相反，知道的信息多，熵就会变小.

综上所述，不难理解熵的一般定义.

对于任意一个随机变量 X，它的熵定义为

$$H(X)=-\sum_{x_i\in X}p(x_i)\log p(x_i). \tag{1.1}$$

当消息是等概率的时，$H(X)=\log N$，N 为随机变量 X 中包含的 x_i 的个数.

事实上，此时 $p(x_i)=\frac{1}{N}$，有

$$H(X)=-\sum_i\frac{1}{N}\log\frac{1}{N}=-\frac{N}{N}\log\frac{1}{N}=\log N.$$

例 1.1 求随机变量 X 的信息熵，其概率分布律为 $\left\{\frac{1}{2},\frac{1}{4},\frac{1}{8},\frac{1}{8}\right\}$.

解：所求熵为 $-(\frac{1}{2}\log\frac{1}{2}+\frac{1}{4}\log\frac{1}{4}+2\times\frac{1}{8}\log\frac{1}{8})=1.75$ 比特.

例 1.2 求二值变量的信息熵，假定随机变量以概率 $\{p,1-p\}$ 在集合 $\{0,1\}$ 上取值.

解：$H(X)=-[p\log p+(1-p)\log(1-p)]$. 当 $p=\frac{1}{2}$ 时，熵 $H(X)$ 最大为 1.

一般来说，当选择等概率时熵最大，当随机变量不再随机时熵为零（此时不存在不确定性）. 熵的一个重要属性是相互独立的随机变量的熵具有可加性.

信息熵是信息论中用于度量信息量的一个概念，解决了对信息的量化度量问题. 一个系统越有序，信息熵就越低；反之，一个系统越混乱，信息熵就越高. 所以，信息熵也可以说是系统有序化程度的一个度量. 熵曾经是热力学第二定律引入的概念，可以把它理解为分子运动的混乱度，信息熵也有类似意义.

有时候为了消除信息的不确定性，还利用相关的信息来分析，为此，引入条件熵的概念.

假定 X 和 Y 是两个随机变量，X 是需要了解的.

假定知道了 X 的随机分布 $p(x)$，则 X 的熵为

$$H(X)=-\sum_{x\in X}p(x)\log p(x).$$

现在假如还知道 Y 与 X 的联合概率分布 $p(x,\ y)$，就可以定义在 Y 条件下 X 的条件熵

$$H(X\mid Y)=-\sum_{x\in X,y\in Y}p(x,\ y)\log p(x\mid y). \tag{1.2}$$

可以证明 $H(X)\geqslant H(X\mid Y)$，即增加了 Y 的信息，关于 X 的不确定性下降了.

总之，信息熵是对一个信息系统不确定性的度量，信息熵是整个信息论的基础，对于通信、数据压缩、自然语言处理都有很强的指导意义.

二、伪随机数

随机数在概率算法设计中十分重要，实际中计算机无法产生真正的随机数，因此在概率算法中使用的随机数都是在一定程度上随机，即伪随机数.

产生伪随机数最常用的方法是线性同余法. 由线性同余法产生的随机序列

$$a_1,\ a_2,\ \cdots,\ a_n,\ \cdots\text{满足}\begin{cases}a_0=d\\ a_n=(ba_{n-1}+c)\bmod m,\ n=1,\ 2,\ \cdots\end{cases}$$

其中 $b\geqslant 0$，$c\geqslant 0$，$d\geqslant m$. d 称为随机序列的种子. 如何选取 b，c，m 直接关系到所产生的随机序列的随机性能. 数论理论可以证明，对于模数 $m=2^L$，当 $b=4k+1$（k 为正整数）且 c 与 m 互素时，也就是两者的最大公约数为 1 时，可以获得最长随机数序列长度为 2^L. 这样如果需要更多的伪随机数，当 m 越大时，随机性能越好.

当 $c=0$ 时，线性同余法就称为乘同余法. 下面举例说明乘同余法，数论理论可以证明，对于模数 $m=2^L$，当 $b=8k\pm 3$ 或者 $b=4k+1$（k 为正整数）且 a_0 为奇数时，乘同余法可以获得最长随机数序列长度为 2^{L-2}. 这里，取 $m=2^6=64$，$b=13$，选取种子 $a_0=1$，可以通过简单计算得到 1～64 之间非重复长度达到 $2^4=16$ 的均匀分布伪随机序列如下.

$$\{1,\ 13,\ 41,\ 21,\ 17,\ 29,\ 57,\ 37,\ 33,\ 45,\ 9,\ 53,\ 49,\ 61,\ 25,\ 5,\ 1\}$$

这就取到了区间 $[1,\ 64)$ 上的均匀分布伪随机序列.

