浅谈补集转化思想在统计问题中的应用

目录

前言

不管是在实际生活中还是在信息学竞赛里，都常常会遇到统计问题，即对满足某些性质的对象进行计数的问题。

解决统计问题有一些常用的方法比如离散化、极大化、二分、事件表等等，利用它们基本上可以解决大多数的统计问题。我们不妨将这些方法称为解决统计问题的常规方法。

然而世上的任何事物都是既有普遍性，又有特殊性的。确实存在一部分统计问题，用常规方法是很难甚至是根本无法解决的。对于这些问题，就应该具体情况具体分析，采用非常规的解法。

本文中将要讨论的就是这些非常规解法中的一种——利用补集转化思想解决统计问题。

关键字

统计问题　常规／非常规（统计）方法　枚举　时间／思维／编程复杂度
补集转化　组合计数　有序化　枚举对象　枚举量

摘要

本文通过对两个例题——单色三角形问题（POI9714）和海战游戏（改编自 Ural1212），探讨了补集转化思想在统计问题中的应用，分析比较了补集转化思想在两个例子中的作用效果和价值。

最后得出结论：补集转化思想应用于统计问题中往往有着很好的效果，我们应该注意培养逆向思维，掌握好这种非常规的统计方法。

正文

统计问题，我们认为是对于满足某些性质的对象进行计数的问题。既然是计数，那么不可避免地，其解法或多或少地建立于枚举之上。在很多情况下，“枚举”往往是低效的代名词。因此，我们通常所见到的统计问题，其最好解法动辄就达到 $O(n^2)$、$O(n^3)$的时间复杂度，也因此统计问题的规模往往不是很大，一般都在 1000 以内。

如果光是这样也就罢了，更糟糕的是很多统计问题，按照直观的想法设计出来的算法往往具有令人望而生畏的时间复杂度：要么 n 的指数过高，要么题目中给的 n 就是很大的，或者时间复杂度是关于另外一个很大的量 M（往往是表格的尺寸、图的边数等等）的表达式……总之，这个时候的统计问题实在不那么讨人喜欢。

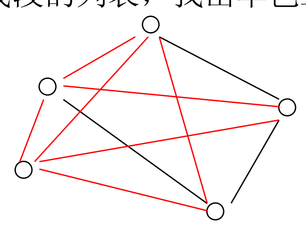
于是我们又需要利用很多技巧来降低复杂度，离散化和极大化思想、二分法、事件表等技术都是非常有用的，在它们的帮助下，我们成功地解决了非常多的刁钻的统计问题，以至于这些技巧也已经成了理所当然的“常规”方法，很多本来棘手的统计问题也成了常规题。然而这些方法显然也有不那么奏效的时候，这时我们就需要一些非常规的方法来解决一些更加棘手的问题。

这些非常规方法之一就是利用补集转化思想来帮助解决统计问题。补集转化思想在很多方面有着广泛的应用，让我们来看看在解决统计问题方面它又有哪些精彩表现吧！

例一　单色三角形问题（POI9714 TRO）

题目大意

空间里有 n 个点，任意三点不共线。每两点之间都用红色或黑色线段（只有一条，非红即黑！）连接。如果一个三角形的三条边同色，则称这个三角形是单色三角形。对于给定的红色线段的列表，找出单色三角形的个数。例如图一中有



图一

5 个点，10 条边，形成 3 个单色三角形。

输入点数 n、红色边数 m 以及这 m 条红色的边所连接的顶点标号，输出单色三角形个数 R。3<=n<=1000，0<=m<=250000。

初步分析

很自然地，我们想到了如下算法：用一个 1000*1000 的数组记录每两个点之间边的颜色，然后枚举所有的三角形（这是通过枚举三个顶点实现的），判断它的三条边是否同色，如果同色则累加到总数 R 中（当然，初始时 R 为 0）。