这个序列是这样获得的：

$a_0=1$，$b=13$，$a_1=ba_0\bmod 64=13$，

$a_2=ba_1\bmod 64=13\times 13\bmod 64=(2\times 64+41)\bmod 64=41$，

$a_3=ba_2\bmod 64=13\times 41\bmod 64=(8\times 64+21)\bmod 64=21$，依此类推.

在各种智能算法中随机数是必不可少的基本要素，计算机中许多语言都提供了伪随机数发生函数，有时还需要控制其长度、范围等，而产生伪随机数的方法也随着需求的增多更加多样化.

三、定积分的随机计算方法

（一）用随机投点法计算定积分

设 $f(x)$ 是 $[0, 1]$ 上的连续函数，且 $0\leqslant f(x)\leqslant 1$. 需要计算积分值 $I=\int_0^1 f(x)\mathrm{d}x$. 积分 I 等于图 1-1 中的面积 G.

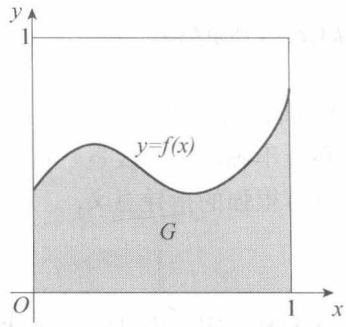


图 1-1　随机投点法计算积分示例

在图 1-1 中所示的单位正方形内均匀地做投点试验，则随机点落在曲线 $y=f(x)$ 下方的概率为

$$\begin{aligned}P\{y\leqslant f(x)\}&=\iint_G \mathrm{d}x\mathrm{d}y=\int_0^1 \mathrm{d}x\int_0^{f(x)}\mathrm{d}y\\&=\int_0^1 f(x)\mathrm{d}x=I.\end{aligned}$$

假设向单位正方形内随机地投入 n 个点，其坐标为 (x_i, y_i)，$i=1, 2, \cdots, n$. 若随机点 (x_i, y_i) 落入 G 内，则 $y_i\leqslant f(x_i)$. 如果有 m 个点落入 G 内，则 $\frac{m}{n}$ 近似等于随机点落入 G 内的概率，即 $I\approx\frac{m}{n}$. 由此可设计出计算积分 I 的数值概率算法.

例 1.3　写出计算积分 $\int_0^1 x^2\mathrm{d}x$ 的随机投点法的步骤.

解：求解步骤如下：

（1）在正方形区域 $D=\{(x, y)\mid 0\leqslant x\leqslant 1, 0\leqslant y\leqslant 1\}$ 内随机投入 n 个点，坐标为 (x_i, y_i) $(i=1, 2, \cdots, n)$；

（2）计算所有满足 $y_i\leqslant x_i^2$ 的点数，并记为 m，则积分 $\int_0^1 x^2\mathrm{d}x\approx\frac{m}{n}$.

（二）用平均值法计算定积分

假设要计算积分 $I=\int_a^b g(x)\mathrm{d}x$，其中被积函数 $g(x)$ 在 $[a, b]$ 上可积.

假设 $\{x_i\}$ $(i=1, 2, \cdots, n)$ 是区间 (a, b) 上均匀分布的一组独立随机数，且函数 $y=g(x)$ 在区间 $[a, b]$ 上可积，则 $\{g(x_i)\}$ $(i=1, 2, \cdots, n)$ 也是一组相互独立且同

分布的随机数，由于 X 服从均匀分布，其概率密度函数为 $f(x)=\begin{cases}\dfrac{1}{b-a}, & a<x<b\\ 0, & \text{其他}\end{cases}$.

依概率论中的数学期望公式有

$$E[g(x)]=\int_{-\infty}^{+\infty}g(x)f(x)\mathrm{d}x=\frac{1}{b-a}\int_a^b g(x)\mathrm{d}x,$$

从而　　$\int_a^b g(x)\mathrm{d}x=E[g(x)](b-a)$,

由大数定律知 $\int_a^b g(x)\mathrm{d}x$ 依概率收敛于 $\lim\limits_{n\to\infty}\dfrac{1}{n}\sum\limits_{i=1}^{n}g(x_i)(b-a)$.

例 1.4　写出计算积分 $\int_1^3 x^2\mathrm{d}x$ 的平均值法的步骤.

解：步骤如下：

(1) 在积分区间 [1，3] 内随机投入 n 个点，产生一个随机点列 $\{x_i\}(i=1,2,\cdots,n)$；

(2) 计算 $\left(\dfrac{1}{n}\sum\limits_{i=1}^{n}x_i^2\right)\times(3-1)$，所得结果就是积分 $\int_1^3 x^2\mathrm{d}x$ 按平均值法得到的近似值.

在上面问题的求解中，我们选择了均匀分布的随机点列，事实上这个分布可以根据实际问题的需要进行改变.

知识点链接：高等数学——定积分　二重积分；概率论与数理统计——概率定义　概率分布

第二节　网页排名、新闻分类的数学原理

一、PageRank 网页排名算法原理

PageRank（由拉里·佩奇发明）又称为网页排名，是根据网站的外部链接和内部链接的数量和质量来衡量网站价值的技术.

（一）布尔代数

计算机搜索引擎大致需要做如下工作：自动下载尽可能多的网页，建立快速有效的索引，根据相关性对网页进行公平准确的排序. 其中最重要的就是索引，而每个搜索引擎都逃不出布尔代数的框框.

参与布尔代数运算的元素只有 1（TRUE 真）和 0（FALSE 假）. 基本运算只有“与（AND）”“或（OR）”“非（NOT）”三种. 比如查询“原子能 AND 应用 AND（NOT 原子弹）”就表示查找有关原子能及其应用但不是原子弹的文献.

（二）PageRank 网页排名算法原理

对于用户的查询会有成千上万条结果，哪些网页排在前面呢？这主要取决于网页的质

量信息和这个查询与每个网页的相关性信息. 下面只是介绍衡量网页质量的方法. 在互联网上，如果一个网页被很多其他网页链接，就说明它受到普遍的认可和信赖，那么它的排名就高. 这就是 PageRank 的核心思想.

其算法如下：

假定向量 $B=(b_1, b_2, \cdots, b_N)^{\mathrm{T}}$ 为第一、二、…、N 个网页的网页排名.

$$\text{矩阵 } A=\begin{pmatrix} a_{11} & \cdots & a_{1n} & \cdots & a_{1N} \\ \vdots & & \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} & \cdots & a_{mN} \\ \vdots & & \vdots & & \vdots \\ a_{N1} & \cdots & a_{Nn} & \cdots & a_{NN} \end{pmatrix}$$

为网页之间链接的数目，其中 a_{mn} 代表第 m 个网页指向第 n 个网页的链接数. A 是已知的，B 是未知的、所求的.

假定 B_i 是第 i 次迭代的结果，那么

$$B_i=A \cdot B_{i-1} \tag{1.3}$$

初始假设：所有网页的排名都是 $1/N$，即

$$B_0=\left(\frac{1}{N}, \frac{1}{N}, \cdots, \frac{1}{N}\right)^{\mathrm{T}}.$$

显然通过式（1.3）简单的矩阵运算（然而计算量是巨大的），可以得到 B_1，B_2，….可以证明 B_i 最终会无限趋近于 B. 此时 $B=A \cdot B$. 故当 B_i 和 B_{i-1} 之间的差异非常小，接近于零时，停止迭代运算，算法结束. 一般地，10 次左右的迭代误差就达到要求.

由于网页之间链接的数量相比互联网的规模非常稀疏，因此计算网页排名也需要对零概率或者小概率事件进行平滑处理. 网页排名是一维向量，对它的平滑处理只能利用一个小的常数 α. 这时式（1.3）变成

$$B_i=\left[\frac{\alpha}{N} \cdot E+(1-\alpha)A\right] \cdot B_{i-1}$$

其中 N 是互联网网页的数量，E 是单位矩阵.

网页排名的计算主要是矩阵相乘，这种计算很容易分解成许多小任务，在多台计算机上并行，这个算法被公认为是文献搜索的最大贡献之一.

二、向量夹角用于新闻分类

互联网上的众多新闻按内容进行分类可以采用如下方法：

比如，将词汇表中的 64 000 个词按词典顺序编号. 这样把词典的大小限制在 65 535 个词以内，在计算机中只要两个字节就可以表示一个词，之后把每个词的 TF-IDF 值计算出来. 下面粗略了解一下 TF-IDF 的概念. 在 TF-IDF（Term Frequency/Inverse Document Frequency）中，Term Frequency 是指网页中按长度对关键词出现的次数的归一化，

也就是关键词出现的次数除以网页的总字数，或者称为关键词的频率或单文本词频. 如果关键词只在很少的网页中出现，通过它就容易锁定目标，它的权重也就应该大；反之，如果一个词在大量网页中出现，看到它无法清楚地知道要找什么内容，它的权重就应该小. 概括地讲，假定一个关键词 w 在 D_w 个网页出现过，那么 D_w 越大，w 的权重越小，反之亦然. 故在信息检索中，使用最多的权重是“逆文本频率指数”(Inverse Document Frequency)，它的公式为 $\log\left(\frac{D}{D_w}\right)$，其中 D 是全部网页数.