这个算法怎么样呢？姑且不论它非常奢侈地需要一个 1M 的大数组（POI97 的时候还没有大内存），它的时间复杂度已经高达 O(C(n,3))，也就是 $O(n^3)$的级别。对于 n 最大达到 1000 的本题来说，实在不能说是个好算法。

那些常规的技巧能不能用到本题上呢？离散化和极大化看起来跟本题毫不沾边。由于本题的限制条件非常少，也不是求最值型的计数问题，所以二分和事件表看来也用不上。

看来，想要循规蹈矩地解决这题是不可能了，我们需要全新的思路！

深入思考

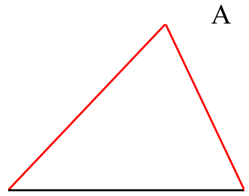
稍稍进一步分析就会发现，本题中单色三角形的个数将是非常多的，所以一切需要枚举出每一个单色三角形的方法都是不可能高效的。

单纯的枚举不可以，那么组合计数是否可行呢？从总体上进行组合计数很难想到，那么我们尝试枚举每一个点，设法找到一个组合公式来计算以这个点为顶点的单色三角形的个数。

这样似乎已经触及到问题的本质了，因为利用组合公式进行计算是非常高效的。但是仔细分析后可以发现，这个组合公式是很难找到的，因为对于枚举确定的点 A，以 A 为一个顶点的单色三角形 ABC 不仅要满足边 AB 和边 AC 同色，而且边 BC 也要和 AB、AC 边同色，于是不可能仅仅通过枚举一个顶点 A 就可以确定单色三角形。

经过上面的分析，我们得出枚举＋组合计数有可能是正确的解法，但是在组合公式的构造上我们遇到了障碍。这个障碍的本质是：

从一个顶点 A 出发的两条同色的边 AB、AC 并不能确定一个单色三角形



图二

ABC，因为 BC 边有可能不同色。

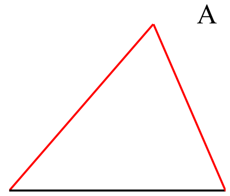
也就是说，我们无法在从同一个顶点出发的某两条边与所有的单色三角形之间建立一种确定的对应关系。

补集转化

让我们换一个角度，从反面来看问题：因为每两点都有边连接，所以每三个点都可以组成一个三角形（单色或非单色的），那么所有的三角形数 $S=C(n,3)=n*(n-1)*(n-2)/6$。又因为单色三角形数 R 加上非单色三角形数 T 就等于 S，所以如果我们可以求出 T，那么显然，$R=S-T$。于是原问题就等价于：怎样高效地求出 T。

纯枚举的算法想都不用想就被排除，那么在上面分析中夭折的枚举＋组合计数的算法又怎样呢？这个算法原先的障碍是无法在“某两条边”与“单色三角形”之间建立确定的对应关系。那么有公共顶点的某两条边与非单色三角形之间是否有着确定的关系呢？对了！这种关系是明显的：

非单色三角形的三条边，共有红黑两种颜色，也就是说，只能是两条边同色，另一条边异色。假设同色的两条边顶点为 A，另外两个顶点为 B 和 C，则从

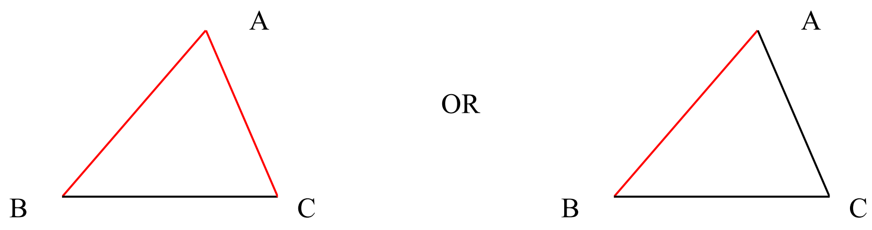


图三

B 点一定引出两条不同色的边 BA 和 BC，同样，从 C 点引出两条不同色的边 CA 和 CB。

这样，一个非单色三角形对应着两对“有公共顶点的异色边”。

另一方面，如果从一个顶点 B 引出两条异色的边 BA、BC，则无论 AC 边是何种颜色，三角形 ABC 都只能是一个非单色三角形。也就是说，一对“有公共顶点的异色边”对应着一个非单色三角形。