现在回到每篇新闻的 TF-IDF 值上来，对应每篇新闻 64 000 个词有 64 000 个 TF-IDF 值，这相当于对应每篇新闻有一个维数为 64 000 的向量. 下面的问题就是如何比较这些向量的相似度，从而对新闻进行分类. 举例来说，在金融类的新闻中，股票、利息、债券、基金、银行、物价、上涨这类词多，而二氧化碳、宇宙、诗歌、木匠、包子之类的词少. 反映在向量上，类似的新闻向量在某几个维度的值都比较大，而其他维度的值都比较小. 但对于一篇 10 000 字的文本和一篇 500 字的文本，单纯考虑其维度还不够. 无论如何，向量之间的夹角如果小，文本的相似度还是比较高的，反之亦然.

故假如新闻 X 和新闻 Y 对应的 TF-IDF 值向量分别是

$$X=(x_1, x_2, \cdots, x_{64\,000})^{\mathrm{T}}, \quad Y=(y_1, y_2, \cdots, y_{64\,000})^{\mathrm{T}},$$

它们夹角的余弦等于

$$\cos\theta=\frac{x_1y_1+x_2y_2+\cdots+x_{64\,000}y_{64\,000}}{\sqrt{x_1^2+x_2^2+\cdots+x_{64\,000}^2}\cdot\sqrt{y_1^2+y_2^2+\cdots+y_{64\,000}^2}}. \tag{1.4}$$

当两条新闻向量夹角的余弦 $\cos\theta$ 等于 1 时，向量夹角为 0，两条新闻完全相同；当夹角余弦接近于 1 时，两条新闻相似，从而可以归为一类；夹角余弦越小，夹角越大，两条新闻越不相关.

在实际计算时，由于网页数量巨大，通过自底向上不断合并的方法，即相似性大于一个阈值的新闻合并成一个小类，再把每个小类新闻作为一个整体，重新计算向量，再计算小类之间余弦的相似性，然后合并大一点的小类，不断重复，直至达到需要的结果.

知识点链接：线性代数——向量夹角　夹角余弦；概率论与数理统计——概率定义　小概率事件

第三节　奇异值分解、数组与压缩矩阵

一、奇异值分解的方法和应用场景

假设用一个大矩阵 A 来描述成千上万篇文章和几十万甚至上百万个词的关联性. 在这个矩阵中，每一行对应一篇文章，每一列对应一个词，如果有 N 个词和 M 篇文章，则就

是一个 $M\times N$ 阶矩阵.

$$A=\begin{pmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1N} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & \cdots & a_{ij} & \cdots & a_{iN} \\ \vdots & & \vdots & & \vdots \\ a_{M1} & \cdots & a_{Mj} & \cdots & a_{MN} \end{pmatrix}$$

a_{ij} 就表示第 j 个词在第 i 篇文章中出现的加权词频（比如用词的 TF-IDF 值）.

奇异值分解可以将大矩阵分解成三个小矩阵，比如 $M=1\,000\,000$，$N=500\,000$，100 万乘以 50 万，即 5 000 亿个元素，经由如下矩阵分解

$$A_{1\,000\,000\times 500\,000}=X_{1\,000\,000\times 100}B_{100\times 100}Y_{100\times 500\,000}$$

即把矩阵 A 分解成一个 100 万乘以 100 的矩阵 X、一个 100 乘以 100 的矩阵 B 和一个 100 乘以 50 万的矩阵 Y. 这三个矩阵的元素总数约 1.5 亿，不到原来的三千分之一. 相应的存储量和计算量都会小很多. 同时这三个矩阵都有非常清晰的物理意义.

第一个矩阵 X 是对词分类的一个结果，它的每一行表示一个词，每一列表示一个语义相近的词类或者称为语义类. 每一行的每个非零元素表示这个词在每个语义类中的重要性（或者说相关性），数值越大，越相关.

这里举一个小例子，若

$$X=\begin{pmatrix} 0.7 & 0.15 \\ 0.22 & 0.49 \\ 0 & 0.92 \\ 0.3 & 0.03 \end{pmatrix},$$

这里有四个词和两个语义类，第一个词和第一个语义类比较相关（相关性 0.7），和第二个语义类不太相关（相关性 0.15）. 第二个词正好相反，第三个词只和第二个语义类相关，和第一个语义类完全无关. 第四个词和每一个类都不太相关，比较而言和第一个语义类相关度略大，为 0.3.