图四

很明显，我们要求的非单色三角形数 T 就等于所有“有公共顶点的异色边”的总对数 Q 的一半。而这个总对数是很好求的：每个顶点有 n-1 条边，根据输入的信息可以知道每个顶点 i 的红边数 E[i]，那么其黑边数就是 n-1-E[i]。枚举顶点 A，则根据乘法原理，以 A 为公共顶点的异色边的对数就是 E[i]*(n-1-E[i])。

所以 $Q = \sum_{i=1}^{n} E[i] * (n-1-E[i])$

求出 Q 之后，答案 R=S－T=n*(n-1)*(n-2)/6-Q/2。这个算法的时间复杂度仅为 O(m+n)，空间复杂度是 O(n)，非常优秀。

小结

在这个例子中，我们经历了如下的思维过程：

直观的枚举算法无法实现，常用的技巧没有用武之地

↓

考虑枚举＋组合计数

↓

正面分析，组合公式很难构造

↓

从反面思考（补集转化思想），得到突破

补集转化思想在其中起着至关重要的作用：通过补集转化，我们才能够在原来无法联系起来的“边”和“三角形”之间建立起确定的关系，并以此构造出组合计数的公式。这样，就由单纯的枚举算法改为了枚举+组合计数的算法，大大降低了时间和空间复杂度。在这里，补集转化思想的作用体现在**为找到一个本质上不同的算法创造了条件**。

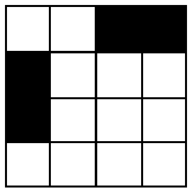
例二　海战游戏（改编自 Ural1212 Sea Battle）

题目大意

“海战游戏”是在一个 N 行 M 列的方格棋盘上摆放“军舰”，一艘军舰是连在一起的 X 行 Y 列方格，每个方格都全等于棋盘上的格子，于是军舰就可以摆放在棋盘上，使军舰的每个格子和棋盘的格子重合。但摆放时必须遵守如下规则：任意两艘军舰的任意两个格子不得重合或相邻（指在上、下、左、右、左上、右上、右下、左下八个方向上相邻）。现在已经摆放了 L 艘军舰（符合摆放的规则），

下一步想要再摆放一个 P 行 Q 列的军舰，求出共有多少种不同的可能摆放方案。输入 N、M、L，已经摆放的 L 艘军舰的信息（左上角和右下角的行列坐标），以及下一步要摆放的军舰的大小 P、Q，输出方案数 R。其中 2<=N,M<=30000，0<=L<=30，1<=P,Q<=5。我们认为所有已摆放的军舰大小都在 P*Q 这样的规模。

例如图五中已经摆放了一个 1 行 2 列的军舰和一个 2 行 1 列的军舰，如果我



图五

们要再摆一个 1 行 2 列的军舰，有两种方案。如果要再摆一个 2 行 2 列的军舰，只有一种方案。

初步分析

枚举每一种摆放方式，再分别进行判断是否符合规则的方法显然是不行的，因为需要枚举的摆放方式已经在 O(N*M)的级别，更不要说还需要嵌套一个判断的过程了。

实际上，原题就是给定了一个网格，上面某些矩形区域已经被占用，现在要在里面放入一个新的矩形，不能和已被占用的格子重合或是相邻。这是一种典型的在有障碍点的网格上求摆放方案数的统计问题。对于看起来如此经典的问题，用常规的方法能否解决呢？实际上，用离散化可以设计出能够接受的算法。由于这不是本文的讨论内容，所以我们不多做研究，只给出初步的结论是：离散化的算法时间复杂度为 O((M+N)*L)，虽然对于原题勉强可以应付，但是一旦数据规模再稍稍扩大一点，必定超时。而且离散化的算法思考比较复杂，编程比较烦琐。

如果想要圆满地解决这道题，我们需要寻找新的算法。

几个工具

在进一步地思考之前，我们先明确几个小问题，以作为下面研究的工具。

- 在一个 X 行 Y 列的矩形 A 中放入一个 P 行 Q 列的矩形 B，共有多少种摆放方案？

结论一：矩形 B 能够放入矩形 A 中的充要条件是 X>=P 且 Y>=Q，所以如果 X<P 或 Y<Q，方案数为 0。否则矩形 B 的左上角可以位于矩形 A 的 1 至 X-P+1 行，1 至 Y-Q+1 列，也就是总共有(X-P+1)*(Y-Q+1)种摆放方案。

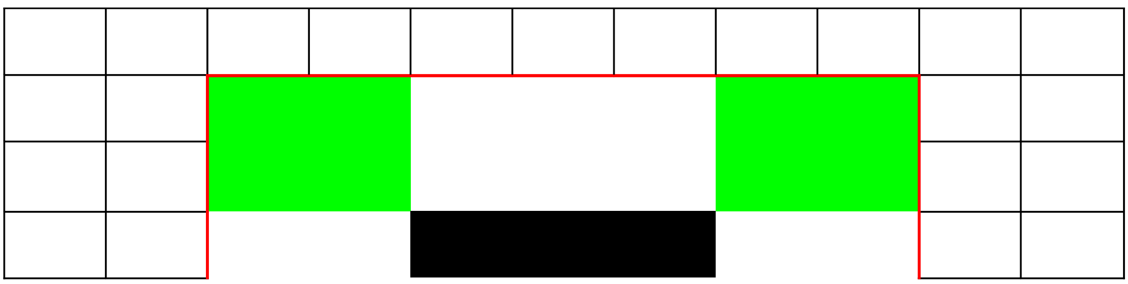
- 矩形 A 的左上角为(AX1,AY1)，右下角为(AX2,AY2)，矩形 B 的左上角为(BX1,BY1)，右下角为(BX2,BY2)，如果存在某两个格子 a∈A，b∈B 且 a、b 相邻或重合，就称 A 和 B“相交”。如何判断 A、B 是否相交？

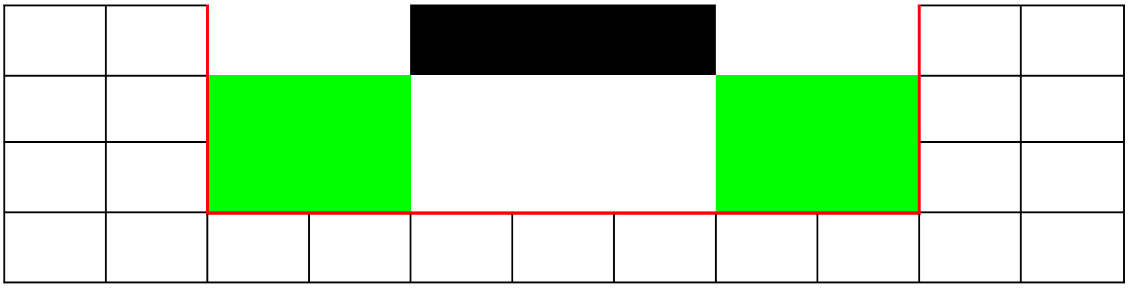
这个问题稍稍复杂一点，但是仔细分析各种情况之后可以得出**结论二**：A 和 B 相接的充要条件是：