最后一个矩阵 Y 是对文本的分类结果. 它的每一列对应一个文本，每一行对应一个主题. 每一列中的每个元素表示该篇文本在不同主题中的相关性. 同样以小矩阵为例.

$$Y=\begin{pmatrix} 0.7 & 0.15 & 0.22 & 0.39 \\ 0 & 0.92 & 0.08 & 0.53 \end{pmatrix}$$

这里有四篇文本和两个主题. 第一篇文本属于第一个主题. 第二篇文本和第二个主题的相关性为 0.92. 第三个文本和两个主题都不太相关，比较而言靠近第一个主题. 第四篇文本和两个主题都有一定的相关性，和第二个主题更近（0.53）.

中间的矩阵

$$B=\begin{pmatrix} 0.7 & 0.21 \\ 0.18 & 0.63 \end{pmatrix},$$

在矩阵 B 中，第一个词的语义类和第一个主题相关，和第二个主题没有太多关系．而第二个词的语义类则相反．

因此，只要对关联矩阵 A 进行一次奇异值分解，就可以同时完成近义词和文章的分类．同时，还能得到每个主题和每个词的语义类之间的相关性．对于矩阵的奇异值分解，可以参见矩阵特征值的谱分解理论，对于普通矩阵，MATLAB 等计算软件就可以实现矩阵的奇异值分解．

二、数组定义中的数学原理

数组是常用的数据结构，大多数程序设计语言都提供数组描述数据．数据结构的顺序存储结构多采用数组来描述．

一维数组 $A[n]$ 是由$(a_1, a_2, \cdots, a_{n-1}, a_n)$组成的有限序列．

二维数组 $A[m][n]$ 是由 $m\times n$ 个元素组成的，与矩阵结构相同，有 $A_{m\times n}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$．由于计算机的存储单元是一维结构，而多维数组是一个多维结构，因此用一维连续的存储单元存放多维结构就必须按照某种次序将数组中的元素排成一个线性序列．

可以把 $A_{m\times n}$ 看成 m 个行向量组成的向量，也可看成 n 个列向量组成的向量．

如果按行优先的顺序，则 $m\times n$ 个元素的线性序列为

$$((a_{11}, a_{12}, \cdots, a_{1n}), (a_{21}, a_{22}, \cdots, a_{2n}), \cdots, (a_{m1}, a_{m2}, \cdots, a_{mn})).$$

如果按列优先的顺序，则 $m\times n$ 个元素的线性序列为

$$((a_{11}, a_{21}, \cdots, a_{m1}), (a_{12}, a_{22}, \cdots, a_{m2}), \cdots, (a_{1n}, a_{2n}, \cdots, a_{mn})).$$

同理，三维数组 $A[m][n][p]$ 由 $m\times n\times p$ 个元素组成．类似地，n 维数组 $A[t_1][t_2]\cdots[t_n]$ 由 $t_1\times t_2\times\cdots\times t_n$ 个元素组成．如果可以将三维数组视为 m 个二维（$n\times p$）数组，那么 n 维数组可视为 t_1 个 $n-1$ 维（$t_2\times\cdots\times t_n$）数组．记为

$$a_1[t_2][t_3]\cdots[t_n], a_2[t_2][t_3]\cdots[t_n], \cdots, a_{t_1}[t_2][t_3]\cdots[t_n].$$

按顺序存储方法，先存储第一个 $n-1$ 维数组 $a_1[t_2][t_3]\cdots[t_n]$，再存储第二个 $n-1$ 维数组 $a_2[t_2][t_3]\cdots[t_n]$，直到最后存储第 t_1 个 $n-1$ 维数组 $a_{t_1}[t_2][t_3]\cdots[t_n]$．

例如三维数组 $a[2][3][4]$ 由 $2\times3\times4$ 个元素组成．可将其视为 2 个二维 3×4 数组，记为 $a[3\times4]$，$b[3\times4]$．将 $a[3\times4]$ 按行排的线性序列为

$$((a_{11}, a_{12}, a_{13}, a_{14}), (a_{21}, a_{22}, a_{23}, a_{24}), (a_{31}, a_{32}, a_{33}, a_{34})),$$

同理 $b[3\times4]$ 按行排的线性序列为

$$((b_{11}, b_{12}, b_{13}, b_{14}), (b_{21}, b_{22}, b_{23}, b_{24}), (b_{31}, b_{32}, b_{33}, b_{34})).$$