(BX1<=AX2+1) and (BX2>=AX1-1) and (BY1<=AY2+1) and (BY2>=AY1-1)。

- 设一个已经摆放的矩形 A 为 X 行 Y 列，新摆放的矩形 B 为 P 行 Q 列，矩形 B 怎样摆放才能和矩形 A 相交呢？根据结论二我们直接就可以得出

结论三：矩形 B 能够与 A 相交的所有方案位于一个 2P+X 行，2Q+Y 列的矩形框内。如图六，正中的黑色矩形是矩形 A，四角的绿色矩形代表矩形 B 能够和 A 相交的摆放方式的“边界情况”。当然，这样说还不是太严密，因为这个矩形框有可能超出了棋盘的边界，此时它的边就要调整到棋盘边界内。所以我们**把结论三改为**：矩形 B 能够与 A 相交的所有方案位于一个**最多** 2P+X 行，**最多** 2Q+Y 列的矩形框内。



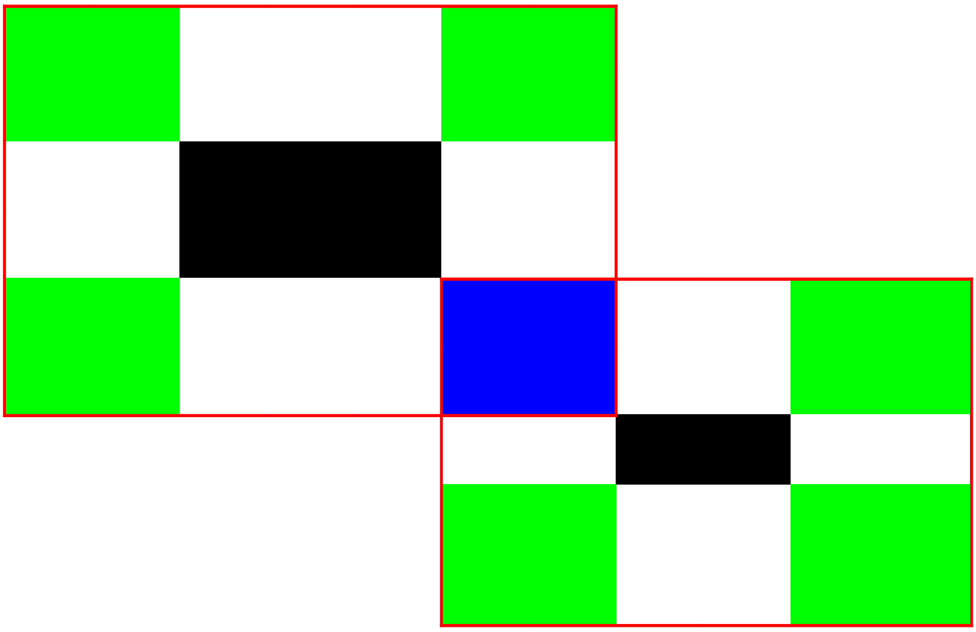


图六

补集转化

根据结论一，在给定的棋盘上不加限制摆放一个矩形，其方案数 S 是可以根据公式计算出来的。而符合规则的摆放方案数 R＋违反规则的摆放方案数 T＝S，因此 R=S-T。问题就转化为：如何高效地求出 T。T 也就是所有与已经摆放的军舰相交的方案数。

在结论三的基础上稍加判断，再结合结论一，我们实际上已经能够直接计算和任意一个已经摆放的矩形相交的方案数。但是我们并不能把和每一个已有矩形相交的方案数累加起来作为 T。因为有些摆放方案会同时和不止一个矩形相交，例如图七中，蓝色的矩形同时和两个黑色黑色已摆放矩形相交。但是按照上面的算法，这种蓝色的摆放方案会被重复计入 T，导致 T 比实际的要大。



图七

如何排除这种重复计数呢？我们采用一种排除重复的常用方法：有序化。也就是设法对于新矩形的一种摆放方案，只在处理与它相交的编号最小的已有矩

形时才允许计入总数 T。例如图七中，如果我们规定左边的黑色矩形编号较小，则在处理右边的黑色矩形时，与它相交的蓝色摆放方案就不允许计入 T，因为右边的黑色矩形并不是与蓝色方案相交的编号最小的已摆放矩形。容易证明，这样计数不会出现重复。

为了做到这一点，我们只需采取如下算法：依次处理每个已经摆放的矩形，设当前处理的矩形编号为 i，一一枚举与它相交的摆放方案，对于每个方案，再依次枚举编号为 1,2,……(i-1)的矩形，判断这些矩形能否与当前枚举的方案相交，如果发现有相交的情况，则此方案不能计入 T，否则就将 T 加 1。

根据结论三，与每个已摆放的 X 行 Y 列的矩形相交的摆放方案位于它周围的一个矩形框内，这个矩形框最多 2P+X 行，2Q+Y 列，再根据结论一，在其中摆放 P 行 Q 列的矩形最多只有(P+X+1)*(Q+Y+1)种方案，由于每个矩形的大小均在 P*Q 这样的级别，所以总共需要处理的方案数规模为 $O(P^2Q^2L)$，而对于每个方案，最多只需要枚举 L－1 个已摆放矩形判断是否与之相交，根据结论二，判断两个矩形是否相交的复杂度为 O(1)，所以处理每个方案的复杂度为 O(L)，因此整个算法的复杂度仅为 $O(P^2Q^2*L^2)$，非常优秀，大大领先于离散化的常规方法。而且，一旦想到了补集转化，则下面的分析、算法设计都非常自然，程序基本上都是循环枚举和简单的判断。可以说，在思维复杂度和编程复杂度上，基于补集转化的算法比离散化的常规算法好得多。

小结

在这个例子中，离散化思想能够帮助我们解决原题，但是时间、思维和编程复杂度都很高。但是利用补集转化思想，我们却可以设计出全面超越离散化算法的新算法。

本题从正面考虑，枚举量太大，所以常规的解法是采用离散化技巧来减少枚举量。但是从反面考虑，枚举量就非常小了。补集转化思想在这里起到的作用是**帮助我们选择了合适的枚举对象，从而减少了枚举量**。

总结

本文中的两个例子都是利用补集转化思想解决统计问题，它们在思想上有着明显的相同点：如果目标 R 比较难以求解，而它的补集 T 以及 R 和 T 的总和

S 相对容易求出，那么不妨从反面考虑，求出 S 和 T，也就间接地求出了 R。

但是补集转化思想体现在两个具体的例子中，又有所不同：

从作用效果上来看，在例 1 中，补集转化思想指导我们设计出了本质上不同于单纯枚举的算法—枚举＋组合计数，可以说是“另辟奚径”；而在例 2 中它并没有改变算法的本质，而只是通过改变枚举对象减少了枚举量。由此可见，**补集转化思想应用于统计问题的形式是多种多样的，可以从解决问题的每个方面帮助我们**。

从意义价值上看，在例 1 中，补集转化思想似乎是解决问题的唯一可行方法；而在例 2 中，用离散化也可以解决问题，但补集转化的算法更自然、更优秀，更有普遍性。由此可见，补集转化思想不仅可以应用于一些非常规的统计问题，而且**对于一些常规算法能够解决的问题，应用补集转化思想也许可以做得更好**。

总之，**补集转化思想可以比较广泛地应用于统计问题中，是解决统计问题的一种很值得掌握的非常规思想**。

补集转化思想，体现了**矛盾对立统一，互相转化**的一种哲学观念。统计问题的本质就是在给定了对象之间的一系列关联、限制（也就是矛盾）的基础上进行计数的问题。所以在统计问题中灵活地应用补集转化思想，往往可以起到“出奇制胜”的效果。我们应该注意培养逆向思维的能力，才能用好、用活补集转化思想。

值得注意的是，利用补集转化思想解决统计问题作为一种非常规的统计方法，它和一些常规的统计方法、技巧之间的关系是辨证的。虽然在本文的例子中，补集转化思想有着很多的闪光之处，但是并不能认为常规方法一定不如非常规方法。大多数的统计问题还是适合用常规方法的，所以

只有将常规方法和非常规方法都灵活地掌握，并对于具体问题选择合适的方法，才能够游刃有余地解决统计问题。